



# Patrones de Integración de Servicios Groupware



Tesis de Grado de  
Licenciatura en Informática



<p>TES 05/23 DIF-02955 SALA</p>	 <p>UNIVERSIDAD NACIONAL DE LA PLATA FACULTAD DE INFORMÁTICA Biblioteca 50 y 120 La Plata catalogo.info.unlp.edu.ar biblioteca@info.unlp.edu.ar</p>  <p>DIF-02955</p>
---	---

Facultad de Informática,  
Universidad Nacional de La Plata  
Viernes, 25 de Noviembre de 2005

Naso Federico



BIBLIOTECA  
FAC. DE INFORMÁTICA  
U.N.L.P.

A mi familia...



# Índice

<b>1. Introducción</b>	<b>3</b>
1.1. Organización	5
<b>2. Análisis del problema</b>	<b>9</b>
2.1. Definiciones	10
2.1.1. Groupware:	10
2.1.2. Servicios groupware:	11
2.1.3. Clasificaciones de servicios y sistemas para el trabajo en grupo	12
2.1.4. Integración e Ínter-operación:	14
2.2. Principios de la integración de servicios groupware	16
2.3. Clasificación de servicios	17
2.3.1. Integraciones dentro de la clasificación	17
2.3.2. Plataformas y frameworks	17
2.4. Requerimientos para la integración de servicios groupware	19
2.4.1. Primer Requerimiento: Espacio conocido	19
2.4.2. Segundo Requerimiento: Documentar la integración	19
2.4.3. Tercer Requerimiento: Análisis del problema	20
2.5. Estado del arte en la integración de servicios Groupware	21
<b>3. Enfoque de solución</b>	<b>25</b>
3.1. Patrones de diseño	26
3.1.1. Origen	26
3.1.2. ¿Que son los patrones de diseño?	26
3.1.3. Características	27
3.1.4. Estructura	28
3.1.5. La "regla de tres".	29
3.1.6. Uso de los patrones	29
3.1.7. Problemas para encontrar patrones groupware	30
3.1.8. La "bala de plata".	30
3.1.9. Catálogos de patrones	30
3.2. Integración de servicios groupware con patrones	32
<b>4. Servicios Groupware</b>	<b>35</b>
4.1. Servicio: <i>Mail</i>	36
4.2. Servicio: <i>Chat (IRC)</i>	39
4.3. Servicio: <i>Mensajería Instantánea</i>	40
4.4. Servicio: <i>Foros</i>	41
4.5. Servicio: <i>WhiteBoards Compartidos</i>	42
4.6. Servicio: <i>Presencia</i>	43
4.7. Servicio: <i>CoWeb</i>	45
4.8. Servicio: <i>Sistemas de Manejo de contenido (CMS)</i>	48
4.9. Servicio: <i>Workflow</i>	49
4.10. Servicio: <i>Calendarios compartidos</i>	50
4.11. Servicio: <i>Videoconferencias</i>	52
4.12. Servicio: <i>Herramientas de Votación</i>	54
4.13. Servicio: <i>Sistema de control de versiones (SCV)</i>	55
4.14. Servicio: <i>Libreta de contactos compartida</i>	56

4.15. Servicio: <i>WebLogs</i> . . . . .	57
4.16. Servicio: <i>Sistemas de co-autoría (Co-Authoring Systems)</i> . . . . .	59
<b>5. Catálogo de Patrones de integración</b>	<b>65</b>
5.1. Estructura de los patrones . . . . .	66
5.2. Resumen de los patrones . . . . .	68
5.3. Patrones . . . . .	70
5.3.1. Rendezvous . . . . .	70
5.3.2. Notifier Broker . . . . .	75
5.3.3. Gossip . . . . .	79
5.3.4. Chat Channel . . . . .	83
5.3.5. Virtual Conference . . . . .	86
5.3.6. Mail Manager . . . . .	89
5.4. Relación entre los patrones . . . . .	93
<b>6. Cobertura de requerimientos</b>	<b>97</b>
6.1. Primer Requerimiento: Espacio conocido . . . . .	98
6.2. Segundo Requerimiento: Documentar la integración . . . . .	99
6.3. Tercer Requerimiento: Análisis del problema . . . . .	100
<b>7. Experiencias de Uso</b>	<b>103</b>
7.1. Desarrollos . . . . .	104
7.1.1. Rendezvous . . . . .	104
7.1.2. Los prototipos de Rendezvous . . . . .	105
7.1.3. Chat Channel . . . . .	106
7.1.4. Los prototipos de Chat Channel . . . . .	107
7.2. El Framework DyCE . . . . .	108
7.2.1. ¿Por que usar un framework? . . . . .	108
<b>8. Conclusión</b>	<b>113</b>
8.1. Alcance del catálogo . . . . .	114
8.2. Integraciones a medida . . . . .	114
8.3. Acoplamiento de servicios . . . . .	114
8.4. Trabajos relacionados . . . . .	115
8.5. Validación del catálogo . . . . .	116
8.6. Revisión de la contribución . . . . .	117
8.6.1. Contribución extra . . . . .	117
8.7. Trabajo Futuro . . . . .	117
<b>Referencias</b>	<b>121</b>
<b>9. Agradecimientos</b>	<b>129</b>

## Índice de figuras

1.	Comunicación, Colaboración y Coordinación. . . . .	11
2.	Taxonomía temporal/espacial de groupware . . . . .	12
3.	Mapa 3x3 de las opciones groupware . . . . .	13
4.	Evolución de los sistemas software según Grudin . . . . .	14
5.	Grupos de servicios por funcionalidad. . . . .	17
6.	L3-News . . . . .	37
7.	Mirc . . . . .	39
8.	Trillian Pro . . . . .	40
9.	Foro de WebInfo . . . . .	41
10.	WhiteBoard de SameTime . . . . .	42
11.	MS Outlook 2003 . . . . .	43
12.	Swiki . . . . .	46
13.	Contenido en un BSCW . . . . .	48
14.	Elementos de un Workflow . . . . .	49
15.	Yahoo Calendar . . . . .	50
16.	Evolución de la video conferencia . . . . .	52
17.	Rooms Based . . . . .	53
18.	Creación de un Poll en MeetOMatic . . . . .	54
19.	Tortoise CVS . . . . .	55
20.	Agenda de contactos de Yahoo . . . . .	56
21.	weblog de un usuario . . . . .	57
22.	Edición de un documento usando Groove . . . . .	59
23.	UMLEditor . . . . .	60
24.	Arquitectura rendezvous . . . . .	72
25.	MS Outlook 2003 con MSN Messenger . . . . .	73
26.	Netscape 7.1 con AOL Messenger . . . . .	74
27.	Arquitectura Notifier. . . . .	77
28.	BSCW . . . . .	78
29.	Dolphin . . . . .	80
30.	VITaL . . . . .	81
31.	Arquitectura Chat Channel . . . . .	84
32.	Pizarra y Chat de "virtual classroom". . . . .	85
33.	Arquitectura de una conferencia. . . . .	87
34.	Eyeball Web Communicator . . . . .	88
35.	MS NetMeeting videoconferencia y Chat. . . . .	88
36.	Envío de mail con Thunderbird . . . . .	90
37.	Arquitectura Mail Manager . . . . .	91
38.	Netscape con su agenda de contactos . . . . .	92
39.	Relación entre los patrones. . . . .	93
40.	Servicios Integrados. . . . .	94
41.	Colaboracion en el Mail . . . . .	104
42.	Mailclient . . . . .	105
43.	Snapchat . . . . .	107
44.	DyCE Component Server . . . . .	108
45.	Servicios Autonomos . . . . .	114
46.	Mapa de Patrones . . . . .	116

---

# Capítulo I

Introducción

# 1. Introducción

Clarence Ellis y sus colegas [42] en principio y Ter Hofte [53] más tarde, definen el *groupware* como "aplicaciones de software diseñadas para brindar soporte a grupos de personas en el área de comunicación y colaboración sobre objetos e información compartida".

El propósito de las *aplicaciones groupware* es brindar soporte a grupos de usuarios que se encuentran trabajando cooperativamente. Para ello, las aplicaciones *groupware*, como por ejemplo un calendario grupal o una pizarra compartida, son definidas según los *servicios groupware* que éstas brindan [70].

Una aplicación *groupware* está compuesta por uno o varios servicios *groupware*. Ejemplos de aplicaciones *groupware* y los servicios que las componen son:

**Calendarios Grupales:** permiten la coordinación de tareas, reuniones entre personas de un grupo de trabajo, se encargan de concretar las mismas teniendo en cuenta el tiempo libre de los participantes. Esta otra aplicación *groupware* brinda como servicio *groupware* el calendario compartido en donde se mantiene la información de las citas, un servicio de notificación, como por ejemplo el mail para notificar sobre las reuniones y un servicio de coordinación que puede ser una herramienta de votación para que los participantes puedan optar entre las diferentes alternativas a una reunión.

**Pizarras Compartidas:** permite a varias personas dibujar sobre un área compartida de trabajo. Las personas están comúnmente distribuidas en diferentes lugares geográficos y se comunican mediante el Chat provisto por la misma aplicación. Con este ejemplo vemos que las pizarras están provistas por dos servicios *groupware*, uno es la pizarra como un área de dibujo compartida y el otro es el Chat como herramienta de comunicación sincrónica.

Un *servicio* es definido como un conjunto de posibles interacciones entre un usuario y un sistema, donde dicho sistema, es el encargado de brindar soporte para tal interacción [36]. Subsecuentemente los *servicios groupware* permiten a los usuarios cooperar, comunicarse y colaborar utilizando información compartida [70].

Los *servicios groupware* no son algo nuevo, el mail, por ejemplo, existe desde los inicios de los sistemas operativos y de las computadoras. Es casi tan antiguo como las redes, tiene más de 30 años. En 1972, Ray Tomlinson [27] escribe el primer programa de e-mail<sup>1</sup> Otro de los servicios es el chat. El chat nació en la década del 80 con el *talk* [31] de UNIX [33]. Con estos dos ejemplos vemos que los servicios *groupware* son algo que existe desde hace mucho tiempo.

También podemos encontrar algunos servicios relativamente nuevos y que son de gran ayuda en el trabajo en grupo. Uno de ellos por ejemplo es el CoWeb [6].

---

<sup>1</sup>**E-mail:** el término e-mail significa *electronic mail*. Es decir, correo electrónico. Aunque hoy en día solo se hace referencia a *mail* a secas.



El concepto de CoWeb fue concebido y desarrollado por Ward Cunningham [34]. Un CoWeb es un ambiente de colaboración en formato hipertexto; cualquiera puede crear o modificar las páginas que lo componen sin la necesidad de contar con un editor, dado que el mismo CoWeb posee un editor específico utilizado para dicho propósito; las páginas son "*linkeadas*" por sus nombres. Mientras que al principio esto puede parecer peligroso, es también muy útil y de gran uso. Todos pueden aportar un "*pedacito*" de escritura en la Web (de a una página por vez) y, entre todos los integrantes del grupo, armar el sitio. Se debe tener cuidado en no "*pisar*" lo que otros escriben ya que todos tienen derecho o permiso de escritura sobre el site que se está armando.

Algunas aplicaciones integran *servicios groupware*. Desde 1997, Roseman y sus colegas [68] hablaban de las expectativas que se tenían sobre la integración de "herramientas" en ambientes groupware como aplicaciones únicas. Un ejemplo de esto son las pizarras compartidas de nuestro ejemplo anterior que integran Chat y dibujo colaborativo en una misma aplicación groupware.

En la actualidad, en lugar de diseñar una solución groupware en función de los servicios individuales que brinda, se la diseña como un todo, como un ambiente integral que ofrece una plataforma de colaboración a grupos de trabajo que se encuentran remotamente distribuidos y llevan a cabo interacciones electrónicas.

Mucho de estos *servicios groupware* se pueden llegar a combinar brindando nuevos tipos de servicios de más alto nivel, que inter-operen entre si para brindar nueva funcionalidad. Consecuentemente éste nuevo servicio integrado será más potente que la suma de las partes.

Por ejemplo, el MS Outlook [19] integra el MSN Messenger [59]. De esta forma se esta integrando un cliente de correo con un cliente de IM y de presencia. Otro ejemplo es el servicio usado para la planificación de reuniones (calendario grupal mencionado en el ejemplo anterior). Este tipo de servicio es combinado con el mail para notificar nuevas reuniones o para que las personas, por el mismo medio confirmen, o rechacen su participación.

La *integración de servicios groupware* se lleva a cabo por medio de ciertos aspectos que son "comunes" a los servicios que se integran [61]. Esos aspectos son el punto de unión entre los distintos servicios. Ejemplos de estos aspectos son: manejo de usuario, awarenes, sesiones, datos, etc.

Si bien estas integraciones son efectivas, para el propósito deseado y dentro de contextos específicos, es extremadamente difícil reubicarlas en otros contextos ya que existen problemas tecnológicos. Dado que la *integración de servicios groupware* está típicamente orientada a la implementación (es decir, como se integra depende de que tecnología groupware se utiliza) se plantea un problema a causa de los servicios existentes ya que utilizan tecnologías muy heterogéneas y cada vez que se necesite una solución de integración, la misma, se la deba programar de cero. Como resultado, se desvía fuerza de trabajo a aspectos de implementación, cuando sería más importante dedicarlos a la concepción y diseño de la integración de los servicios [70]. Por ejemplo, un servicio desarrollado en Smalltalk no puede ser integrado con un servicio desarrollado en JAVA.

Como podemos ver claramente en el párrafo anterior, la falta de asistencia en el diseño de *integraciones de servicios groupware*, ejemplos y arquitecturas que sirvan de guía y ayuden a realizar integraciones efectivas, conlleva a que cada vez que alguien necesite realizar una integración, la misma, se lleve a cabo desde cero. Con lo cual los problemas que ya fueron resueltos por otras personas previamente se vuelvan a resolver desde cero una y otra vez.

Una forma de abstraerse de los problemas inherentes de los diseños orientados a la implementación, es la utilización de *patrones de diseño de software*. Los mismos documentan soluciones ya probadas por medio de arquitecturas mínimas que luego pueden ser implementadas por medio de las tecnologías que se seleccionen.

Teniendo en cuenta lo planteado anteriormente, el objetivo de esta tesis es el estudio de los *servicios groupware* existentes, el desarrollo y la compilación de un catálogo de patrones [45] de servicios groupware y de patrones de *integración de servicios groupware*.

Con lo propuesto se logran satisfacer básicamente tres objetivos:

- Realizar una revisión y un catálogo del estado del arte de los servicios groupware actualmente utilizados. Y así también los usos que se le dan y las aplicaciones más conocidas que los implementan.
- Un catálogo que presenta el estado del arte en integración de servicios de groupware. Por estar basada en el concepto de patrones, de demostrada expresividad, adquiere un alto poder comunicativo.
- Documentar para cada forma de integración conocida, un método de integración de efectividad estudiada y probada. Esto se consigue gracias a que los patrones por definición documentan soluciones que han sido efectivamente implementadas (known uses<sup>2</sup>).

## 1.1. Organización

En éste capítulo se presenta la idea básica de lo que propone ésta tesis y hacia donde se intenta llegar. Además se detalla la contribución de la misma y se explica brevemente el alcance de cada capítulo.

En el *segundo* capítulo se explicará brevemente que es el groupware y se presentarán definiciones que se utilizarán en el resto de la tesis. Además en éste capítulo encontraremos que son los servicios groupware, que son las integraciones de servicios groupware, como así también los requerimientos que hay que tener en cuenta para llevar a cabo una integración entre servicios.

En el *tercer* capítulo está enfocado a los patrones de diseño, su origen, cómo y para que se usan. Además se especifica la importancia de tener catálogos de

---

<sup>2</sup>**Known Uses:** es un término que se utiliza para hacer referencia a algo ya existente y conocido.

patrones de diseño. Se explicará también que son los patrones de integración de servicios.

En el *cuarto* capítulo se realiza una revisión del estado del arte de los servicios groupware. Se documentan los servicios groupware actualmente utilizados dentro de los grupos de trabajo. Por cada servicio se dará ejemplos de uso como así también ejemplo de sistemas que usan dichos servicios.

El *quinto* capítulo es donde está documentado el catálogo de patrones de integración de servicios groupware propuesto en el capítulo uno. El mismo consta de 6 patrones documentados siguiendo una estructura similar a la que proponen los patrones de diseño.

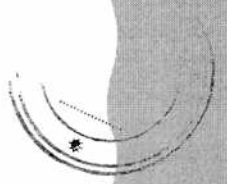
En el *sexto* capítulo se especifica cómo ésta tesis logra cubrir lo requerimientos planteados en el *segundo* capítulo.

En el *séptimo* capítulo se muestran las experiencias de uso del catálogo. En particular se instancian dos patrones específicos mediante herramientas groupware.

En el *octavo* y último capítulo de esta tesis se ven los trabajos relacionados de esta tesis, como se diferencia dicha tesis con estos trabajos. Se hace una revisión de la contribución y se plantea el trabajo a futuro.

---

# Capítulo II



Análisis del Problema

## **2. Análisis del problema**

En éste capítulo explicaremos brevemente que es el groupware y brindaremos las definiciones que son necesarias y serán utilizadas en el resto de la tesis.

Ademas en este mismo capítulo explicaremos qué son los servicios groupware y que son las integraciones de servicios groupware, como así también los requerimientos que hay que tener en cuenta para llevar a cabo una integración entre servicios.

## 2.1. Definiciones

### 2.1.1. Groupware:

Desde los años 60s se tenía la necesidad de compartir recursos de cómputo, como la memoria, las unidades de almacenamiento y principalmente el procesador; pero no se compartía la información, ya que existían diferentes barreras que lo impedían, como las distancias entre oficinas o los diferentes sistemas operativos. Con la llegada de Internet algunas barreras se rompieron, con este avance tecnológico se logró compartir información. A partir de esta necesidad se comenzó a trabajar en un nuevo tipo de tecnología: *el software colaborativo* o como se lo llama comúnmente *groupware*<sup>3</sup>.

El término **groupware** fue acuñado por primera vez por Peter y Trudy Johnson-Lenz en 1978 de la siguiente manera:

*"Procesos de trabajo en grupo que tienden a un objetivo preciso y aplicaciones concebidas para facilitar este trabajo en grupo"*

Otra definición más *formal y cercana* al software de groupware la da en 1991 Clarence Ellis, de la siguiente manera:

*"Aplicaciones de software diseñadas para brindar soporte a grupos de personas en el área de comunicación y colaboración sobre objetos e información compartida."*

El groupware es un conjunto de herramientas de software que permiten que la gente trabaje en conjunto, coordine sus esfuerzos y comparta la información de la organización; el Groupware es una plataforma que soporta la comunicación, colaboración y coordinación entre la gente de la organización o entre organizaciones sin importar la ubicación geográfica de las mismas [42]. El groupware puede ayudar a las personas y a las compañías a trascender sus fronteras.

Los sistemas groupware se están volviendo más populares dentro de las empresas, ya que resulta más barato instalar una Intranet y comprar o implementar un sistema de colaboración a estar transportando personal de un lugar a otro. Además si se necesita tomar una decisión urgente y las personas se encuentran en diferentes partes del mundo. Con los sistemas groupware se pueden tomar decisiones sin importar la distancia entre cada miembro del equipo. Es por esto que los sistemas groupware deben proporcionar tres funciones esenciales dentro de un grupo, llamadas las tres C's: *Comunicación, Colaboración y Coordinación*. [42, 65]

Estas tres C's no son independientes entre sí, cada una depende de la otra en menor o mayor medida.

- Comunicación es la transmisión de conocimiento en una organización, los colegas se comunican unos con otros de muy diversas formas: reuniones, presentaciones, charlas informales, memos, por teléfono, etc.

---

<sup>3</sup>A veces no se hace distinción entre Groupware y Computer Suported Cooperative Work (CSCW). CSCW es el área que investiga el trabajo colaborativo realizado por personas, sus espacios de trabajo, y la tecnología empleada a tal fin. También tiene en cuenta cuestiones sociológicas y antropológicas del trabajo colaborativo. Mientras que el Groupware es mucho más específico y se centra en la tecnología y el software utilizado por grupos. El groupware es uno de los productos del CSCW.

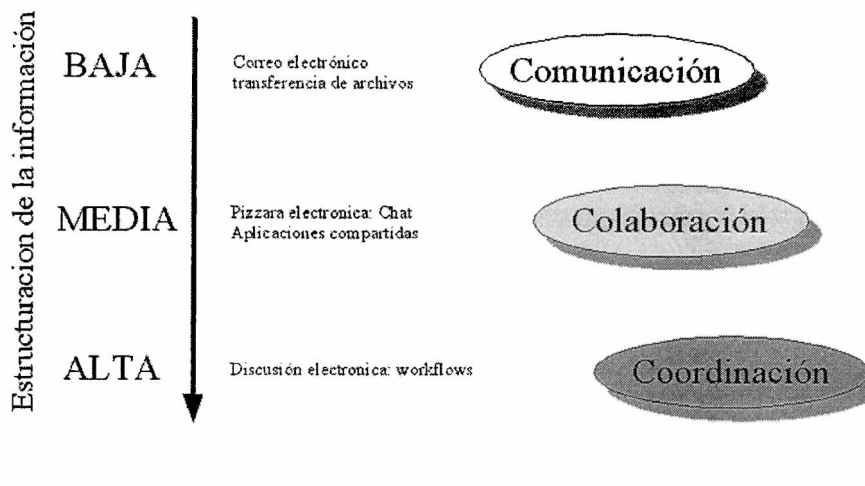


Figura 1: Comunicación, Colaboración y Coordinación.

- Colaboración requiere de un espacio compartido que puede ser una sala de juntas, una oficina, un pizarrón, etc., como un medio esencial para manejar las interacciones humanas, en otras palabras para colaborar se requiere estar en el mismo espacio, al mismo tiempo.
- Coordinación, es más que la simple automatización de una secuencia de tareas, es la ordenación de la gente o actividades involucradas en llevar a cabo proceso.

Como se puede apreciar, el groupware potencia la comunicación, colaboración, y la coordinación entre los miembros del equipo. Pero existe otro factor fundamental que es el factor humano. Sin dicho factor y compromiso por parte de los integrantes, la colaboración no podría ser posible.

### 2.1.2. Servicios groupware:

Un *servicio* es definido como un conjunto de posibles interacciones entre un usuario y un sistema, donde dicho sistema, es el encargado de brindar soporte para tal interacción [36]. Subsecuentemente los *servicios groupware* permiten a los usuarios cooperar, comunicarse y colaborar utilizando información compartida [70]. Básicamente un servicio groupware es un servicio que brinda soporte sobre algún tema específico a grupos de trabajo, como por ejemplos el mail o la mensajería instantánea. Es decir el mail, el Chat o la mensajería instantánea son *servicios groupware*.

### 2.1.3. Clasificaciones de servicios y sistemas para el trabajo en grupo

#### Colaboración asincrónica vs. Colaboración sincrónica:

Un factor clave en la colaboración es la duración del tiempo sobre el cual las colaboraciones tienen lugar. Trabajar sobre un problema complejo puede ocupar a un grupo durante meses o años. Un grupo puede estar integrado por personas las cuales pueden trabajar semi-independientemente en diferentes momentos y estar separadas geográficamente. Nos referimos a esta noción como "*colaboración asincrónica*".

Los colaboradores pueden emplear una gran parte del tiempo trabajando independientemente, pero muchas actividades requieren períodos de interacción directa, explotando, cuestionando, proponiendo, negociando, acordando, etc. Nos referimos a tales situaciones de interacciones directas entre miembros de un grupo como "*colaboración sincrónica*".

#### Clasificaciones:

Existen diferentes criterios para clasificar los servicios y sistemas para trabajo en grupo o groupware. Una primera clasificación la propuso R. Johansen [56] mediante la denominada *taxonomía temporal/espacial* de groupware, en la que se toman como dimensiones el tiempo y el espacio. (Figura 2)

	<b>SINCRONO</b> (mismo tiempo)	<b>ASÍNCRONO</b> (distinto tiempo)
<b>LOCAL</b> (mismo lugar)	<b>REUNIONES CARA A CARA</b> Pantalla compartida para explicaciones Utilidades con respuesta de la audiencia Entornos de conversación y tormentas de ideas (posible aplicación: toma de decisiones)	<b>ADMINISTRACIÓN / MANEJO DE DATOS</b> Raramente utilizado. un posible ejemplo: trabajo en turnos (en el mismo ordenador)
<b>REMOTO</b> (en lugares distintos)	<b>REUNIONES REMOTAS</b> Pizarra electrónica Charla (chat) Aplicaciones compartidas Video/tele conferencia	<b>MECANISMOS DE COORDINACIÓN</b> Transferencia de ficheros Correo electrónico Grupos de noticias (news) Foros de debate (posible aplicación: toma de decisiones) Flujo de trabajo (workflow)

Figura 2: Taxonomía temporal/espacial de groupware

En la taxonomía anterior tenemos, en primer lugar (arriba a la izquierda), los servicios o aplicaciones de los que disponen los miembros de un grupo cuando



interaccionan cara a cara. En segundo lugar (a la derecha del anterior), están los servicios que proporcionan interacción asincrónica entre los usuarios, servicios orientados mayoritariamente al manejo de datos e información. Abajo, a la izquierda, están los servicios que permiten interacción distribuida sincrónica y que, por lo tanto, facilitan la realización de reuniones entre personas distribuidas geográficamente. Y por último (abajo a la derecha), están los servicios que proporcionan interacción distribuida asincrónica, aquéllos que permiten que los usuarios además de estar distribuidos geográficamente puedan trabajar juntos sin necesidad de hacerlo en el mismo momento.

	<b>MISMO TIEMPO</b>	<b>DISTINTO TIEMPO PERO PREDECIBLE</b>	<b>DISTINTO TIEMPO PERO IMPREDECIBLE</b>
<b>MISMO LUGAR</b>	Reuniones cara a cara	Trabajo por turnos	Habitaciones de equipo
<b>DISTINTOS LUGARES PERO PREDECIBLES</b>	Vídeo/tele conferencia	Correo electrónico	Escritura colaborativa
<b>DISTINTOS LUGARES PERO IMPREDECIBLES</b>	Seminarios interactivos	Grupos de noticias	Flujo de trabajo

Figura 3: Mapa 3x3 de las opciones groupware

La clasificación que nos proporciona esta taxonomía no cubre todos los servicios o sistemas colaborativos, debido a que hay sistemas que pueden trabajar tanto de forma sincrónica como asincrónica. Además, existen distintos grados de colaboración, tema que no está especificado en esta taxonomía.

Jonathan Grudin amplía la taxonomía anterior (Figura 3) con lo que él denomina "*mapa 3x3 de opciones groupware*" [64, 49]. En ésta propuesta la actividad colaborativa puede llevarse a cabo: en "tiempo real" (mismo tiempo), para hacer referencia a distintos tipos de reuniones; en distintos momentos pero estos predecibles (distinto tiempo pero predecible), por ejemplo cuando se envía un correo electrónico a un usuario se espera que éste sea leído al poco tiempo de su recepción; en distintos momentos pero estos impredecibles (distinto tiempo pero impredecible), donde un ejemplo de actividad de este tipo es la creación en grupo de documentos.



Figura 4: Evolución de los sistemas software según Grudin

En lo referente a la dimensión espacial, la actividad colaborativa puede llevarse a cabo: en un mismo lugar, en distintos lugares pero estos conocidos por los participantes o en distintos lugares no todos ellos conocidos por los participantes. Como se puede observar (Figura 4), con groupware se denota a los sistemas que dan soporte a un grupo de colaboradores en la realización de una tarea común.

#### 2.1.4. Integración e Ínter-operación:

En el caso de integrar servicios se debe tener en cuenta dos cosas muy importantes a la hora de "unir" dos servicios, ellas son la integración y la ínter operación.

**Integración:** (Del lat. *integratio*, -onis). f. Acción y efecto de integrar o integrarse. Hacer que alguien o algo pase a formar parte de un todo.

Tiene que ver con la disponibilidad transparente de los distintos servicios desde un mismo ambiente. La idea básica del portal de servicios, con un único look and feel<sup>4</sup> y acceso a los servicios desde cualquier punto donde tiene sentido.

**Ínter- operación:**

\* inter-. (Del lat. *inter*). pref. Significa 'entre varios'. *Inter. operar*.

\* operar. (Del lat. *operari*). intr. Obrar, trabajar, ejecutar diversos menesteres u ocupaciones.

Slagter [70] define interoperabilidad como: "La interoperabilidad denota la habilidad de de compartir información por dos o mas partes de una aplicación".

<sup>4</sup>Look And feel: Es una expresión en inglés que se refiere a la imagen y, a la apariencia visual de las interfaces gráficas.

Esta definición habla de la capacidad de que los distintos servicios puedan cooperar entre sí, por ejemplo compartiendo datos. Dos ejemplos muy simples: tener un solo password<sup>5</sup> para todo (Ej. MSN passport para todos los servicios de Microsoft); tener una libreta única de direcciones que permite mandar mail, IM, fax, y ver el status de cada usuario.

---

<sup>5</sup>**El password:** Es una clave de acceso que conjuntamente con el nombre de usuario sirven para reconocer al usuario individualmente.

## 2.2. Principios de la integración de servicios groupware

Podemos definir la *integración de servicios groupware* como "la actividad de seleccionar y combinar *servicios groupware* para obtener un nuevo tipo de servicio que aporte nueva funcionalidad, potenciando los servicios básicos lo cual lo componen" [61]. Por ejemplo, teniendo en cuenta que "el todo es más que las partes", una aplicación de calendario compartido por si sola no tiene mucho uso en grupos de trabajo, pero si agregamos un servicio de comunicación para la notificación de eventos como el e-mail y un servicio que permita coordinar las reuniones como un servicio de votación, se obtiene un aplicación groupware mucho más funcional y que brinda mayor soporte a la hora de tener que programar reuniones de grupos.

La *integración de servicios groupware* se lleva a cabo por medio de ciertos aspectos que son "comunes" a los servicios que se integran. Esos aspectos son el punto de unión entre los distintos servicios. Algunos de esos aspectos son, por ejemplo:

- **Manejo de usuario:** para conseguir una integración a través del manejo de usuario, los usuarios deben tener una identidad única y ser solo una instancia en toda la aplicación, sin importar en que servicios ellos sean utilizados. Estos usuarios son los mismos en cada uno de los servicios integrados y forman el punto de unión entre los mismo. Como por ejemplo, en un servicio de mensajería instantánea la dirección de mail que sirve como ID del usuario es la misma dirección que utiliza el servicio de mail.
- **Awares o percepción:** el awaren permite conocer información sobre las acciones que están llevando a cabo otras personas en el área de trabajo [50]. Por ejemplo nos permite saber quién se encuentra trabajando con un documento en particular o quien está disponible para colaborar en ese instante. Para conseguir integración por awaren, debemos tener conocimiento de las actividades que están llevando a cabo el resto de los usuarios del área de trabajo.
- **Sesiones:** las sesiones son invocaciones a los sistemas groupware. Una sesión provee una interfase a un contexto compartido a un grupo de participantes [41]. Por ejemplo una sesión de chat entre dos usuarios, o una sesión de colaboración para editar un gráfico mediante una pizarra compartida. La sesión es el punto de integración entre los diferentes servicios groupware.
- **Datos:** los datos son los objetos a compartir dentro de los servicios integrados. Por ejemplo un documento, el dibujo de una pizarra compartida; los datos de un usuario, etc.
- **Protocolos:** los protocolos son un conjunto de reglas formales, mediante ellos se puede, por ejemplo, establecer una comunicación sincrónica entre usuarios. Para ello los usuarios deben usar un servicio que implemente dicho protocolo de comunicación. Un ejemplo de protocolos, es el protocolo IRC utilizado entre los diferentes clientes de chat.

### 2.3. Clasificación de servicios

Teniendo en cuenta el concepto de las 3C's (*Comunicación, Colaboración y Coordinación*) de Ellis [42] vistas en el capítulo 2.1 podemos clasificar a los servicios groupware en familias o grupos. Esta clasificación de los servicios ayuda a entender los patrones de manera más rápida y a clasificar nuevos servicios en los distintos grupos, como así también detectar los patrones de integración que se pueden formar entre dichos servicios.

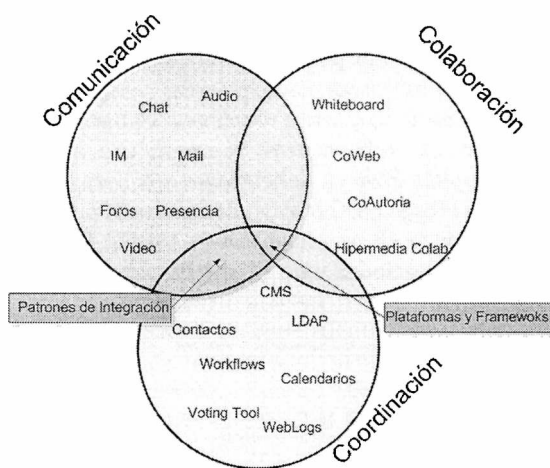


Figura 5: Grupos de servicios por funcionalidad.

En la Figura 5<sup>6</sup> podemos observar que los servicios groupware pueden ser agrupados según la funcionalidad específica que los mismo aportan.

#### 2.3.1. Integraciones dentro de la clasificación

Las integración de servicios se pueden dar entre servicios de cualquiera de los tres grupos, aún entre servicios del mismo grupos, como por ejemplo el Audio, el Video y el Chat los cuales forman un patrón llamado *Conferencia*, el cual estudiaremos y detallaremos en los capítulos siguientes. Pero es más probable, y son más productivas, que las integraciones se lleven a cabo entre servicios de diferentes grupos. Un ejemplo de esto son los servicios de Chat y de Whiteboards que formarn otro patrón llamado *Chat Whiteboards*.

#### 2.3.2. Plataformas y frameworks

En área donde confluyen todas las funcionalidades (Plataformas y Frameworks), se alojan las plataformas de colaboración las cuales proveen más de dos funcionalidades, integrando varios servicios groupware en aplicaciones concretas. En dicha area podemos encontrar las plataformas que brindan soporte integral a los grupos de trabajo distribuidos. Ejemplos de estas plataformas son: Lotus

<sup>6</sup>En la figura solo se tienen en cuenta los servicios groupware involucrados en esta tesis. Los mismos son detallados en el capítulo 5.

Notes [14], GroupWise [10], Microsoft Exchange Server [18], etc. Las mismas son soluciones integrales y cerradas.

Además podemos encontrar en la misma área framework que proveen soporte para el desarrollo de plataformas o de aplicaciones groupware ad-hoc. Los mismos brindan los servicios básicos (comunicación, manejo de usuarios, manejo de sesiones, etc) ya integrados y algunos de los hotspots<sup>7</sup> necesarios para el resto de las integraciones. Ejemplos de estos frameworks son: Lotus Sametime [15], GroupKit [9], Habanero [11], ANTS [1], etc.

GroupKit [9] ha sido desarrollado por la Universidad de Calgary. Esta herramienta nos permite crear, entre otras cosas, pizarras compartidas, juegos multiusuario o herramientas para dar soporte a reuniones. Habanero, desarrollado por la Universidad de Illinois, es un framework que permite transformar aplicaciones escritas en Java o applets Java en aplicaciones colaborativas. Y por último, ANTS es un entorno de trabajo colaborativo desarrollado por la Universidad de Murcia en colaboración con la Universidad Rovira Virgili. Este entorno permite la creación de potentes aplicaciones para el aprendizaje colaborativo o aplicaciones CSCL.

---

<sup>7</sup>Hotspots: Las áreas del framework que se deben codificar explícitamente.

## **2.4. Requerimientos para la integración de servicios groupware**

Para llegar a una solución en la integración de servicios groupware es aconsejable tener en cuenta ciertos requerimientos. Los mismos se detallan a continuación:

### **2.4.1. Primer Requerimiento: Espacio conocido**

Se debe proveer un catálogo de integración de servicios que describa el espacio conocido en términos atómicos. Es decir, un catálogo en donde estén incluidos cada uno de los servicios groupware y la integración de los mismos. En éste espacio están clasificados cada uno de los servicios así como también la integración entre los mismos según sus respectivas familias. Esta clasificación ayuda a una utilización mas efectiva de las integraciones disponibles en el catálogo.

Idealmente, el proceso de construcción de aplicaciones utilizando un catálogo de integraciones conocidas se reduce a que cada vez que el diseñador encuentra un problema, busca en los catálogos si existe una integración que resuelve dicho problema. Si no lo encuentra, el diseñador debe pensar en una solución al problema. En cambio, si existe, el diseñador debe incluir la solución provista por el catálogo en su diseño. El diseño final estará constituido idealmente por una combinación de varias soluciones provistas por el catálogo y partes no involucradas en ninguno.

### **2.4.2. Segundo Requerimiento: Documentar la integración**

Se deben documentar las integraciones de servicios de forma clara de manera que se permitan entenderlas y diferenciarlas entre ellas. Es decir, mediante una clara especificación del problema que se resolvió a través de la integración de los servicios.

Se debe tener en cuenta en la documentación a realizar, al menos lo siguiente:

- Como ésta integración propuesta soluciona el problema planteado. Se puede obtener información en esta documentación para futuras integraciones utilizando el espectro de espacios conocidos
- Dejar en claro el contexto donde la integración se está llevando a cabo.
- Identificar cada una de los servicios que componen la integración y la funcionalidad que los mismos brindan.
- Especificar las restricciones que posee la integración.
- Mostrar las similitudes y diferencias entre las diferentes soluciones planteadas.
- Consideraciones sobre los problemas que se pueden llegar a experimentar al aplicar una u otra solución.

### **2.4.3. Tercer Requerimiento: Análisis del problema**

Para proveer una solución a la integración de servicios se debe realizar un análisis del problema que se desea solucionar. Especialmente si el problema es nuevo, es decir, si no se está familiarizado con el mismo.

En caso de que se desee utilizar una integración ya probada y documentada en el catálogo se debe tener en cuenta:

- Definir cuál es el problema que se desea atacar y el alcance del mismo.
- Leer todas las soluciones anteriormente documentadas en el catálogo para tener una idea de las soluciones que el mismo brinda.
- Observar los problemas planteados en cada una de las soluciones y verificar si alguno es similar al problema que se desea resolver.
- Analizar el contexto del problema y el ejemplo de la solución documentada.
- Identificar el patrón que soluciona el problema teniendo en cuenta el espectro de soluciones dado.



## 2.5. Estado del arte en la integración de servicios Groupware

Los servicios groupware actualmente son integrados de dos formas: mediante la utilización de frameworks o mediante el uso de protocolos estándares que conformar las reglas de negocio entre los servicios.

Un framework provee guías arquitecturales para descomponer un diseño en clases abstractas y define sus responsabilidades y relaciones. Predefine parámetros de diseño, con lo cual el diseñador solo tiene que concentrarse en aspectos específicos de su aplicación. Los desarrolladores adaptan el framework para una aplicación en particular subclassificando y compartiendo instancias de las clases de dicho framework.

La mayoría de los frameworks [9, 72, 75, 69, 57] brindan un conjunto básico de servicios, como por ejemplo la comunicación entre estos vía Chat, mail o mensajería instantánea, manejo de sesiones, manejo de usuarios, etc. Luego provee soporte para desarrollar componentes que utilicen estos servicios integrándose al framework. Esto, obviamente no es compatible con los servicios ya desarrollados sin la utilización de dicho framework, ni con las tecnologías tan heterogéneas actuales, dado que cada framework propone su propia tecnología y forma de desarrollo.

Si bien estas integraciones realizadas mediante la utilización de frameworks son efectivas, para el propósito deseado y dentro de contextos específicos, es extremadamente difícil reubicarlas en otros contextos dado que existen problemas tecnológicos. Esto es a causa de que los servicios existentes utilizan tecnologías muy heterogéneas. Por ejemplo, un servicio desarrollado en Smalltalk no puede ser integrado con un servicio desarrollado en JAVA<sup>8</sup>. Los diseños deberían plantear soluciones abstractas y sin tener en cuenta la tecnología subyacente.

En el caso de los protocolos, estos son un conjunto de reglas formales que permiten establecer comunicación de forma no ambigua entre las entidades que los implementen. Dado que son conjuntos de reglas, para usar un protocolo se debe implementar el mismo de manera íntegra. Es decir, si quisiéramos tener un cliente de mail que se comunique con otros clientes de mail de manera transparente, deberíamos implementar completamente el protocolo de envío y recepción de mail en el lenguaje que deseáramos. En este caso tanto lo que transmitimos como lo que enviamos, es decir el mail, debe respetar el estándar RFC822 [28].

A diferencia de los frameworks los protocolos son más abstractos dado que regulan, mediante reglas pactadas, el modo en que las aplicaciones debe comunicarse entre sí. En los frameworks se parte de código ya escrito y se reutiliza el mismo.

Como podemos ver en los dos casos anteriores, tanto en el uso de frameworks como en el uso de protocolos, la metodología de diseño en la integración de servicios groupware está típicamente orientada a la implementación. Éste tipo de metodologías provoca que cada vez que se necesite una solución se codifique la

---

<sup>8</sup>**JAVA**: Lenguaje de programación orientado a objetos desarrollado por SUN.  
<http://java.sun.com>

misma. De este modo se tiene una mayor consideración a los problemas inherentes a la implementación que aspectos relacionados con los *servicios groupware* [70]. Esto trae aparejado que los diseñadores no tengan una vista general en el diseño de las funciones que el sistema debe proveer y sus relaciones.

La falta de asistencia en el diseño de integraciones de servicios groupware, ejemplos y arquitecturas que sirvan de guía y ayude a realizar integraciones efectivas, conlleva a que cada vez que alguien necesite realizar una integración, la misma, se lleve a cabo desde cero. Con lo cual los problemas que ya fueron resueltos por otras personas previamente se vuelvan a experimentar una y otra vez dado que no se tiene en cuenta la experiencia recolectada en las integraciones anteriores.

---

# Capítulo III

Enfoque de Solución

### **3. Enfoque de solución**

En éste capítulo detallaremos qué son los patrones de diseño, su origen, como y para que se usan. Cuales son sus características más importantes y la estructura de los mismos. Además se especifica la importancia de tener catálogos de patrones de diseño. Se explicará también que son los patrones de integración de servicios y los requerimientos que se deben tener en cuenta para poder llevar a cabo su integración.

### 3.1. Patrones de diseño

La solución propuesta para llevar a cabo integraciones de servicios groupware es utilizar *patrones* de integración de servicios groupware. Cada *patrón* en sí, es un análisis del problema de integración y de su solución.

#### 3.1.1. Origen

La noción de *patrones de diseño* está basada al trabajo del arquitecto Christopher Alexander, manifestado en dos libros (*A Timeless Way of building* [38] y *A Pattern Language* [39]). Es decir, los precedentes a los patrones de diseño provienen del campo de la Arquitectura, Christopher Alexander a finales de los 70 escribe varios libros acerca de urbanismo y construcción de edificios, y se plantea reutilizar diseños ya aplicados en otras construcciones que cataloga como modelos a seguir.

En 1987 Ward Cunningham [34] y Kent Beck [13] utilizan las ideas de Alexander para desarrollar un lenguaje de patrones como guía para los programadores de Smalltalk [30], dando lugar al libro "*Using Pattern Languages for Object-Oriented Programs*".

Posteriormente en 1991 Jim Coplien publica el libro "*Advanced C++ Programming Styles and Idioms*", donde realiza un catálogo de "*idioms*" (especie de patrones).

Entre 1990 y 1994, Erich Gamma, Richard Helm, Ralph Johnson y Hohn Vlissides realizan el primer catálogo de patrones de diseño, que publican en el libro<sup>9</sup> "*Design Patterns: Elements of Reusable Object-Oriented Software*" [45]. Este libro, uno de los referentes a la hora de diseñar aplicaciones orientadas a objetos, documenta veintitrés patrones reconocidos que proveen soluciones aceptadas de problemas recurrentes en problemas de diseños orientados a objetos.

#### 3.1.2. ¿Que son los patrones de diseño?

Christopher Alexander y sus colegas del área de diseño de edificios, en su libro [39], dan la siguiente definición de patrón:

*"Cada patrón describe un problema que ocurre una y otra vez en nuestro entorno, para describir después el núcleo de la solución a ese problema, de tal manera que esa solución pueda ser usada más de un millón de veces sin hacerlo ni siquiera dos veces de la misma forma".*

El software ha ido aumentando su complejidad de forma pareja según aumentaba la potencia de las computadoras, el aumento del conocimiento de los usuarios, incremento de las posibilidades de la informática, de las nuevas posibilidades que ofrecen las redes e Internet, etc.

Cuando construimos una pieza de software, desde una pequeña aplicación, hasta un sistema completo distribuido, estamos realizando un trabajo de análisis,

---

<sup>9</sup>El libro se conoce como el libro del Gang of Four.

diseño y desarrollo. Esa actividad puede realizarse en grupo o individualmente. En muchas situaciones, hemos notado que la implementación de una solución, en un caso de diseño o de programación, es similar a otra que hemos adoptado en el pasado.

Suele suceder que descubrimos, más adelante, que la solución adoptada para un problema en particular, es opacada por otra solución no contemplada inicialmente. Es común encontrar pistas e ideas interesantes en los artículos y libros que vamos consultando.

Los patrones de diseño proponen una forma de reutilizar la experiencia de los desarrolladores. Para ello clasifican y describen formas de solucionar problemas que ocurren de forma frecuente en el desarrollo. Por lo tanto está basado en la recopilación del conocimiento de los expertos en desarrollo de software.

Un patrón es simplemente una forma de documentación. Los autores de patrones de diseño, documentan los mismos luego de mirar muchos proyectos de software. El "poder" de los patrones es la documentación del conocimiento previamente detectado en las experiencias, capturado y compartido [67]. Son una experiencia real, probada y que funciona. Es Historia y nos ayuda a no cometer los mismos errores.

No necesariamente al brindar una solución a un problema en un contexto, éste es un patrón. Para ello, además, necesitamos asociar la característica de recurrencia con la definición de un patrón. Los patrones deben comunicar soluciones de diseño a los desarrolladores y arquitectos que los leen y los utilizan. Aunque el concepto de patrón es bastante simple, definir realmente el término es muy complejo.

### 3.1.3. Características

- Son soluciones concretas. Proponen soluciones a problemas concretos, no son teorías genéricas.
- Son soluciones técnicas. Indican resoluciones técnicas basadas generalmente en *Programación Orientada a Objetos* (POO). En ocasiones tienen más utilidad con algunos lenguajes de programación y en otras son aplicables a cualquier lenguaje.
- Se utilizan en situaciones frecuentes; ya que se basan en la experiencia acumulada en resolver problemas reiterativos.
- Favorecen la reutilización de código. Ayudan a construir software basado en la reutilización, a construir clases reutilizables. Los propios patrones se reutilizan cada vez que se vuelven a aplicar.
- El uso de un patrón no se refleja en el código. Al aplicar un patrón, el código resultante no tiene por que delatar el patrón o patrones que lo inspiró. No obstante hay múltiples esfuerzos enfocados a la construcción de herramientas de desarrollo basados en los patrones y frecuentemente se incluye en los nombres de las clases el nombre del patrón en que se basan facilitando así la comunicación entre los desarrolladores.

- Es difícil reutilizar la implementación de un patrón. Al aplicar un patrón aparecen clases concretas que solucionan un problema concreto y que no será aplicable a otros problemas que requieran el mismo patrón. Por ejemplo: En el caso de la arquitectura, una ventana es una solución para ventilar e iluminar una habitación, pero la ventana de una casa no es útil en el camarote de un barco.

#### 3.1.4. Estructura

Un patrón describe, con algún nivel de abstracción, una solución experta a un problema. Normalmente, un patrón está documentado en forma de una plantilla. Aunque es una práctica estándar documentar los patrones en un formato de plantilla especializado, esto no significa que sea la única forma de hacerlo. Además, hay tantos formatos de plantillas como autores de patrones, esto permite la creatividad en la documentación de patrones.

Los patrones de diseño siguen una estructura descriptiva para cada uno de los mismos en donde se detallan las características principales de cada patrón.

En general los patrones tienen cuatro elementos esenciales:

- El **nombre** del patrón, es un indicio que se puede usar para describir el problema que trata de resolver el patrón.
- El **problema**, describe cuando aplicar el patrón. Explica el problema y el contexto del mismo. Algunas veces el problema incluye una lista de condiciones que deben satisfacerse antes de aplicar el patrón como solución. Estas condiciones son las fuerzas (los objetivos y restricciones) del patrón.
- La **solución**, describe los elementos que compondrán a la solución del problema. La solución no describe un problema en particular, sino uno en general. Provee las pautas de manera abstracta y debe adaptarse a cada problema en particular.
- Las **consecuencias**, son el resultado de aplicar el patrón de solución. Las mismas son importantes para evaluar alternativas, entender el costo de implementar la solución y los beneficios de aplicar el patrón.

Según el formato descrito por Gamma y sus colegas [45], un patrón además posee las siguientes características que lo describen:

- **Intención:** Un párrafo corto en donde se respondan las siguientes pregunta sobre el patrón:
  - Que es lo que el patrón debe hacer?
  - Cual es el análisis de su intención?
  - Que solución particular aporta?
- **Conocido como:** Otros nombres o alias con los cuales pueda ser conocido el patrón.

- **Motivación:** Un escenario que ilustre el problema a resolver y como el patrón lo resuelve. Este escenario ayuda a entender de una manera más abstracta el patrón en si.
- **Aplicaciones:** Cuales son las situaciones en las cuales el patrón puede ser aplicado. Es decir sus posibles usos.
- **Estructura:** Esquema gráfico que describe el patrón.
- **Participantes:** Descripción de los elementos que intervienen y sus responsabilidades.
- **Colaboraciones:** Descripción de las relaciones entre los participantes y sus respectivas responsabilidades.
- **Consecuencias:** Beneficios, riesgos, perjuicios que puede ocasionar su uso.
- **Implementación:** Información de detalle sobre la implementación, decisiones de diseño, formas de codificación.
- **Código de ejemplo:** Código de un ejemplo de utilización.
- **Usos conocidos o ejemplos de uso:** Productos y sistemas conocidos que recuerde rápidamente el lector y facilite su comprensión.
- **Patrones Relacionados:** Patrones que pueden funcionar como alternativa o de forma complementaria.

### 3.1.5. La "regla de tres".

Cuando se ha visto un problema y una solución recurrentes, se intenta identificar y documentar sus características usando una plantilla de documentación de patrones. Los candidatos a patrones no se añaden al catálogo de patrones hasta que no se ha podido observar y documentar su utilización varias veces en diferentes proyectos.

Pero *cuándo forman parte del catálogo?*: Hay una "Regla de Tres", como se la conoce en la comunidad de los patrones. Esta regla es una guía para "ascender" un patrón candidato al catálogo de patrones. De acuerdo a esta regla, una solución permanece como un patrón "candidato" hasta que se ha verificado su existencia en al menos *tres* sistemas diferentes. Ciertamente que hay mucho espacio para la interpretación en reglas como ésta, pero ayudan a proporcionar un contexto para la identificación de patrones.

Esta regla asegura que el patrón documentado se basa en aplicaciones reales y probadas y no solo en una buena idea sin ser usada realmente. [67]

### 3.1.6. Uso de los patrones

El proceso de construcción de aplicaciones utilizando patrones consiste en que cada vez que el diseñador encuentra un problema, busca en los catálogos si existe un patrón que resuelve dicho problema. Si no lo encuentra, el diseñador debe



pensar en una solución al problema. En cambio, si existe, el diseñador debe incluir el patrón en su diseño identificando las clases, métodos y atributos de la aplicación que juegan el rol de aquellos prescritos por el patrón. El diseño final estará constituido idealmente por una combinación de varios patrones y clases no involucradas en ningún patrón.

Una vez que el diseño ha sido terminado, la aplicación debe ser codificada. El programador debe implementar todos los patrones utilizados en el diseño. En los catálogos, la implementación de los patrones es descripta de manera informal, por medio de recomendaciones informales, ejemplos o pseudo código.

### **3.1.7. Problemas para encontrar patrones groupware**

En el campo de la arquitectura urbana, Alexander y sus colegas podían obtener información de las experiencias culturales que tenían más de mil años de antigüedad. Los múltiples ejemplos se pueden encontrar, y los principios subyacentes pueden ser derivados de una amplia gama de lugares. En cambio los patrones en un campo joven tal como es CSCW son más problemáticos. Las soluciones no se han empleado todavía a menudo. Sino que han sido usadas por periodos relativamente cortos de tiempo [52].

### **3.1.8. La "bala de plata".**

Se debe tener en cuenta que los patrones no son una "bala de plata", ni marcan la dirección de cómo rehusar ítems o una simple solución para las crisis de software. No transforman a novatos en expertos. Pero, proveen una simple a intuitiva solución para problemas que surgen una y otra vez recurrentemente.

### **3.1.9. Catálogos de patrones**

Los patrones han sido agrupados y organizados en catálogos. Los patrones no son entidades aisladas, sino que forman parte de un conjunto de patrones que se complementan [39] formando así un lenguaje de patrones.

Cada catálogo provee diferentes clasificaciones y descripciones. Algunos catálogos describen patrones de análisis mientras que otros patrones de diseño [63, 40, 44], incluyendo también patrones de un dominio particular o independiente del dominio.

Los catálogos de patrones son colecciones de patrones que se utilizan para resolver problemas en un determinado dominio en particular. En el caso de los patrones de Gamma y sus colegas, el dominio de los patrones, es el diseño de soluciones orientadas a objetos.

Los catálogos hacen que la reutilización de los patrones sea más fácil, clara y ordenada. Sin ellos no se podrían utilizar patrones en el diseño.

Nuevos patrones son identificados y documentados continuamente. Lamentablemente, a pesar que se han definido varios catálogos de patrones no existe aún una lista de patrones definitiva. Varios autores han escrito sobre diferentes

aspectos de los patrones, por ejemplo, formalización, recuperación automática de los patrones dados el código de una aplicación, herramientas de soporte de patrones, documentación de diseños con patrones, relación entre patrones, incorporación de patrones en el desarrollo de software, documentación y definición de patrones.

### **3.2. Integración de servicios groupware con patrones**

La solución que se propone se compone de un catálogo de servicios groupware (Capítulo 4) y un catálogo de patrones de integración (Capítulo 5).

Así como los patrones de diseño han sido agrupados y organizados en catálogos, cada uno dando diferentes clasificaciones y descripciones, los servicios groupware también pueden catalogarse según el área del problema que intentan resolver.

Los catálogos de integración de servicios son una guía estructurada que nombran, explican y evalúan integraciones recurrentes de servicios. En nuestro caso, servicios groupware. La ventaja principal de estos catálogos, es la recopilación de experiencia de integración de servicios para que las personas puedan reutilizar una solución ya probada y documentada de integración. Esto hace más fácil la integración ya que el catálogo provee técnicas y métodos para que la integración se lleve a cabo de manera eficiente y eficaz.

# Capítulo IV

## 4. Servicios Groupware

Dentro del mundo del groupware existe una gran variedad de servicios groupware básicos tales como el mail, Chat, etc. o pueden ser adaptaciones y derivados de los mismos como por ejemplo las listas de mailing que son derivados del mail tradicional.

Los servicios groupware aquí descritos son luego los servicios que formarán parte de los patrones de integración de servicios del capítulo 5. Nos enfocaremos solo en un conjunto limitado de servicios, estudiaremos aquellos servicios groupware que son más conocidos y a su vez maduros.

#### 4.1. Servicio: Mail

☞ *El mail es casi tan antiguo como las redes, tiene más de 30 años. En 1972, Ray Tomlinson escribe el primer programa de e-mail, quien es además el creador de la convención user@host. El signo @ es elegido arbitrariamente de entre los símbolos no alfabéticos del teclado. La primera versión del mail era muy primitiva, ofrecía solo la función rudimentaria de componer y leer mensajes de texto.*

Por mucho, el mail, o correo electrónico (e-mail) es el más usado de los servicios en Internet. Es uno de los primeros servicios en ser considerado dentro del groupware. Es un servicio que puede transportar todo tipo de información: documentos, audio y programas.

El formato para la distribución de mail está estandarizado bajo un protocolo llamado SMTP<sup>10</sup> que es parte de un protocolo más general llamado TCP/IP<sup>11</sup>. Para la descarga de los mails que se encuentran en el servidor, existen dos protocolos bien definidos, el IMAP<sup>12</sup> (Mediante IMAP se puede tener acceso al mail desde cualquier equipo que posea una conexión a internet. Una vez configurada una cuenta IMAP, se puede especificar qué carpetas se desean mostrar y cuales no) y el POP<sup>13</sup> (a diferencia del IMAP, el POP no necesita una conexión permanente a internet sino que cuando está establecida dicha conexión envía una solicitud al servidor pidiendo los mails almacenados para un usuario específico, los cuales se descargan en la máquina cliente). Hoy en día es casi una herramienta indispensable.

**Diferentes usos:** El mail es comúnmente usado para la comunicación asincrónica vía mensajes. Es casi imposible, pensar alguna aplicación con acceso a Internet que no soporte el envío de mail. En los paquetes de oficina por ejemplo, ya se integran las aplicaciones de planilla de cálculo o procesadores de texto con el mail. Los usuarios en lugar de grabar sus documentos, pueden enviarlos por mail. Si bien el mail permanece casi inalterable desde su creación es el servicio, más usado.

**Implementaciones:** Dentro de las implementaciones de clientes de mail podemos encontrar aplicaciones de escritorio, tales como: MS Outlook, MS Outlook Express, Netscape Mail, Eudora, TopMail, PocoMail, IncrediMail, Pegasus o aplicaciones de acceso vía Web, Yahoo, Hotmail, IMP (IMP del inglés *Internet Messaging Program* es un programa escrito en PHP para el manejo de mails), etc. La mayoría de los usuarios usa algunos de los clientes antes mencionados.

**Derivados del Mail:** utilizando el protocolo de comunicación del mail se encuentran disponibles otros servicios tales como las listas de mail y las news.

**1. Listas de mail:** Las Listas de mail ("Mail lists"), también llamadas Listas de Distribución ("Listserv"), es un sistema de difusión de información, basado en el correo electrónico y orientado a grupos de interesados en temas específicos. La base del sistema es un ordenador (el servidor de la lista) en el que un programa especial mantiene una lista con las direcciones mail de una serie de personas interesadas en un tema específico, los denominados Suscriptores<sup>14</sup>; el sistema también guarda copias de los mensajes recibidos durante un

<sup>10</sup>SMTP: Simple Mail Transfer Protocol.

<sup>11</sup>TCP/IP: Transfer Control Protocol / Internet Protocol

<sup>12</sup>IMAP: Internet Message Access Protocol.

<sup>13</sup>POP: Post Office Protocol.

<sup>14</sup>Suscriptores: se llama así a las personas que están inscritas en algún tipo de lista de distribución de mail.

cierto tiempo, por ejemplo el último mes. Aparte del mantenimiento de la lista (altas, bajas, etc.), su actividad se reduce a que, cuando recibe un mensaje para la lista, de cualquiera de los suscriptores, envía una copia a todos los demás miembros del grupo.

**Diferentes usos de las listas de mails:** Como puede verse, se trata de un medio excelente para divulgación e intercambio de conocimiento, opiniones y noticias sobre temas concretos entre grupos de interesados, que pueden ser pocos o decenas de miles, dependiendo del interés que suscite el tema tratado.

**Implementaciones:** YahooGroups, Grupos MSN, urbi, etc.

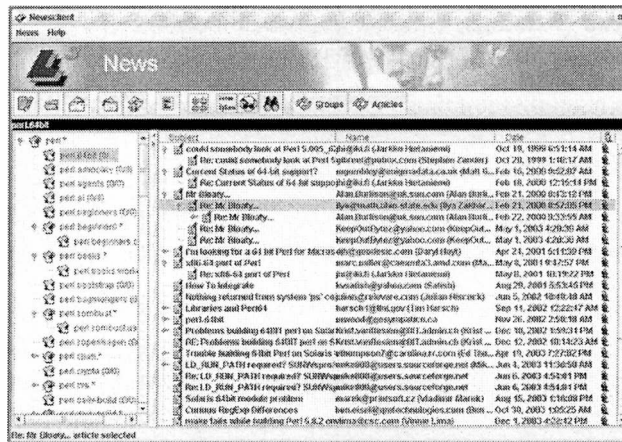


Figura 6: L3-News

**2. News:** De cierta similitud con el correo electrónico [53] y las listas de correo (mail lists), éste se basa en que en determinados puntos de la Red, existen computadoras con un programa especial, "Servidor de Noticias", que se comporta como un conjunto de grandes buzones de mensajes con la particularidad de que pueden ser consultados públicamente desde cualquier punto de la Web. A cada buzón se lo conoce como "Grupo". Estos mensajes son análogos al mail. Todo lo que se necesita para acceder a un grupo de noticias es un programa especial (cliente de noticias), con el que el usuario se conecta con el servidor utilizando un protocolo específico llamado NNTP<sup>15</sup>. Al conectar con un grupo de noticias, el aspecto es parecido a un directorio en el que en vez de nombres de ficheros aparecen las cabeceras de los mensajes, autor y fecha de envío. Mediante el mismo programa es posible "leer" los contenidos (esto último se hace bajándose el cuerpo del mensaje). El mismo programa sirve para "publicar" en el grupo de noticias, es decir, para enviar mensajes al servidor.

Los Grupos de noticias (news groups) son un servicio de Internet que comienza en 1979 con el nombre de Usenet News Service, con lo que Usenet ha quedado como sinónimo de Grupo de Noticias.

<sup>15</sup>NNTP: Network News Transfer Protocol.

Para publicar los mensajes se utiliza el mismo protocolo que utiliza el mail (SMTP).

Cuando se responde a un news en particular la respuesta se encadena al mensaje original. De ésta formas se van formando árboles de respuesta los cuales son conocidos como Threads.

**Diferentes usos:** Se utiliza para la comunicación asincrónica de noticias, bajo demanda, a un grupo de personas. Funciona como un buzón de anuncios, donde aparecen publicados todos los mensajes enviados al grupo.

**Implementaciones:** Al compartir gran cantidad del protocolo de manejo de mensajes, la mayoría de los clientes de correo dan soporte para news (Netscape, Outlook Express, PocoMail, etc). Pero también podemos encontrar aplicaciones de news dedicadas o específicas, como lo es el L3-News [75].



## 4.2. Servicio: Chat (IRC)

En el área de comunicación mediante el uso de la computadora, encontramos varios tipos de herramientas, entre ellas, el Chat. Los chats generan un espacio común entre varios usuarios, donde cada uno puede comunicarse con los demás mediante el envío de mensajes en formato de texto al resto de los participantes. La idea del Chat, es que varias personas se junten en algún lugar<sup>16</sup>, para intercambiar ideas, planificar reuniones o solamente charlar.

El Chat o IRC<sup>17</sup> es uno de los servicios más interactivos, atractivos y populares de Internet. Los Chats son algo muy común, y no hay que buscar mucho para encontrar algún canal de IRC con 20 o 30 usuarios conectados en cualquier momento del día.

☞ *El nacimiento del IRC fue en Agosto de 1988. La fecha concreta es una incógnita, pero su autor (Jarkko Oikarinen) estima que fue hacia finales de mes. Cuando el IRC empezó a tener ocasionalmente más de 10 usuarios, su autor pidió a algunos amigos que montasen también servidores de IRC en Finlandia. Así, se establecieron vínculos con servidores de la Tampere University of Technology y la Helsinki University of Technology.*

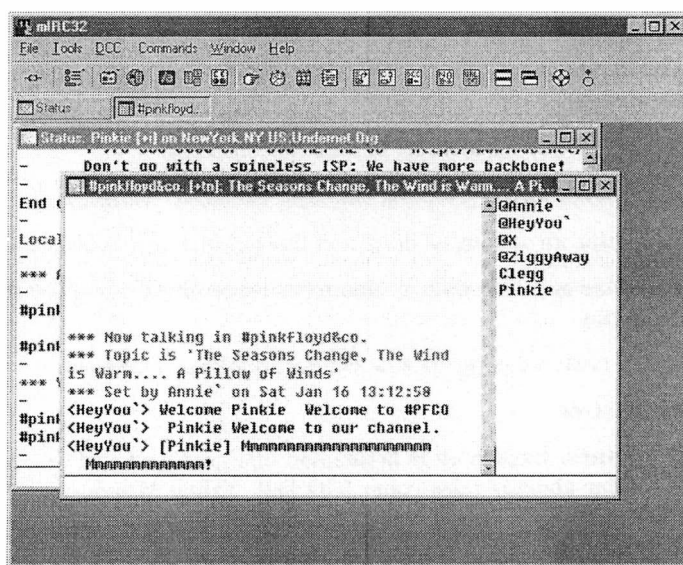


Figura 7: Mirc

**Diferentes usos:** Las herramientas de Chat son comúnmente usadas para la comunicación textual sincrónica con un grupo variado de personas. Se puede usar para "emular" discusiones o debates (en este último caso, no es lo más indicado usar un Chat tradicional, sino algún Chat específico que brinde la funcionalidad necesaria para moderar el debate), intercambiar información, etc.

**Implementaciones:** Mirc (Figura 7) es uno de las primeras implementaciones de Chat, ChatBlocks [46]: es un framework para la construcción de sistemas de comunicación totalmente configurables basado en texto utilizados para realizar conferencias virtuales o dictar clases a distancia. Talk (unix), Pirch, Virc, Ilus.

<sup>16</sup>NOTA: No se trata de lugares físicos sino de lugares virtuales.

<sup>17</sup>IRC: Internet Relay Chat.



Figura 8 Trillian Pro

### 4.3. Servicio: Mensajería Instantánea

Si bien el concepto de enviar mensajes a otras personas es algo viejo, el servicio de Mensajería Instantánea (IM - Instant Messages) es un servicio relativamente nuevo dentro de las comunidades de Internet.

Los mensajeros instantáneos son un conjunto de programas que utilizan el protocolo TCP/IP para enviar y recibir mensajes instantáneos con otros usuarios conectados a Internet, además saber cuando están disponibles para hablar. Cada uno de estos mensajeros permite enviar y recibir mensajes en tiempo real de otros usuarios usando los mismos software clientes.

#### Algunas características de la mensajería Instantánea

Los sistemas de mensajería brindan funciones donde algunas son comunes a todos o casi todos los clientes:

- Mostrar varios estados (Disponible, Disponible para hablar, Sin actividad, No disponible).
- Registrar y borrar usuarios de la lista de contactos.
- Agrupar los contactos (Familia, Trabajo, Facultad, Amigos).
- Se puede usar un avatar, es decir una imagen que nos identifique.
- Muchas veces se puede utilizar emoticons representen que diferentes estados de ánimo.
- Se pueden crear salas (grupos de charla), publicas o privadas.
- Enviar archivos.

**Implementaciones:** Existen en la actualidad una gran cantidad de clientes de IM. Podemos citar algunos tales como ICQ [54], Yahoo Messenger [77], MSN Messenger [59], AOL [37], PowNow [66], EyeBall [8]. Algunos ad-hoc como DYCE Instant Messenger [73].

Cada una de los clientes anteriores tiene su propio protocolo de comunicación e interacción entre el server y el cliente. Por ejemplo, el protocolo de comunicaciones utilizado por ICQ es conocido como OSCAR<sup>18</sup>, utilizado también por AIM. MSN por su parte tiene su propio protocolo. Es por ello que la interoperabilidad entre servicios de IM es, en la actualidad, imposible. En este momento se están trabajando en estándares para lograr la interoperabilidad de los servicios. Hay, hasta el momento, dos iniciativas de interoperabilidad. Un grupo respalda el protocolo, open source, Jabber [55] y otro estándar es IMPP<sup>19</sup>.

Existen también multi-mensajeros como el trillian (Figura 8) utilizado para conectarse a varias redes de mensajería (ICQ, AIM, MSN, etc.).

<sup>18</sup>OSCAR: Open System for CommunicAtion in ReAltime.

<sup>19</sup>IMPP: Instant Messaging and Presence Protocol)

Figura 8: Trillian Pro

#### 4.4. Servicio: Foros

Los foros son sistemas utilizados para el intercambio de noticias. Los mismos están centralizados en sitios Web que pueden ser navegados con un "Browser". Los usuarios pueden visitar este tipo particular de Web para leer los mensajes y dejar los suyos.

**Diferentes usos:** Se utiliza para el intercambio de opiniones de manera asincrónica con grupos de personas. Algunos foros informan por medio de mail que se agregó alguna opinión nueva.

**Implementaciones:** En general no hay implementaciones estándares, ni conocidas. Es común que cada portal tenga sus propios foros y que maneje su propio protocolo y formato para los mismos. Así por ejemplo hay foros en terra.com, yahoo.com, clarín.com. En la Figura 9 podemos ver el foro web del entorno de aprendizaje WebInfo [74] que es utilizado como soporte para las materias dictadas en la facultad de informática de la UNLP.

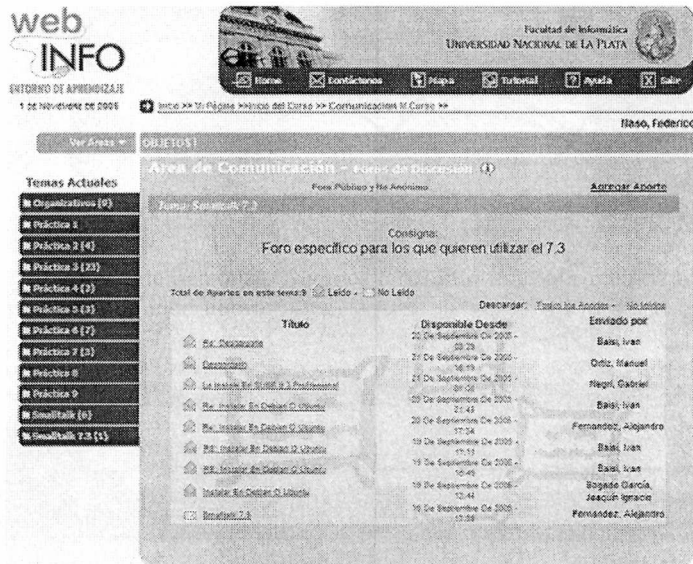


Figura 9: Foro de WebInfo

Aunque por el momento no hay estadísticas fiables al respecto, se estima que hay más de 55.000 foros Web en funcionamiento. Los foros Web ("Web Forums") son un servicio relativamente nuevo de la Web.

#### 4.5. Servicio: *WhiteBoards Compartidos*

Un WhiteBoard o pizarra compartida es un programa especial para trabajo en grupo que permite que varias personas dibujen a la vez en este pizarrón emulando los pizarrones reales. Aunque las personas no estén físicamente en un mismo lugar, pueden compartir el uso del whiteboard, de manera sincrónica desde cualquier punto conectado a través de Internet (Figura 10).

Los whiteboard se pueden utilizar para introducir texto, imágenes o realizar dibujos a mano alzada.

**Implementaciones:** EyeBall [8], ConversationBoard [5], GroupSketch [47], Wscrawl [76], GroupDraw [47], e incluso el NetMeeting [62] de Microsoft incluye uno de estos WhiteBoard.

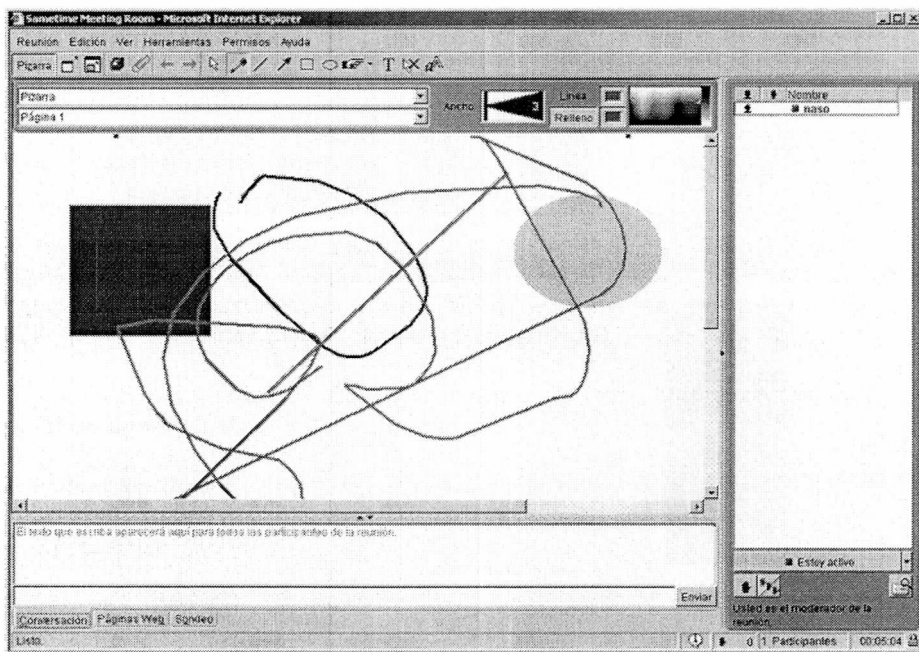


Figura 10: WhiteBoard de SameTime

#### 4.6. Servicio: *Presencia*

El servicio de presencia permite conocer información sobre las acciones que están llevando a cabo otros usuarios en el área de trabajo [50]. Posibilita saber, por medio de algún indicador, como por ejemplo un icono de color verde si el usuario está presente en la aplicación y un icono rojo en caso contrario, el estado de los usuarios en determinada aplicación [26, 51]. Para poder proveer presencia, el servicio de presencia debe obtener información sobre el estado de los usuarios [51].

El servicio de presencia mantiene la información sobre el estado del usuario (tanto del usuario observado como también del observador). Dicha información es enviada por los clientes al servicio de presencia, luego los mismos clientes pueden consultar información sobre la presencia de otros usuarios.

Los usuarios reciben notificación del servicio de presencia por cada cambio (presente, no presente, ocupado) en el estado del resto de los usuarios.

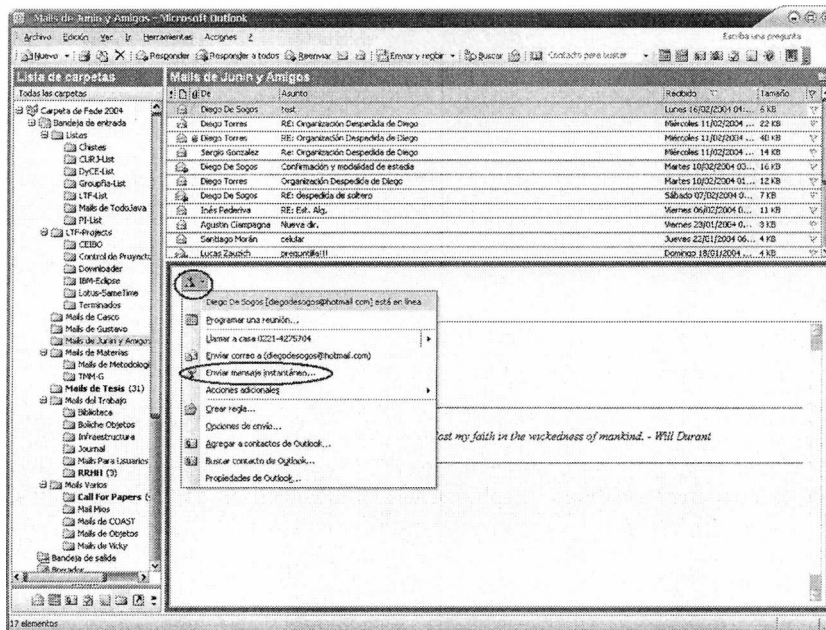


Figura 11: MS Outlook 2003

No es un servicio independiente, sino que siempre se lo utiliza con algún otro servicio como por ejemplo la IM (Capítulo 4.3). Está fuertemente ligada a la mensajería instantánea [58] y a las aplicaciones sincrónicas. Podemos ver presencia en aplicaciones como por ejemplo ICQ, MSN, AOL, etc.

Otro ejemplo de presencia es el caso del MSN junto al MS Outlook, al recibir

un mail, el Outlook<sup>20</sup> nos dejan saber si el usuario que envió el mail esta online o no. Por supuesto para ello el usuario debe ser miembro de la comunidad de usuarios de MSN y tener una cuenta en dicha comunidad. En (Figura 25) podemos ver como el Outlook 2003 informa si el usuario que envió el mail se encuentra o no online.

**Implementaciones:** las implementaciones son múltiples y muy variadas, podemos encontrar implementaciones en clientes de IM como por ejemplo MSN, ICQ, AOL, yahooMessenger, etc. o en clientes de mail como por ejemplo, Outlook, Outlook Express, NetScape, etc.

---

<sup>20</sup>A partir de la versión XP y del 5.0 en el Outlook Express

#### 4.7. Servicio: CoWeb

Un CoWeb es un ambiente de colaboración compuesta por páginas Web en formato hipertexto; cualquiera puede crear o modificar las páginas que lo componen; pueden ser visitadas y editadas por cualquier persona (aunque en algunos se exige el registro como usuario) en cualquier momento; las páginas son "linkadas" por sus nombres.

Todos pueden aportar algo de escritura en la Web (de a una página por vez, es decir que la página es editada de a un usuario por vez) y entre todos armar el sitio. Se utiliza una notación sencilla para dar formato, crear links, etc. Cuando alguien edita una página, sus cambios aparecen inmediatamente en la web, sin pasar por ningún tipo de revisión previa. Se debe tener cuidado en no "pisar" lo que otros escriben ya que todos pueden escribir. Mientras que al principio esto puede parecer peligroso, es también muy útil y de gran uso.

👉 *Ward Cunningham es el inventor de este tipo de sitios web, las llevó a cabo mediante la implementación de la WikiWikiWeb.*

El concepto de CoWeb fue concebido por Ward Cunningham [34] él mismo implementó el primer CoWeb al que llamo wikis<sup>21</sup>.

Un CoWeb debe respetar los siguientes tópicos:

- **Abierto:** Si un usuario desea editar la página, y tiene los permisos correspondientes puede hacerlo.
- **Incremental:** Las páginas pueden citar otras páginas, incluyendo algunas que todavía no estén escritas.
- **Orgánicas:** La estructura organizacional del sitio está abierta a modificaciones y evoluciones al igual que el texto de las mismas.
- **Mundano:** Un pequeño número de convenciones en el texto proveen de acceso a muchas herramientas útiles de formateo de texto.
- **Universal:** Los mecanismos de edición y de organización son el mismo permitiendo a un editor ser al mismo tiempo editor y organizador.
- **Unificado:** Los nombres de las páginas son simples, y no agregan ninguna información de contexto, que requiera una interpretación compleja del nombre.
- **Observable:** La actividad dentro del sitio puede ser observada y revisada por cualquier visitante del sitio que posea los permisos para hacerlo.
- **Convergente:** El contenido duplicado puede ser fácilmente eliminado mediante la búsqueda de contenido similar en el sitio.
- **Preciso:** Las páginas serán tituladas con suficiente precisión para evitar el solapamiento<sup>22</sup> de nombres.

---

<sup>21</sup>**WIKI:** wiki significa "rápido" en hawaiano.

<sup>22</sup>**Solapamiento:** paginas con el mismo nombre.

El objetivo de un CoWeb es democratizar la creación y el mantenimiento de las páginas, al eliminar el "síndrome de un solo webmaster o administrador".

El gran potencial del CoWeb radica en que no es necesario aprender a utilizar complicadas etiquetas para escribir de forma sencilla documentos y establecer enlaces en el sitio web.

Las sintaxis difieren de una implementación a otra. Los CoWeb definen un conjunto de reglas y sintaxis y a partir de ahí cada implementación tiene su propia sintaxis. Algunos CoWeb soportan HTML, pero la idea no es que el usuario sepa los tags de HTML sino que solo recuerde una sintaxis simple y flexible.

**Diferentes usos:** El uso que se les puede dar a los CoWeb es muy variado. Algunas personas lo utilizan para armar un sites de manera colaborativa poniendo en ellos información que todo el mundo aporta. Se lo puede utilizar también como portal de proyectos en donde no se puede dedicar gran cantidad de tiempo en la actualización del portal, sino que se necesita rapidez, funcionalidad, seguridad, etc. Se lo utiliza también como block de notas rápidas en donde los usuarios pueden ver, modificar y borrar.

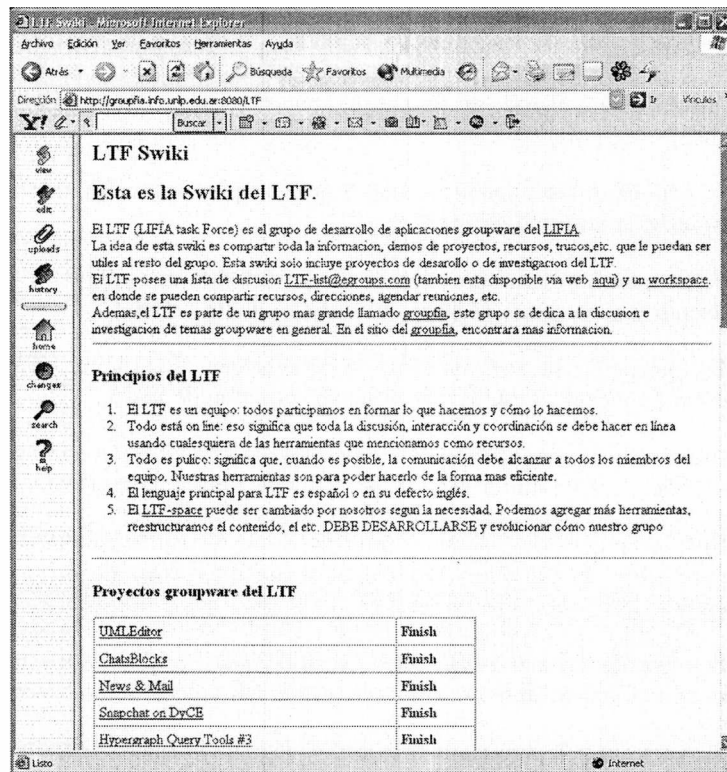


Figura 12: Swiki

**Implementaciones:** Las wiki son implementaciones funcionales de CoWeb. Es decir, un CoWeb es solo una descripción mientras, que como por ejemplo, la



Swiki (Figura 12) [71], ChikiWiki, JSPWiki son productos finales que implementan dicha terminología de CoWeb. Los CoWeb están siendo cada vez más usados a medida que se conoce de su existencia [21].

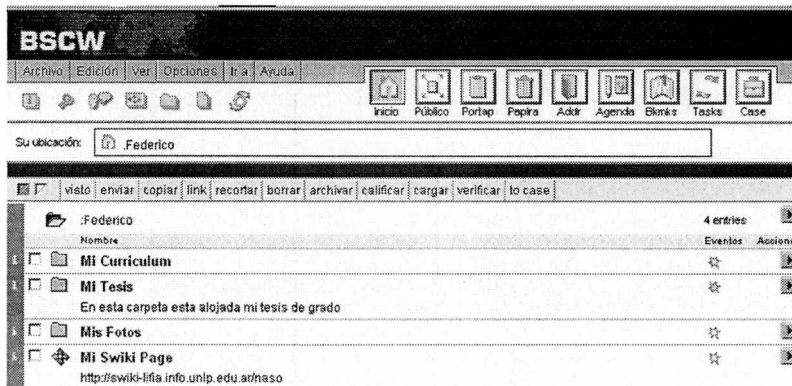
**Podemos encontrar implementaciones de coWeb en:** ActiveServerPages, CeeLanguage (C), CeePlusPlus (C++), CsharpLanguage (C#), Cold Fusion, CommonLisp, HtagLanguage, JavaLanguage (Java), PerlLanguage (PL), PhpHypertextPreprocessor (PHP), Prolog Language, PythonLanguage, SmalltalkLanguage.

#### 4.8. Servicio: *Sistemas de Manejo de contenido (CMS)*

Un *CMS* (*Content Management System* o *Sistemas de Manejo de contenido*) es un *espacio de trabajo* compartido, una aplicación general que puede ser usada para, por ejemplo, almacenar documentos (u otros objetos, como por ejemplo, link, reuniones, etc.) que traten sobre un proyecto o sobre un grupo de trabajo en particular. Se puede acceder a un espacio de trabajo, navegar a través de las carpetas y obtener objetos de igual manera que en las páginas WWW ordinarias.

Los CMS poseen varias funcionalidades que los potencian, como por ejemplo: versionar cada uno de los documentos que se van subiendo al CMS, notificar sucesos ocurridos (Ej.: creación, lectura o modificación de objetos) por medio de email o algún otro servicio de comunicación, control de acceso y privacidad (solo los usuarios autorizados pueden entrar al CMS y manipular los objetos almacenados)

**Diferentes usos:** Se utiliza para compartir documentos en un espacio de disco o de trabajo con un grupo de personas a través de distintas plataformas (Windows, Macintosh o Unix). Por cada documento u objeto subido al repositorio mantiene una versión de ese objeto, es decir, versiona el documento y su acceso es mediante un browser.



BSCW © 1995-2004 FIT and OrbiTeam

Figura 13: Contenido en un BSCW

**Implementaciones:** Una de las implementaciones más completas es el BSCW (Figura 13) [3], el mismo permite compartir documentos, direcciones URL<sup>23</sup>, reuniones, etc. Otra de las implementaciones es la tikiwiki [32]. La tikiwiki además de ser un repositorio de documentos, tiene su base en las wikis (CoWeb, sección 4.1.7). Jakarta Slide [JakartaSlide], Microsoft posee el Microsoft Content Management Server y Microsoft Sharepoint Portal Server [17].

<sup>23</sup>URL: Uniform Resource Locator.

#### 4.9. Servicio: *Workflow*

Los Workflows son sistemas que ayudan a administrar y automatizar la secuencia de acciones y pasos usados en procesos de negocios. Una aplicación de Workflow automatiza la secuencia de acciones, actividades y tareas usadas para ejecutar los procesos, incluyendo monitoreo en cada instante del proceso. Un workflow puede ser descripto como el flujo y control en un proceso de negocio.

Los workflows son parte de las aplicaciones utilizadas en el área de CSCW, más específicamente en CSCWP<sup>24</sup>.

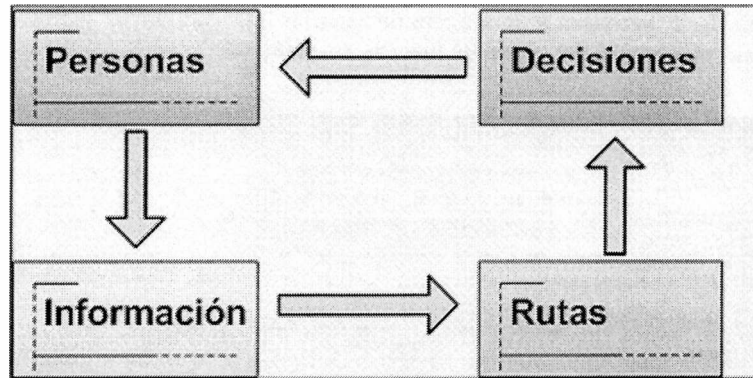


Figura 14: Elementos de un Workflow

Para poder identificar cada elemento dentro de cada workflow se puede utilizar el modelo de componentes de proceso de negocio (BPCM<sup>25</sup>). Este modelo plantea cuatro elementos clave que forman parte de los componentes de un proceso de negocios y por lo tanto de un workflow (Figura 14).

**Diferentes usos:** Los workflows se utilizan cuando se desea automatizar una secuencia de pasos (de la vida real) a seguir para realizar una tarea. Se utiliza cuando se quiere controlar una cierta cantidad de pasos en donde se debe utilizar algún método o herramienta que ayude a realizar esta tarea, debido a la gran cantidad de ítems o pasos que se deben tener en cuenta.

**Implementaciones:** Staffware, InConcert, FlowMark, COSMOS, AMIGO, GRACE, DOMINO, Micro Workflow [16], etc.

<sup>24</sup>CSCWP: Computer Supported Colaborative Work Processing.

<sup>25</sup>BPCM: Bussines Process Component Model.

#### 4.10. Servicio: *Calendarios compartidos*

Los calendarios compartidos son uno de los sistemas de grupo que proveen mayores beneficios, para los grupos de trabajo, al ser usados con computadoras.

Éste tipo de calendarios buscan tiempo libre entre los calendarios de las personas del grupo para establecer reuniones o tareas a realizar. Los participantes confirman o declinan su asistencia vía mail o por algún otro medio de comunicación. De ésta manera hacen más eficiente la organización del tiempo del grupo.

Se utilizan para poder programar citas o reuniones tanto individuales como de grupo. Es decir, permiten a un número de usuarios que sincronicen sus tareas y reuniones con el resto del grupo de manera automática.

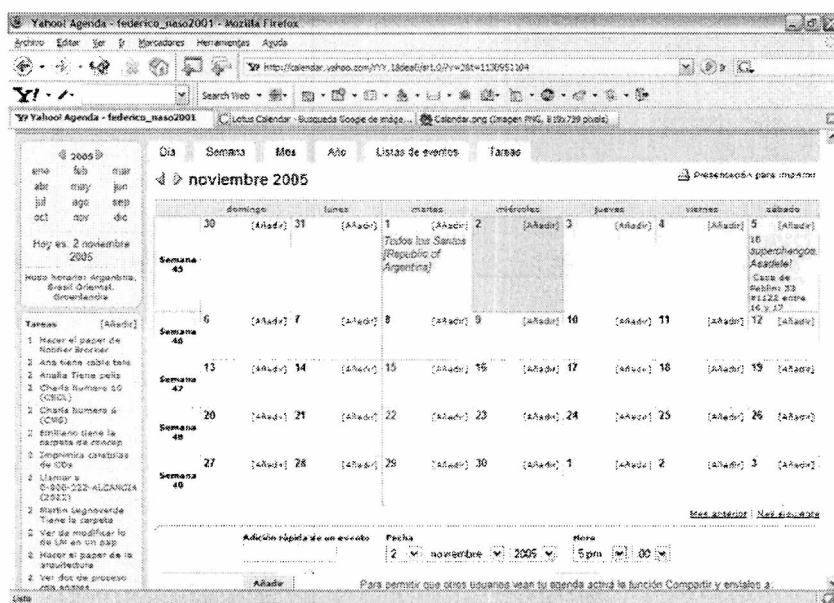


Figura 15: Yahoo Calendar

Si bien existen autores [48] que sostienen que este tipo de aplicaciones colaborativas no son adecuadas para ser usadas en grupos, si se tienen recaudos y pautas a seguir son de muy útil ayuda.

Uno de los problemas que tiene este tipo de aplicaciones es el uso de un formato de fecha estándar. Tengamos en cuenta que los integrantes del grupo pueden estar en diferentes partes del mundo, cada uno con su formato de fecha y hora específico. Para ello existen formatos como el CIP<sup>26</sup> o el CAP<sup>27</sup> que resuelven éste problema.

<sup>26</sup>CIP: Calendar Interoperability Protocol.

<sup>27</sup>CAP: Calendaring Access Protocol.

**Implementaciones:** Microsoft Schedule+, Lotus Calendar y Novell Groupwise. Estas implementaciones trabajan en función de buscar el tiempo libre entre los usuarios. Se basan en una arquitectura mediante la cual, vía mail, se invita, se confirma o se declina la invitación a una reunión. Existen además algunas otras implementaciones en donde se puede acceder al calendario vía Web como por ejemplo: yahooCalendar (Figura 15), MyTime.com, when.com.

#### 4.11. Servicio: Videoconferencias

El servicio de Videoconferencia permite la comunicación entre grupos de personas geográficamente distantes mediante la utilización de sistemas capaces de conjugar audio y video en un mismo paquete ubicados en cada extremo. Este servicio provee todas las opciones de intercambio de información de forma bidireccional, interactiva y en tiempo real, que son posibles en una reunión cara a cara, por lo que se convierte en una alternativa para optimizar recursos ya que los participantes no se tienen que trasladar de sus lugares.

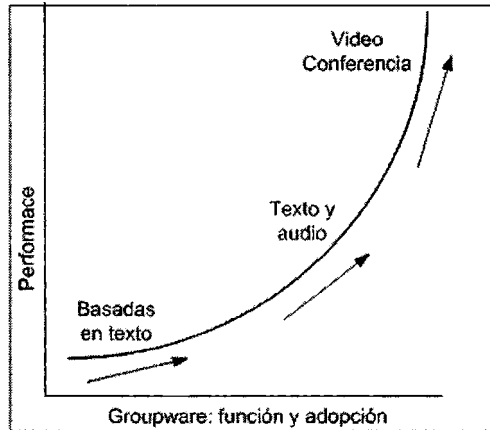


Figura 16: Evolución de la video conferencia

Se utiliza para poder comunicarse con otra persona, o grupo de personas, en forma *sincrónica* y de modo que las personas se pueden ver entre si por medio de una cámara. Simula la interacción cara a cara. Hay que tener en cuenta que este tipo de aplicaciones están muy condicionadas al ancho de banda y a las condiciones de transmisión de la red.; ya que consumen muchos recursos.

Clasificaciones	
<b>Rooms based</b>	Habitaciones con grandes pantallas, micrófonos etc. (líneas telefónicas)
<b>Roll-about</b>	Similar al room pero con equipos móviles.
<b>Desktop videoconferencing</b>	Muchas de estas aplicaciones están vinculadas a application sharing (netmeeting) y/o computer conferencing system (acompañando un pizarron compartido por ej.)

Las conferencias por computadora fueron evolucionando con el correr de los años. Se fueron mejorando sus funciones tecnológicas y su adopción por parte de los usuarios (Figura 16).

**Implementaciones:** EyeBall, Netmeeting, Netscape conference, CRUISER, etc. Tanto NetScape conference como MS NetMeeting ofrecen conferencia con



Figura 17: Rooms Based

sonido, tableros electrónicos, texto y transferencia de archivos. Conference solo soporta conferencias punto a punto entre dos participantes. Mientras que NetMeeting permite que tres o mas usuarios trabajen en conjunto aunque solo dos de ellos puedan tomar parte en la conferencia de sonido y video a la vez. Además NetMeeting soporta otra utilidad que es el poder compartir aplicaciones, es decir, permitir que otro usuario tome posesión de nuestras aplicaciones.

#### 4.12. Servicio: Herramientas de Votación

Las herramientas de votación o polls son herramientas en donde uno de los integrantes abre una votación para elegir entre varias opciones. Los integrantes del grupo reciben una invitación mediante mail para elegir alguna (algunas herramientas tienen la opción de dejar elegir por más de una opción) de las opciones establecidas. Los polls tienen un tiempo de duración y una vez finalizado el plazo se recibe un mail indicando cual de las opciones fue la más votada.

New Meeting | About | Quick Start | Tour | FAQ | Help | Contact Us

### MeetOMatic

## 1. Select possible dates, press 'Go!'

Next: 2. Email participants; 3. Monitor progress

The following fields are required (explain):

Meeting name:  My name:  My email:

Remember my name and email address

Tip: Use [Select All Weekdays](#) or [Unselect All Weekdays](#) for speed

November 2005							December 2005						
Mon	Tue	Wed	Thu	Fri	Sat	Sun	Mon	Tue	Wed	Thu	Fri	Sat	Sun
	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>				<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

**GO!**

[Add 2 months](#)

[View AM and PM](#)

Meet-O-Matic: The World's Simplest Meeting Scheduler. Version 2.5. Copyright © [meetomatic.com](#), 2003-2005. All Rights Reserved.

Figura 18: Creación de un Poll en MeetOMatic

**Implementaciones:** Una de las implementaciones es polls4free [25], ésta permite abrir un poll de votación en 5 pasos.

Otra de las más usadas y conocidas es el poll de yahooGroups. El poll de yahoo permite ingresar el tópico de la votación y varias (al menos dos) opciones a elegir. Además da la posibilidad de configurar la votación para que se puedan elegir una o mas opciones, que se vean o no los resultados parciales y que el cierre de la votación sea manual o por tiempo.

MeetOMatic (Figura 18) es una implementación donde fácilmente se puede crear un poll e invitar a un grupo de personas a que elijan la mejor opción de las que hay en el poll que ellos crean conveniente.



### 4.13. Servicio: Sistema de control de versiones (SCV)

Los sistemas de versionamiento son administradores de código fuente u objetos, que permiten compartir el producto del trabajo dentro de una red de un modo protegido y seguro. Cada vez que un usuario modifica un objeto, el sistema procede a realizar una nueva versión del mismo.

**Diferentes usos:** Se utiliza en los grupos de trabajo, pueden ser usados de manera individual también, para llevar un control de las versiones de sus archivos. Permiten mantener versiones antiguas de archivos (generalmente códigos fuente), mantiene una bitácora de quién, qué y por qué ocurrieron cambios.

**Implementaciones:** CVS, Visual Source Safe, ENVY, StarTeam, Arch, PVCS, RCS, Clear Case, ADELE

La mayoría de los IDE<sup>28</sup> de desarrollo tienen la opción de poder trabajar con alguno de estos sistemas o protocolos de versionamiento. El más usado por los IDEs es el CVS. Existen clientes para trabajar con CVS de todo tipo, tenemos por ejemplo el Tortoise que se integra al Explorer de Windows, el WinCVS que es un cliente independiente o el SmartCVS desarrollado en JAVA. Además existen clientes para el web como el WebCVS.

Herramientas de desarrollo tales como el Eclipse y Websphere Studio Application Developer cuentan con un cliente en la misma IDE que permiten trabajar con el CVS en una forma totalmente intuitiva y que simplifica muchísimo el manejo de esta herramienta de manejo de versiones.

CVS: CVS<sup>29</sup> es un software Open-Source que permite administrar los cambios realizados en el código fuente de cualquier archivo ya sea en: C, C++, Java, HTML, XML e inclusive hasta diferentes versiones de archivos ejecutables o binarios.

CVS también es capaz de producir versiones de archivo a cierto punto en el tiempo, esto es, si la versión más reciente de un archivo es la 1.3, CVS (Figura 19) puede producir la 1.4 o 2.4 del depósito, también es capaz de subjerarquizar archivos y otras funcionalidades utilizadas en proyectos que requieren control de versiones.

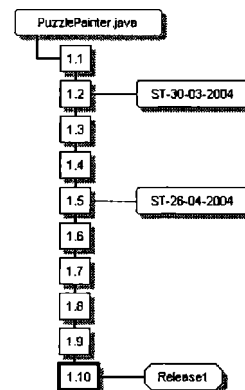


Figura 19 Versiones del CVS

*CVS nace en 1989, de la mano de Brian Berliner a partir de unos "script" publicados en Usenet (puerto 119). Desde entonces, se han sucedido multitud de cambios realizados por un gran número de empresas. El sistema de versionamiento CVS es el sistema de versionamiento más usado en los proyectos OpenSource. Cabe destacar que el CVS es OS en si mismo.*

<sup>28</sup>IDE: Integrated Development Environment.

<sup>29</sup>CVS: Concurrent Versioning System.

Figura 19: Tortoise CVS

#### 4.14. Servicio: Libreta de contactos compartida

Una agenda es una aplicación en donde las personas mantienen una lista de contactos e información de otras. Es como la guía de teléfonos personalizada. Todas las aplicaciones de correo hoy en día tienen una de éstas agendas. Son muy útiles a la hora de mantener la información, datos, fechas de cumpleaños, direcciones de mail de amigos, empresas y demás.

Se utiliza para compartir información o datos de personas entre el resto de los integrantes del grupo. Algunas permiten armar grupos de contactos y compartir esos grupos con el resto de los usuarios.

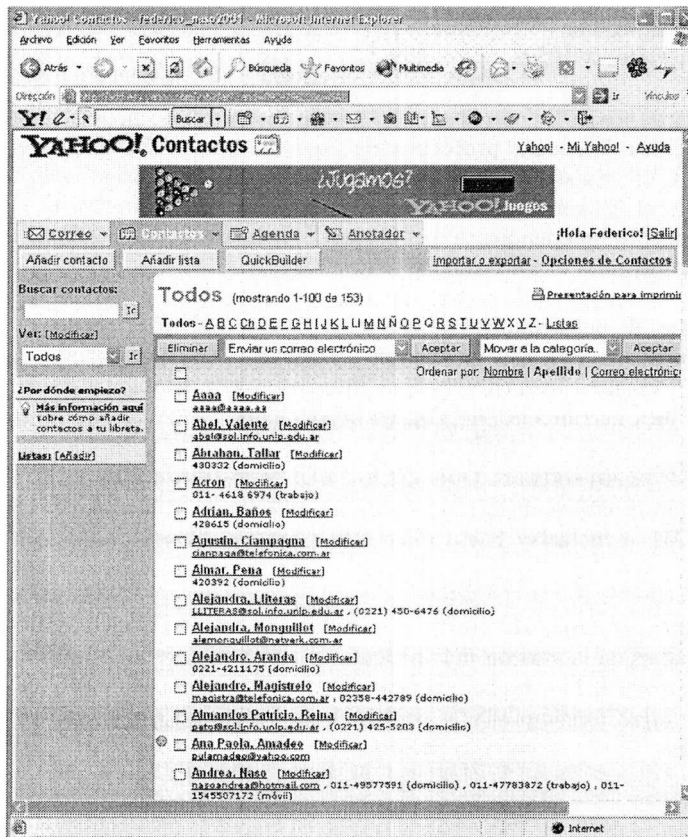


Figura 20: Agenda de contactos de Yahoo

**Implementaciones:** MyTime.com ([www.mytime.com](http://www.mytime.com)), ScheduleOnline, Visto Briefcase, Yahoo Contactos (Figura 20) (<http://address.yahoo.com/>)

#### 4.15. Servicio: WebLogs

Básicamente, un weblog<sup>30</sup> (también llamado blog) es un espacio personal de escritura en Internet. Se puede pensar como *un diario personal*, un site que una persona usa para escribir periódicamente, en el que toda la escritura y el estilo se maneja via Web. Un weblog está diseñado para que, como en un diario, cada artículo (post) tenga fecha de publicación, de forma tal que el escritor (weblogger) y los lectores pueden seguir un camino de todo lo publicado y archivado.

👉 El concepto de weblog existe desde los principios de Internet. Tim Berners Lee (uno de los creadores de la web), tenía una página en la que incluía los nuevos sites que se iban creando y agregaba un comentario sobre cada uno de ellos.

Un weblog suele tener al menos las siguientes secciones diferenciadas:

- **Archivo:** En el archivo se guardan todas las entradas publicadas (también llamadas posts, notas, anotaciones.) de tal modo que todo el mundo tenga acceso a ellas.
- **Información:** Espacio para mostrar alguna información tanto personal como sobre el sitio en sí.
- **Enlaces:** Lista de enlaces a los blogs que se suelen leer.
- **Contacto:** Es una forma de contacto ente los visitantes y el responsable del weblog mediante un mail, mediante un programa de mensajería o con un formulario

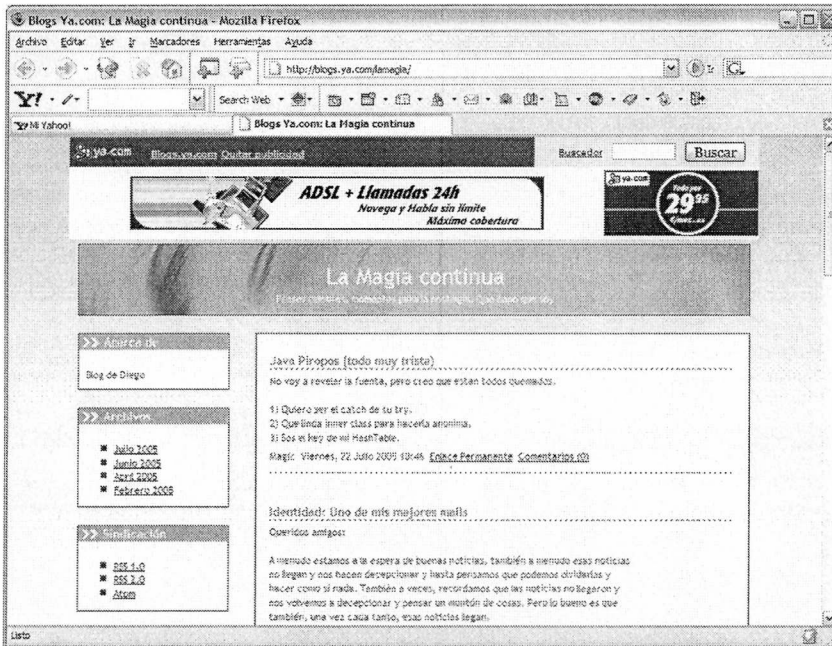


Figura 21: weblog de un usuario

<sup>30</sup>Log: un log es un registro de las actividades que se realizan en determinado tiempo.

**Diferentes usos:** En general, los bloggers forman verdaderas comunidades online con aquellos con quienes comparten intereses. Estas comunidades pueden tener diferentes formas. Existen grupos de trabajo colaborativo que utilizan un weblog como herramienta para comentar, opinar y escribir noticias relativas a su tarea.

**Implementaciones:** Las más utilizadas son Blogger, Movable Type, Antville y Pitas. Cualquiera de ellas garantizan la creación y publicación de un weblog, una operación que es tan sencilla como elegir un nombre de usuario y un "*password*", escribir un artículo y publicarlo. La complejidad de las páginas depende de su autor. Arrozabanda [2] Weblog grupal relacionado con el software libre, las nuevas tecnologías y el desarrollo web. Pequeños Enigmas [23] Weblog dedicado a pequeños enigmas y acertijos. Edmz [7] PHP, XML, IT, MX y otros

#### 4.16. Servicio: *Sistemas de co-autoría (Co-Authoring Systems)*

La creación de documentos<sup>31</sup> es una de las tareas más frecuentes en las cuales se usan las computadoras. En la actualidad los editores y procesadores de texto (los más usados) proveen funcionalidad para co-editar documentos, revisar, etc. Muchos de los documentos creados son llevados a cabo por más de un autor. En el campo de la ciencia más del 65 % de las publicaciones científicas son realizadas por dos o más autores [43]. En el caso de los procesadores de texto la idea de co-autoría es solo enviar por mail o algún otro medio de comunicación el documento para que el otro autor lo revise y pongan sus marcas de comentarios.

Pero existen también otro tipo de sistemas de co-autoría en donde lo que se comparte es el documento en sí. Estos sistemas de co-autoría están explícitamente diseñados para dar soporte simultáneamente a varios usuarios. Es decir, cada usuario puede participar de la creación de forma segura y transparente dado que el sistema se encarga de las tareas de coordinar y sincronizar las tareas necesarias para que todos los clientes trabajen sobre el mismo documento.

Se los utiliza para la creación, manipulación, y edición de documentos en los cuales participan varios autores en su edición. Dichos autores pueden trabajar sincrónicamente o bien asincrónicamente sobre el mismo documento.

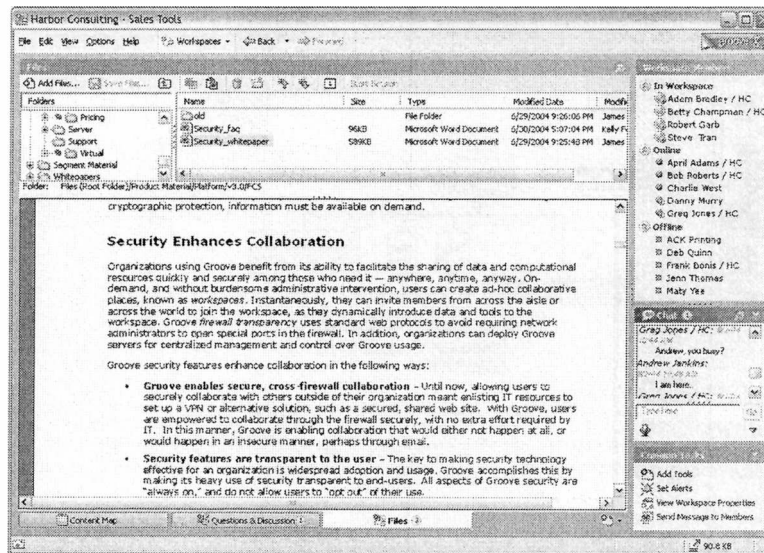


Figura 22: Edición de un documento usando Groove

#### Implementaciones:

- **QUILT**: soporta documentos compartidos a través de una base de datos en la cual se almacenan textos y anotaciones sobre el documento. Las

<sup>31</sup>Se hace referencia a *documento* como el resultado de utilizar un editor. No solo un documento de texto, sino cualquier otro que sea llevado a cabo mediante una aplicación específica.

anotaciones pueden ser sugerencias para revisar y comentarios de voz o texto.

Permite que múltiples usuarios editen el documento de forma concurrente. Utiliza el sistema de transacciones de la base de datos para resolver conflictos de inconsistencia en la edición de los documentos entre los usuarios. Como así también para bloquear el uso y/o edición de documentos.

- **Groove**: es una herramienta de edición que permite que usuarios que se encuentran distantes trabajan sobre el mismo documento (Figura 22). Es básicamente un outliner similar a los que se encuentran en cualquier editor de texto. La diferencia radica en que Grove es un editor de grupo en tiempo real. Cada uno de los usuarios puede seleccionar una parte del documento para trabajar. Así cada una puede trabajar en una sección de documento concurrentemente con el resto de los usuarios.
- **CoMedia**: es una aplicación desarrollada para soportar la manipulación de documentos de hypermedia utilizando conferencias y coordinación basada en roles sociales [53]. Soporta texto, audio, video gráficos 2D e imágenes. Soporta bloqueo de los documentos para mantener la consistencia de los mismos.

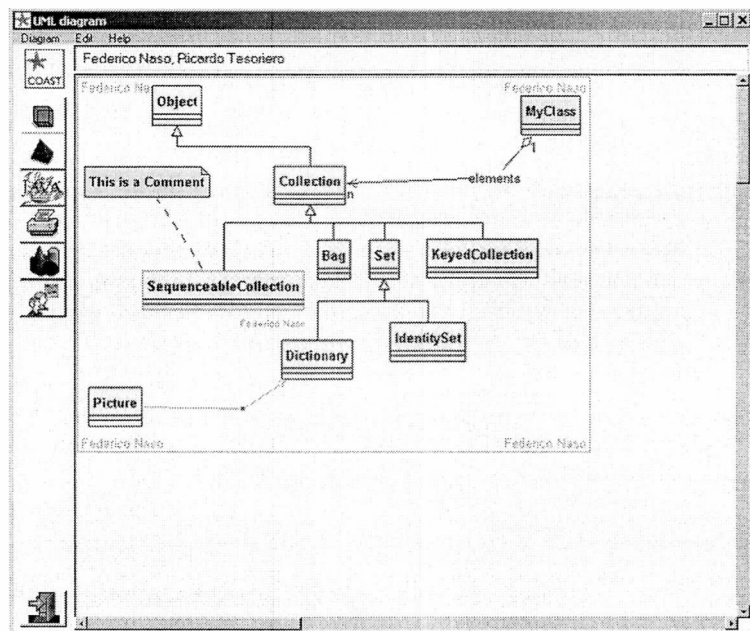


Figura 23: UMLEditor

- **UMLEditor**<sup>32</sup>: es un editor de diagramas UML colaborativo (Figura 23) en donde el documento compartido es el diagrama de UML. Los usua-

<sup>32</sup>UMLEditor: El UMLEditor fue desarrollado por el LIFIA de la Universidad Nacional

rios pueden crear y editar el diagrama de manera concurrente. El nivel de concurrencia son los objetos (una clase, interfase, relación, etc.) gráficos que componen el diagrama. El UMLEditor utiliza un framework de cooperación llamado COAST [4]. El mismo fue desarrollado por el Instituto Alemán de investigación GMD. Dentro de este tipo de servicio se podrían agrupar también los editores de hypermedia colaborativos como se describe en 4.18. Pero los servicios de hypermedia son más específicos que los de co-autoría. Es decir, son un subconjunto de los mismos.

---

de La Plata en conjunto con el Instituto de investigación Alemán GMD en el año 2000. Una versión del UMLEditor se distribuye libremente con el VisualWorks 7.

# Capítulo V

Catálogo de Patrones de Integración



## 5. Catálogo de Patrones de integración

El siguiente capítulo es uno de los más importantes y es donde está documentado el catálogo de patrones de integración de servicios groupware propuesto en el capítulo uno. El mismo consta de 6 patrones documentados siguiendo una estructura similar a la que proponen los patrones de diseño. Dicha estructura fue modificada, extendida para que la documentación de la integración de servicios sea más clara y aplicable al contexto de integración de servicios groupware.

## 5.1. Estructura de los patrones

La estructura elegida para documentar los patrones de integración de servicios es la misma que se utiliza en los patrones de diseño [45] con algunas extensiones que ayudan a entender mejor la integración.

A continuación se da una descripción de cada uno de los ítems que describen al patrón:

- **Nombre:** Aquí se detalla en nombre con el que se conoce el patrón. Es un indicio del problema que el patrón trata de solucionar.
- **Intención:** Un párrafo corto en donde se respondan las siguientes pregunta sobre el patrón:
  - ¿Que es lo que el patrón debe hacer?
  - ¿Cual es el análisis de su intención?
  - ¿Que solución particular trata aportar?
- **También conocido como:** Otros nombres con los cuales pueda ser conocido el patrón.
- **Problema:** Una descripción que ilustre cuando aplicar el patrón. Explica el problema y el contexto del mismo. Algunas veces el problema incluye una lista de condiciones que deben satisfacerse antes de aplicar el patrón como solución
- **Escenario:** Un escenario que ilustre el problema que se desea resolver y como el patrón resuelve dicho problema. Este escenario ayuda a entender de una manera más abstracta al patrón en sí.
- **Fuerzas:** Que cosas se deben tener en cuenta. (Objetivos y restricciones).
- **Solución:** Descripción de los elementos que compondrán a la solución del problema en forma general. Provee las pautas de manera abstracta y debe adaptarse a cada problema en particular.
- **Participantes:** Un breve detalle de los servicios, componentes, protocolos, etc que se deben integrar en el patrón.
- **Colaboración entre los servicios:** Descripción de las relaciones entre los servicios integrados, sus respectivas responsabilidades y como interactúan entre si.
- **Detalles de Implementación:** Se pueden documentar diagramas o diseños de la integración. Esto ayuda a que la implementación sea más clara y más fácil de entender.
- **Aplicación:** Situaciones en donde el patrón puede ser aplicado. Es útil en el caso de que se tenga que reconocer las situaciones donde aplicarlo.
- **Consecuencias:** Los resultados que se obtienen luego de aplicar el patrón. Beneficios, riesgos, perjuicios que puede ocasionar su uso.

- **Ejemplos de uso:** Ejemplos de productos y sistemas conocidos que recuerde rápidamente el lector, facilite su comprensión y valide el patrón con una solución existente y probada.
- **Patrones Relacionados:** Patrones que pueden funcionar como alternativa o de forma complementaria.

## 5.2. Resumen de los patrones

**Rendezvous:** El groupware asincrónico no provee indicadores de presencia y no provee facilidades para iniciar colaboración sincrónica. En casos donde los usuarios se encuentran online simultáneamente, las características mencionadas hacen que se utilice de manera ineficiente el tiempo y esfuerzo empleados en resolver una tarea. La solución documentada en este patrón extiende el mecanismo asincrónico de groupware con un mecanismo de presencia de usuario, el cual informa la disponibilidad online al resto de los usuarios que estén usando groupware sincrónico. De ésta manera los usuarios que se encuentran utilizando el servicio de groupware asincrónico no pierden oportunidades de interactuar con otros usuarios a través de un servicio sincrónico.

**Notifier Broker:** La carencia de un mecanismo que informe a grupos de trabajo, los cuales se encuentran distribuidos geográficamente, sobre la disponibilidad de material e información de importancia, conlleva a pérdida de tiempo y a un uso ineficiente de la información disponible o, de cambios en la misma ya que se debe informar por otros medios los cuales no son parte del sistema; con lo cual la propuesta de este patrón es la integración de un servicio de comunicación que, mediante eventos programados, informe sobre la disponibilidad de nuevo, o de cambios, de material publicado a usuarios registrados para que puedan hacer uso del mismo de manera inmediata.

**Gossip:** Un workspace de trabajo colaborativo es creado y modificado dinámicamente, por lo tanto es de esperar que más de un usuario esté trabajando en el mismo momento. Dichos usuarios que se encuentran trabajando en el mismo espacio de trabajo colaborativo, no se percatan de que hay otros trabajando al mismo tiempo. Es por ello que es muy útil poseer alguno de los mecanismos de presencia de usuario para informar al resto sobre otros usuarios que se encuentran trabajando en el mismo espacio de trabajo.

**Chat Channel:** La carencia de servicios de comunicación en las aplicaciones colaborativas, conducen a la imposibilidad de comunicarse y sincronizarse con el resto de los usuarios que se encuentran trabajando dentro del mismo espacio de forma simultánea; con lo cual la solución es integrar una aplicación colaborativa con un servicio de comunicación sincrónica por texto (Chat) para que de éste modo los usuarios puedan comunicarse en tiempo real y así poder discutir su trabajo y sincronizar el mismo.

**Virtual Conference:** Imposibilidad de llevar a cabo reuniones o conferencias entre personas o grupos de personas, debido a que las mismas se encuentran geográficamente distantes. Tales conferencias son precizadas porque no requieren que los participantes se encuentren en el mismo lugar físico.

La posibilidad de interactuar en una conferencia remota reduce el tiempo y esfuerzo que se requiere para resolver un problema mediante otra vía. Para ello se utiliza la integración de los servicios de video, audio y de Chat como un paquete de comunicación integral. Se logra una comunicación, compuesta por varias vías, entre personas o grupos de personas que se encuentran geográficamente distantes.

**Mail Manager:** La carencia de gestores de mail integrados con agendas de contactos de usuarios (Nombre, dirección, dirección de e-mail, etc.) hace que el envío de e-mail sea improductivo, tedioso y lento. Dado que el usuario debe estar buscando la dirección de e-mail en una agenda independiente y luego copiando dicha dirección en el cliente de e-mail.

La solución a este problema propuesta por este patrón sería integrar el servicio de mail para la comunicación (asincrónica) con una agenda que gestione los contactos en donde se tienen los datos de las personas, incluyendo su dirección de mail la cual es utilizada por el cliente de mail para enviar los mensajes a dicha dirección.

### 5.3. Patrones

A continuación se detallan los patrones correspondientes al catálogo.

#### 5.3.1. Rendezvous

☞ Patrón aceptado y publicado en: *The 11th Conference on Pattern Languages of Programs 2004 September 8th - 12th, 2004 Monticello, Illinois, USA.* [60]

- **Nombre:** *Rendezvous*<sup>33</sup>.
- **Intención:** Informar a usuarios de un sistema de groupware *asincrónico* sobre la posibilidad de colaborar con otras herramientas de manera *sincrónica* (groupware sincrónico).
- **También conocido como:** Lugar de encuentro, Reunión, Confluencia.
- **Problema:** La carencia de indicadores de presencia y la imposibilidad de iniciar colaboración sincrónica que son inherentes a los servicios de groupware asincrónico conducen, en casos donde los usuarios se encuentran simultáneamente online, a un desaprovechamiento del tiempo y esfuerzo que se requiere para resolver una tarea compartida.
- **Escenario:** Imaginemos un grupo de desarrollo de software open source donde programadores, testers, diseñadores y el líder del proyecto se encuentran trabajando distribuidos geográficamente.  
El grupo está en constante comunicación vía e-mail. Dos horas antes de terminar el día, el equipo de programadores encuentra un problema que les impide continuar su trabajo. Los programadores envían una descripción del problema vía e-mail al líder de diseño, único responsable de proveer la respuesta. Conscientes de la naturaleza *asincrónica* del e-mail, posponen las actividades pendientes para el día siguiente. En el mismo momento que los programadores envían el e-mail, el líder de diseño se encuentra casualmente trabajando en su estación, sin embargo no presta atención al e-mail recibido, el cual se propone tratar mas tarde.  
En este escenario, un problema que se podría haber solucionado de forma inmediata, se pospone hasta el próximo día y se pierden preciadas horas de trabajo.
- **Fuerzas:**
  - La colaboración en forma asincrónica espreciada porque ofrece a los participantes tiempo para elaborar sus contribuciones y la posibilidad de contribuir cuando les sea más oportuno.
  - En ciertos casos la posibilidad de interactuar de forma sincrónica reduce el esfuerzo y el tiempo que se requiere para resolver una tarea compartida.
  - La colaboración sincrónica requiere que los participantes se encuentren simultáneamente online, ya sea casualmente o de forma intencional.
  - La integración de servicios de colaboración, toma lo mejor de cada uno y de esta forma potencian la aplicación donde ellos estén integrados.

---

<sup>33</sup>**Rendezvous:** Este nombre se le utiliza para mencionar cuando un emisor y un receptor se reúnen para realizar una **comunicación sincrónica**.

- Los servicios de colaboración sincrónica cuentan comúnmente con un indicador de *presencia*. En contraste, los servicios de groupware asincrónico carecen del mismo. El mecanismo de *presencia* provee información sobre la disponibilidad (libre, ocupado, desconectado, etc.) de los usuarios.
- **Solución:** Extender el servicio de colaboración asincrónica con un mecanismo indicador de *presencia* en el/los servicio/s de colaboración sincrónica.
- **Participantes:**
  - **Servicio de Groupware Asincrónico:** Permite la colaboración de manera asincrónica. Ejemplos de servicios asincrónicos son: e-mail, foros, y Web-logs. El groupware asincrónico posee comúnmente un mecanismo de identificación de usuarios. Por ejemplo: en el caso del mail se utiliza la dirección de e-mail.
  - **Servicio de Groupware Sincrónico:** Permite la colaboración de manera sincrónica. Ejemplos de servicios sincrónicos son: el Chat, IM, y los whiteboards. Posee comúnmente un mecanismo de identificación de usuario. Por ejemplo: en el caso del Chat se utiliza el nickname.
  - **Servicio de Presencia:** Mantiene y provee información sobre el estado de los usuarios de un sistema de software. Ej.: Conectado, desconectado, ocupado, etc.
- **Colaboración entre servicios:** El usuario se conecta y hace uso del servicio sincrónico, eventualmente cambia su estado y se desconecta. El servicio *sincrónico* comunica cambios de estado al servicio de *presencia*. A su vez, consulta el estado de otras personas para comunicarlo al usuario. El usuario inicia el servicio *asincrónico*. Cada vez que el servicio *asincrónico* debe mostrar al usuario una contribución hecha por alguna otra persona, consulta el estado de la persona en los servicios de presencia disponibles, para mostrarlos junto con la contribución. El usuario observa que el autor de una contribución se encuentra disponible (Por ejemplo: disponible para chatear) e inicia una interacción *sincrónica*.
- **Detalles de Implementación:** La arquitectura (Figura 24) se basa en la integración de servicios provistos por varios servidores, cada uno brindando servicios específicos. En nuestro caso un servidor de usuarios brinda el servicio de *presencia*. El estado de los usuarios, y el cambio del mismo, es comunicado por el servicio *sincrónico* al servicio de *presencia*. De este modo el servicio *asincrónico* es extendido usando el servicio de presencia para comunicar el estado de los usuarios y, de esta forma, potencia la *colaboración sincrónica*. El servicio de presencia se encarga de manipular la información del estado de los usuarios manteniendo un registro de los mismos. El mapeo entre el *servicio asincrónico* y el de presencia, para informar el estado del usuario, se realiza a través de la dirección de e-mail o en su defecto por algún nickname. Esto implica tener y respetar un protocolo<sup>34</sup> que sirva de interfaz entre los servicios.

<sup>34</sup>Si bien actualmente cada servicio de mensajería tiene su propio protocolo definido, se está trabajando en la elaboración de protocolos estándares. Ej.: Jabber, IMMP.

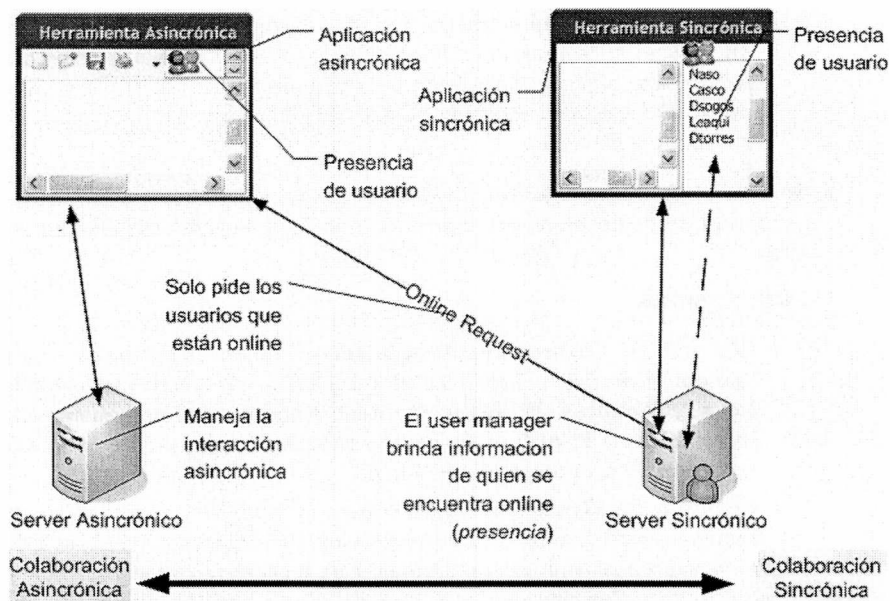


Figura 24: Arquitectura rendezvous

Este patrón es más fácil de implementar cuando: 1) los servicios a integrar son provistos por el mismo sistema. 2) los servicios a integrar son ofrecidos por sistemas independientes pero que comparten un mismo protocolo de comunicación y trabajando sobre la misma red.

■ **Aplicación:** Se debe usar este patrón cuando:

- Un grupo de usuarios que se encuentran distribuidos y utilizando groupware asincrónico, eventualmente deban realizar una tarea que requiera colaborar de manera sincrónica por un corto período de tiempo (Por ejemplo: ponerse de acuerdo sobre la interpretación de algún requerimiento). Y se desea mostrar la posibilidad de contactarse de manera sincrónica.
- Un grupo de usuarios estén trabajando sobre los mismos objetos (Por ejemplo, en un documento de texto, los autores del mismo son más de uno). En algún momento se necesita contactar a alguno de los usuarios que trabajó sobre esos documentos para que clarifique algún punto que no queda claro. Se desea informar de la posibilidad de contactar al usuario para acelerar el proceso de clarificación.

■ **Consecuencias:** Este patrón ofrece los siguientes beneficios:

- Reducir el tiempo y el esfuerzo usado en sincronizar usuarios que desean trabajar de manera sincrónica.



- Ofrecer una visión de los servicios que se deben integrar para soportar la interacción entre las aplicaciones asincrónicas y las aplicaciones sincrónicas.

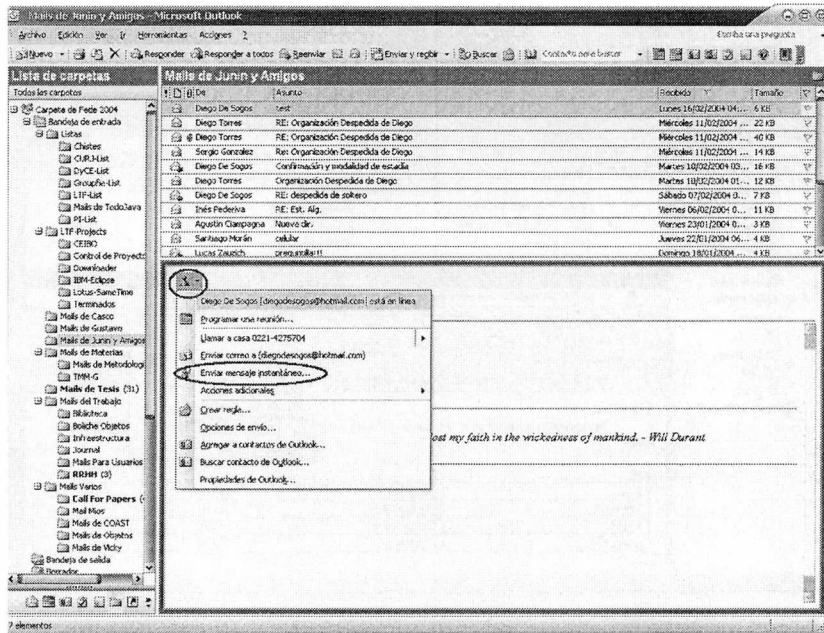


Figura. 25: MS Outlook 2003 con MSN Messenger

- **Ejemplos de uso:** El cliente de mail de MS Outlook integrado con ICQ. El servicio de presencia es provisto por la red de ICQ. El mapeo entre los IDs usados por Outlook (e-mail address) y el ID usado por ICQ (en ICQ se usa UIN que significa "Universal Internet Number" o "número universal de Internet") se lleva a cabo a través de la información de los contactos que posee ICQ. Un solo UIN de ICQ puede ser mapeado a muchas direcciones de mail. Cuando un mail es seleccionado para ser visto en el Outlook, se realiza una búsqueda a través del mapeo en la lista de contactos del ICQ. Si en la lista se encuentra en contacto se muestra el estado (Online, offline, etc.) del contacto en el Outlook.

De manera similar Outlook integra el cliente de mensajería MSN Messenger (Figura 25). El servicio de presencia es provisto por la red de MSN network. El MSN Messenger usa la dirección de e-mail de su propia red (Ej.: Hotmail.com; msn.com y passport.com) como identificación de usuarios. Se puede asociar también una cuenta propia de e-mail a un passport (Por Ejemplo: federico.naso@gmail.com).

Netscape 7.1 se integra también con AOL Instant Messenger (AIM) (Figura 26). Trabaja de manea similar a Microsoft Outlook con ICQ. Pero

sin embargo, el mapeo (permite solo un e-mail por AIM screen name) es realizado a través de la libreta de direcciones de Netscape. Y con en el resto de los casos el servicio de presencia es provisto por la propia red AOL AIM Network.

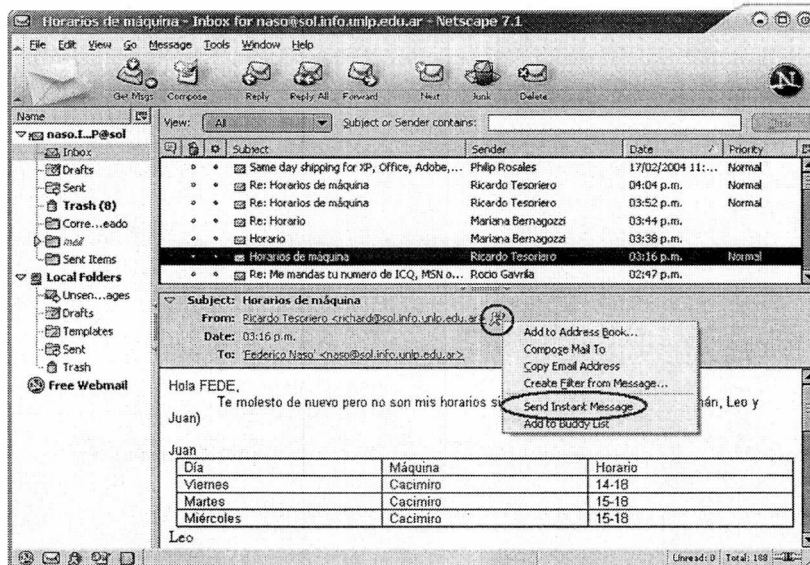


Figura 26: Netscape 7.1 con AOL Messenger

■ **Patrones Relacionados:**

- **Mail Manager:** Permitir la utilización de un cliente de mail como un gestor integral de correo utilizando las agendas contactos de usuarios con sus respectivos datos (Nombre, dirección, teléfono, mail, etc.).
- **Conference:** Simular la interacción cara a cara entre personas o grupos de personas que se encuentran en diferentes lugares geográficamente distantes.
- **Gossip:** Informar a usuarios que estén trabajando en un *espacio de trabajo colaborativo* sobre las actividades que se encuentran realizando otros usuarios dentro del mismo *espacio de trabajo*.

### 5.3.2. Notifier Broker

- **Nombre:** *Notifier Broker*
- **Intención:** Notificar a usuarios de sistemas groupware, sobre la disponibilidad de nuevo contenido o actualización del mismo.
- **También conocido como:** Publisher, Proclama.
- **Problema:** La carencia de un mecanismo que informe, a usuarios o grupos de usuarios, sobre la disponibilidad de material o cambios en el mismo, conlleva a pérdida de tiempo y a uso ineficiente de la información disponible, ya que se debe informar por otros medios que no son parte del sistema.
- **Escenario:** Supongamos que tenemos un equipo de desarrollo distribuido geográficamente en donde la información generada por cada una de los integrantes de los grupos es vital para que los grupos se mantengan actualizados en lo concerniente a cuestiones de análisis, diseño, y documentación del sistema en desarrollo.

Cada nuevo documento de análisis es publicado en un sistema a fin de que los integrantes del resto del grupo que se encuentran en otros lugares geográficos puedan acceder a esa información. Luego los integrantes de los grupos pueden acceder a dicha información y de este modo mantenerse actualizados. Pero los mismo no acceden a la información de inmediato, sino cuando ellos están disponibles o intencionalmente buscan cambios en la misma. Esto hace que información que puede llegar a ser vital en la toma de algunas decisiones no se use o se use en el momento incorrecto dado que se desconocía la misma.

También el responsable de publicar la información podría enviar un mail notificando sobre la publicación de nuevo contenido. Esta solución sería insuficiente ya que se desea mantener un historial de la información y que los usuarios puedan acceder a la misma desde cualquier lugar. Por otro lado si los grupos son demasiado grandes, se cae en el problema de no informar a todos los usuarios que lo desean, como así también de mantener un listado actualizado con las publicaciones a la cual cada usuario esta "subscripto".

- **Fuerzas:**
  - Los servicios de publicación no cuentan con un mecanismo automático que informe mediante eventos a los usuarios sobre cambios en el contenido o sobre nuevo contenido publicado.
  - Los servicios de publicación son casi imprescindibles a la hora de compartir información y mantener un historial de la misma en contextos de grupos distribuidos.
  - Los servicios de comunicación brindan la posibilidad de mantener comunicados a personas geográficamente distantes.

☞ *Un aversión inicial y simplificada de éste patrón fué incluida en el paper "Patrones de Integracion de Servicios Groupware" aceptado y publicado en: CLEI - XXXI conferencia latinoamericana de Informática [61].*

- La comunicación, mediante eventos automáticos, entre grupos es crucial para mantener a los mismos informados de los cambios y decisiones establecidas en el momento que se realizan.
- **Solución:** Integrar un servicio de comunicación que, mediante la programación de eventos automáticos, informen sobre la disponibilidad de nuevo, o de cambios del material publicado a usuarios registrados para que puedan hacer uso del mismo de manera inmediata.
- **Participantes:**
  - **Servicio de publicación de contenido:** es un servicio en donde se puede publicar contenido para que grupos que se encuentran distribuidos puedan acceder a dicho contenido. Servicios de este tipo son los repositorios, los foros, los CMS, los CoWeb. Etc.
  - **Servicio de comunicación:** son servicios de colaboración que permiten mantener a los usuarios comunicados. Ejemplos de los mismos son el Chat, el mail, IM, etc.
- **Colaboración entre servicios:** El servicio de publicación brinda el soporte para gestionar la información que se desea compartir con el resto de los integrantes de los grupos. Además gestiona las diferentes versiones de cada objeto (se habla de objeto ya que algunos sistemas soportan la publicación de contenido variado) que se publica. Cuando se publica nuevo contenido o se cambia contenido ya existente, el sistema notifica mediante algún servicio de comunicación elegido sobre dicho suceso. La notificación puede ser a usuarios individuales o a grupos de usuarios. Como así también, el sistema puede enviar la notificación automáticamente a los usuarios o solo a aquellos que se han suscrito para recibir dicha información. Tales decisiones se deben tomar según la orientación del sistema que se desea desarrollar.
- **Detalles de implementación:** La arquitectura mostrada (Figura 27) se basa en la integración de un módulo que controle y gestione la publicación de contenido (1) y verifique que usuarios solicitaron que se les notifique sobre los cambios en el contenido (2). En caso de que existan usuarios a los cuales notificar, otro módulo encargado de enviar las notificaciones (3) se encarga de hacer llegar las mismas a los usuarios (4). El módulo de comunicación puede contener uno a mas servicios de comunicación según sea necesario.
- **Aplicación:** Se debe usar este patrón cuando:
  - Se desea compartir contenido con grupos de trabajos que se encuentran distribuidos geográficamente y que los mismos sean notificados mediante algún mecanismo de comunicación sobre la disponibilidad de dicho contenido.
  - Se desee mantener un mecanismo de comunicación de información, mediante la publicación de la misma, que se pueda acceder desde lugares geográficamente distantes.

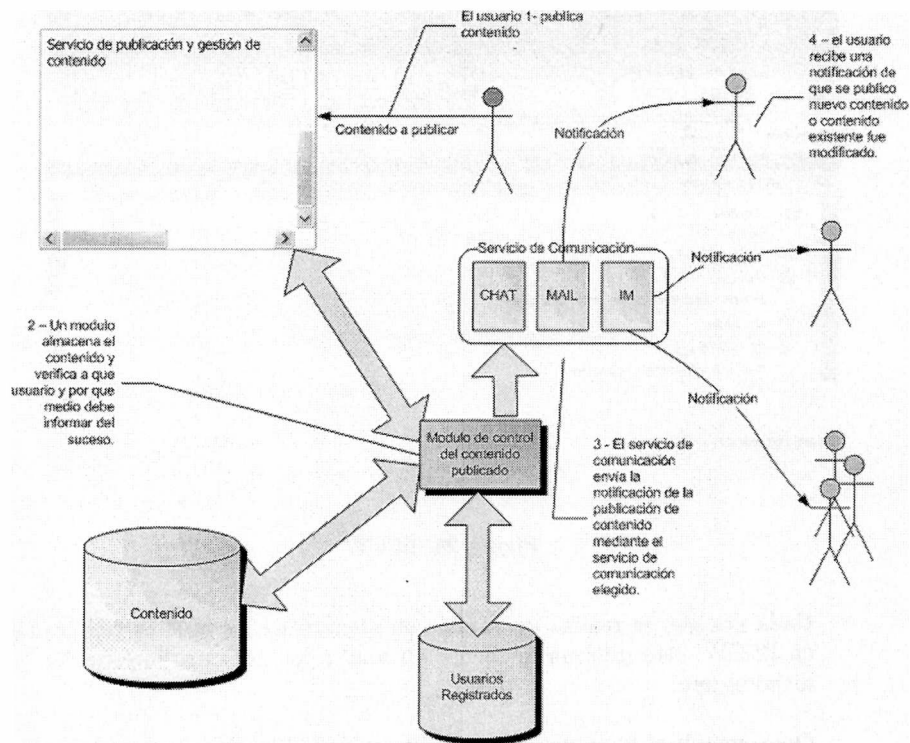
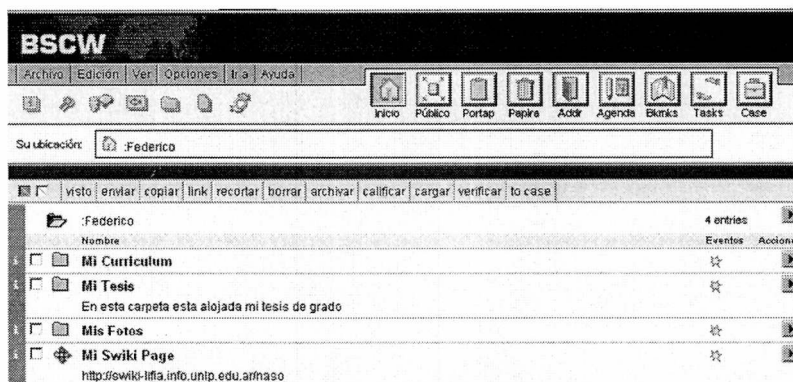


Figura 27: Arquitectura Notifier.

- **Consecuencias:** Este patrón ofrece los siguientes beneficios:
  - Provee una solución al problema de la publicación de contenido y posterior envío de notificaciones automáticas por medio de algún servicio de comunicación.
  - Reduce el tiempo que se emplea en analizar qué servicios se deben integrar para realizar la publicación y envío de notificación a usuarios interesados.
  - Orienta al usuario sobre las pautas que se deben tener en cuenta para la integración de servicios de publicación con servicios de comunicación.
  
- **Ejemplos de uso:** Al ser un patrón de integración entre servicios diversos se pueden encontrar infinidad de ejemplos de uso. BSCW, IBM Lotus Team Workplace (QuickPlace), MS Sharepoint, diversos Foros de discusión. Así por ejemplo podemos encontrar BSCW. BSCW es una de las implementaciones más completas (Figura 28). Permite compartir documentos, direcciones URL<sup>35</sup>. Los usuarios poseen un espacio de trabajo el cual puede ser compartido mediante una invitación con otros usuarios.

<sup>35</sup>URL: Uniform Resource Locator.



BSCW © 1995-2004 FIT and OrbiTeam

Figura 28: BSCW

Cada vez que se realiza un cambio en algunos de los objetos publicados en BSCW, este informa mediante un mail a las personas interesadas en dicho objeto.

Otro ejemplo es Sharepoint. Microsoft posee el Microsoft Content Management Server y Microsoft Sharepoint Portal Server. Facilita la colaboración proporcionando una forma de compartir contenido y documentos a personas individuales o equipos de trabajo. Además diversos foros, permiten que luego de realizar un aporte, el sistema nos informe cuando se realiza algún cambio sobre el mismo, por ejemplo, se recibe una respuesta a una pregunta.

■ **Patrones Relacionados:**

- **Mail Manager:** Permitir la utilización de un cliente de mail como un gestor integral de correo utilizando las agendas de contactos de usuarios con sus respectivos datos (Nombre, dirección, teléfono, mail, etc.).

### 5.3.3. Gossip

- **Nombre:** *Gossip*
- **Intención:** Informar a usuarios que estén trabajando en un *espacio de trabajo colaborativo* sobre las actividades que realizan otros usuarios dentro del mismo *espacio de trabajo*.
- **También conocido como:** Vigía.
- **Problema:** Usuarios que se encuentran trabajando en el mismo *espacio de trabajo colaborativo*, no se percatan de que hay otros trabajando al mismo tiempo y pierden la posibilidad de interactuar entre ellos.
- **Escenario:** Supongamos que se está desarrollando el sitio hipermedial de una compañía virtual. Este tipo de compañías no posee un lugar físico común donde los empleados trabajan, sino que los mismos están distribuidos en diferentes lugares geográficos. Los empleados rara vez tienen contacto cara a cara. Es por ello que el desarrollo se lleva a cabo utilizando una herramienta de hipermedia colaborativa. La cual, posibilita la tarea de desarrollar el sitio entre un grupo de personas trabajando simultáneamente. La cantidad de diseñadores y demás usuarios que están desarrollando el sitio es variable y pueden trabajar en diferentes instantes de tiempo. Es por ello que es imprescindible contar con algún mecanismo que indique quienes están trabajando y en que objetos sobre el sitio, para así poder sincronizar el trabajo. De otro modo, al no contar con mecanismos que informen explícitamente las tareas que se están llevando a cabo por los diferentes usuarios, no se sabría quién y que objetos están siendo editados al mismo tiempo, y en situaciones de conflictos, como por ejemplo, varios usuarios manipulan el mismo objeto, se perdería el trabajo o el mismo se llevaría a cabo de una manera totalmente ineficiente dado que se tendrían que resolver los conflictos por algún otro método alternativo.
- **Fuerzas:**
  - Los usuarios no se percatan de la existencia de otros usuarios en el mismo *espacio de trabajo*.
  - Por no tener conocimiento de quien está trabajando en el mismo *espacio de trabajo*, en casos donde surjan conflictos, no existe una forma de sincronizar el trabajo entre ellos.
  - Es deseable poseer cierta información de quien y que objetos se están manipulando en los distintos *espacios de trabajo*.
- **Solución:** Utilizar algunos de los mecanismos de *presencia de usuario* para informar al resto sobre la presencia de otros usuarios en el mismo *espacio de trabajo*.
- **Participantes:**
  - **Espacio de trabajo colaborativo:** permite a un conjunto de usuarios navegar y manipular objetos en un espacio de trabajo colaborativo. Los ambientes de trabajo colaborativo soportan la colaboración y coordinación entre usuarios.

👉 Una versión inicial y simplificada de este patrón fue incluida en el paper "Patrones de Integración de Servicios Groupware" aceptado y publicado en: CLEI - XXXI conferencia latinoamericana de Informática [61].

- **Servicio de Presencia:** Mantiene y provee información sobre el estado de los usuarios de un sistema de software. Ej.: Conectado, desconectado, ocupado, etc.
- **Colaboración entre servicios:** El ambiente de trabajo colaborativo aporta el poder para la creación de estructuras mediante la manipulación de objetos compartidos. Por su parte el servicio de presencia aporta información sobre las actividades que están desarrollando y la disponibilidad del resto de los usuarios. De dicho modo mediante el servicio de presencia se puede conocer que usuarios están trabajando en el mismo *espacio de trabajo* y que objetos de la estructura están manipulando para así poder sincronizar el trabajo de los mismos.
- **Detalles de Implementación:** En este caso un servidor de usuarios brinda el servicio de *presencia*. El estado de los usuarios, y el cambio del mismo, es comunicado por el servicio *sincrónico* al *servicio de presencia*. De este modo el espacio de trabajo colaborativo, es extendido usando el servicio de presencia para comunicar el estado de los usuarios que se encuentran online, como así también el lugar del workspace que están ocupando y las actividades que se están desarrollando sobre los mismos. El servicio de *presencia* es el encargado de manipular la información del estado de los usuarios manteniendo un registro de los mismos.

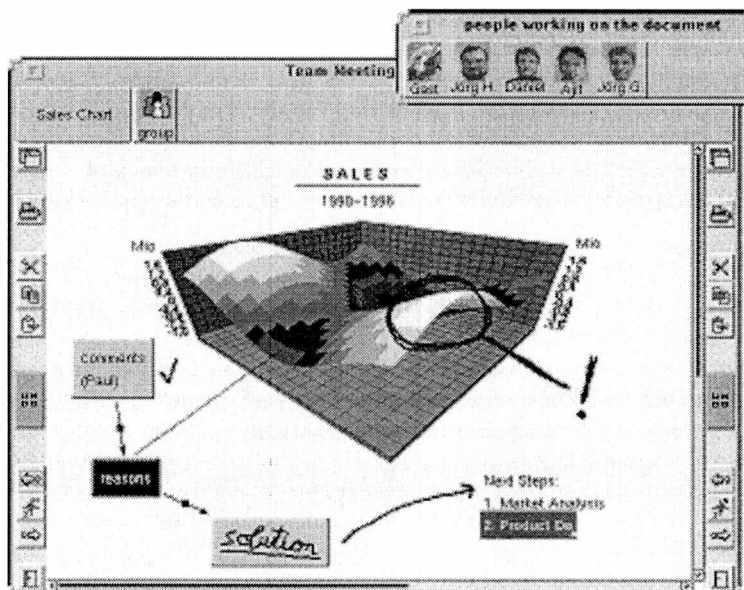


Figura 29: Dolphin

- **Aplicación:** Se debe usar este patrón cuando:
  - Desea conocer que personas se encuentran trabajando en el mismo *espacio de trabajo colaborativo*.



- Cuando se desee coordinar el trabajo de las personas que están usando el mismo espacio de trabajo.
- **Consecuencias:** Este patrón ofrece los siguientes beneficios:
  - Brinda la posibilidad de percatarse de las personas que están trabajando en el mismo *espacio de trabajo* y además saber en que lugar del mismo lo están haciendo.
  - Ofrece una clara visión de que servicios se deben integrar si se desea tener un *espacio de trabajo colaborativo* y en el mismo un servicio de presencia de usuario.

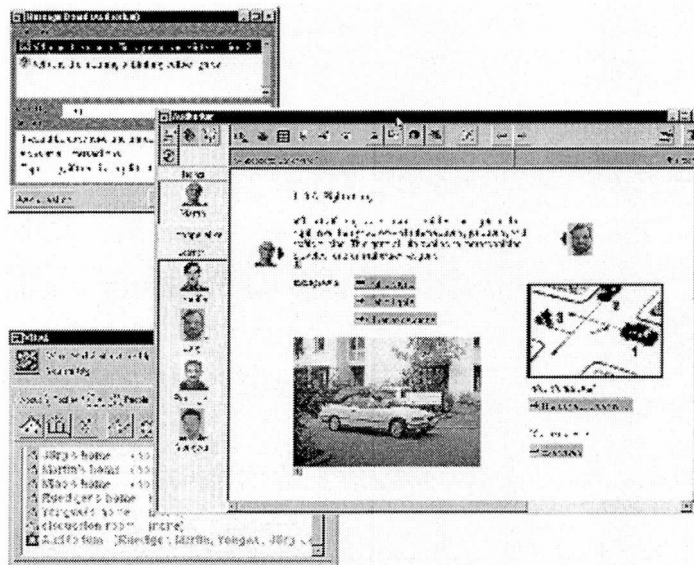


Figura 30: VITaL

- **Ejemplos de uso:** Dolphin (Figura 29) está basado en un modelo de datos de hypermedia dentro de un espacio de trabajo público. Este modelo soporta la construcción de estructuras, como así también la representación de dependencias entre documentos.
- Otras implementaciones de espacios de trabajo colaborativos en donde se use mecanismo de presencia son: SEPIA es un ambiente que soporta producción de documentos de forma sincrónica y asincrónica entre autores que se encuentran en grupos distribuidos geográficamente. VITaL<sup>36</sup> es un ambiente hypermedial para la enseñanza y aprendizaje colaborativo (Figura 30). Utiliza el concepto de rooms en donde los participantes pueden hacer sus contribuciones. Y CHIPS provee soporte hypermedial a procesos de desarrollo realizados en ambientes cooperativos.

<sup>36</sup>VITaL: Virtual Teaching and Learning.

- **Patrones Relacionados:**

- **Rendezvous:** Informar a usuarios de un sistema de groupware asincrónico sobre la posibilidad de colaborar con otras herramientas de manera sincrónica (groupware sincrónico).

#### 5.3.4. Chat Channel

- **Nombre:** *Chat Channel*
- **Intención:** Permitir la comunicación textual sincrónica (Chat) entre grupos de usuarios dentro de aplicaciones colaborativas.
- **También conocido como:** Communication Channel.
- **Problema:** La carencia de servicios de comunicación en las aplicaciones colaborativas, conducen a la imposibilidad de comunicarse y sincronizarse con el resto de los usuarios que se encuentran trabajando dentro del mismo espacio grupal de forma simultánea.
- **Escenario:** Supongamos que tenemos dos grupos distribuidos trabajando sobre un diseño de un sistema. La comunicación es vía e-mail sobre un documento de especificación. En el documento figura también un diseño de la arquitectura del sistema. En determinado caso se necesita usar una aplicación colaborativa; para éste caso se utiliza una pizarra colaborativa para hacer un diagrama y clarificar ciertos puntos del diseño con un dibujo. Al usar la pizarra, los grupos están trabajando sobre el mismo dibujo colaborativamente, pero como la herramienta no posee un servicio de comunicación integrado se hace casi imposible sincronizar y discutir sobre dicho dibujo.
- **Fuerzas:**
  - Las aplicaciones colaborativas son útiles en grupos de trabajos distribuidos ya que permiten a todo los grupos trabajar concurrentemente sobre el mismo espacio de manera concurrente.
  - Al no poseer herramientas de comunicación sincrónica, las aplicaciones colaborativas que son usadas en ambientes distribuidos, se tornan difíciles de usar ya que es casi imposible comunicarse con el resto de los participantes en tiempo real y así discutir sobre lo que se esta llevando a cabo en ese momento.
  - Los servicios de comunicación textual sincrónica (Chat) brindan la posibilidad de comunicarse en tiempo real con grupos de trabajo que se encuentran distribuidos.
- **Solución:** Integrar una aplicación colaborativa con un servicio de comunicación sincrónica por texto (Chat) para que de este modo los usuarios puedan comunicarse en tiempo real y así puede discutir su trabajo y sincronizar el mismo.
- **Participantes:**
  - **Aplicación Colaborativa:** permite que varias personas puedan compartir un área de trabajo en la cual se puede trabajar simultáneamente a través de una red.
  - **Chat:** Canal de comunicación textual sincrónico entre varios usuarios, donde cada uno puede comunicarse de manera textual sincrónica.

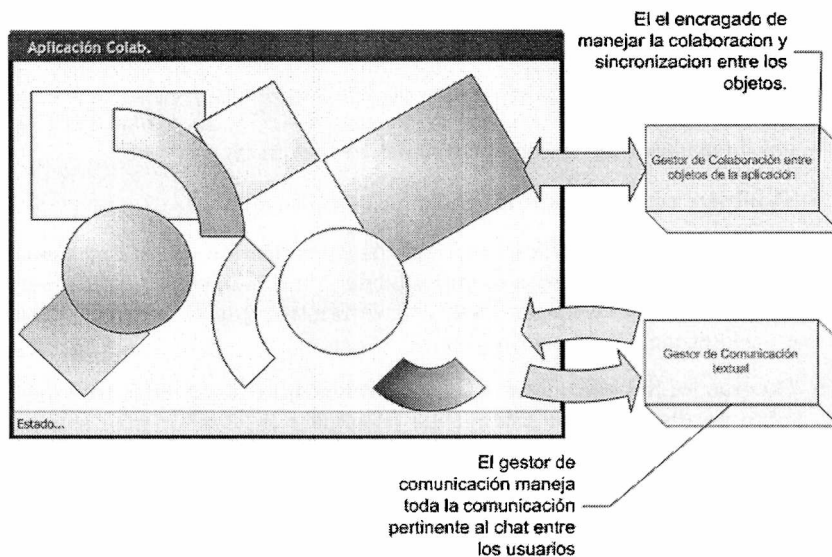


Figura 31: Arquitectura Chat Channel

- **Colaboración entre servicios:** En realidad estos servicios no interactúan entre sí. Es decir no colaboran mutuamente, sino que el servicio de Chat aporta su potencia de comunicación sincrónica o una vía de comunicación para fortalecer el trabajo realizado en la aplicación colaborativa, paliando así, los problemas de comunicación que se tienen de no contar con él.
- **Detalles de implementación:** Se debe incluir en las aplicaciones colaborativas, además del mecanismo para sincronizar el trabajo que se está llevando a cabo (en éste caso vemos como ejemplo la edición de un dibujo en una pizarra) un mecanismo o módulo de comunicación sincrónica que permita la interacción por medio de texto (Chat) entre los usuarios que se encuentran trabajando concurrentemente (Figura 31). Este módulo es el único encargado de mantener la comunicación entre los grupos. El módulo puede seguir un protocolo estándar de comunicación, como por ejemplo IRC o funcionar con una ad-hoc.
- **Aplicación:** Se debe usar este patrón cuando:

  - Se necesite comunicación textual sincrónica entre los usuarios que están compartiendo trabajo en una aplicación colaborativa.
  - Se necesite implementar una aplicación colaborativa en donde los grupos de trabajo están distribuidos geográficamente.
- **Consecuencias:** Este patrón ofrece los siguientes beneficios:

  - Da una clara visión de que servicios se deben integrar para implementar una aplicación colaborativa para ser usada en ambientes dis-

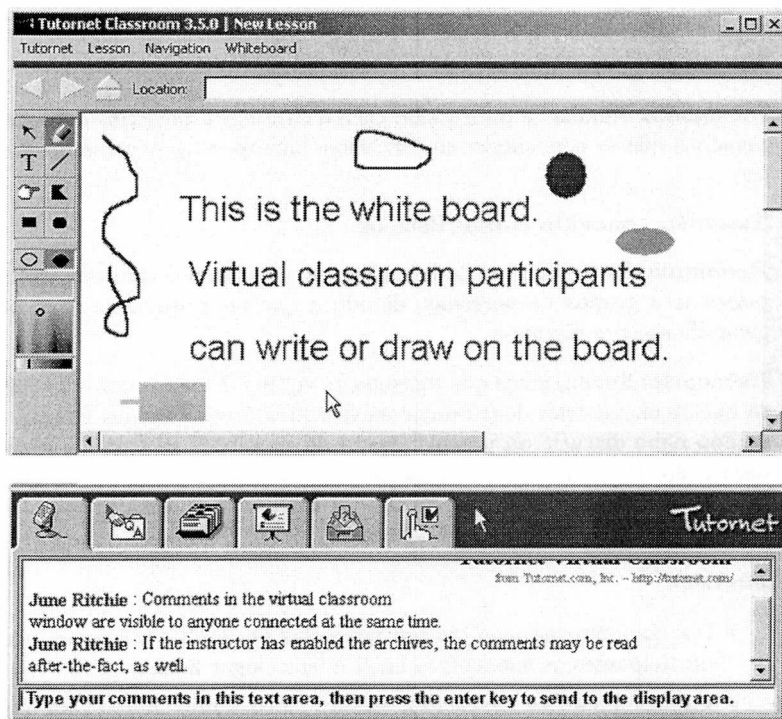


Figura 32: Pizarra y Chat de "virtual classroom".

tribuidos donde los usuarios deban comunicarse en tiempo real entre ellos.

- **Ejemplos de uso:** Un claro uso de este tipo de patrones se puede encontrar en (Figura 32) donde se lo utiliza como un *virtual classroom*<sup>37</sup> o un Aula Virtual. Aquí se combina la pizarra y un Chat para ser usado en un curso de educación a distancia. Otro ejemplo de uso es SnapChat. SnapChat es un Chat desarrollado por el LIFIA de la Universidad Nacional de La Plata<sup>38</sup> en conjunto con el instituto Alemán GMD<sup>39</sup>. Snapchat posee una pizarra en la cual se puede dibujar sobre la misma y hacer marcas sobre los dibujos para intercambiar opiniones sobre los mismos. Fue desarrollado para ser integrado en un ambiente de educación a distancia.
- **Patrones Relacionados:**
  - **Notifier Broker:** Notificar a usuarios de sistemas groupware, sobre la disponibilidad de nuevo contenido o actualización del mismo.

<sup>37</sup><http://www.odu.edu/dl/ct/bb/tutorials/virtualclass.html>

<sup>38</sup>**LIFIA:** <http://www-lifia.info.unlp.edu.ar>

<sup>39</sup>**GMD:** <http://www.gmd.de>

### 5.3.5. Virtual Conference

☞ Una versión inicial y simplificada de éste patrón fué incluida en el paper "Patrones de Integración de Servicios Groupware" aceptado y publicado en: CLEI - XXXI conferencia latinoamericana de Informática [61].

- **Nombre:** *Virtual Conference*
- **Intención:** Simular la interacción cara a cara entre personas o grupos de personas que se encuentran en diferentes lugares geográficamente distantes.
- **También conocido como:** Reunión.
- **Problema:** Imposibilidad de llevar a cabo reuniones o conferencias entre personas o grupos de personas, debido a que las mismas se encuentran geográficamente distantes.
- **Escenario:** Supongamos que tenemos un equipo de diseñadores de software que se encuentran distribuidos en distintas áreas (lugares físicos). Ese equipo debe discutir un posible diseño de arquitectura para un próximo desarrollo. Dado que los participantes se encuentran distribuidos geográficamente, una discusión por medio de mail o chat sería improductiva, desordenada y extremadamente lenta.
- **Fuerzas:**
  - Las conferencias remotas son apreciadas porque no requieren que los participantes se encuentren en el mismo lugar físico.
  - La posibilidad de interactuar en una video/audio conferencia remota reduce el tiempo y esfuerzo que se requiere para resolver un problema mediante otras vías, como por ejemplo el e-mail.
- **Solución:** Utilizar la integración de los servicios de video, audio y de chat como un paquete de comunicación integral para permitir la comunicación entre personas o grupos de personas que se encuentran geográficamente distantes.
- **Participantes:**
  - **Video:** Se utiliza para poder comunicarse con otra persona en forma *sincrónica* y de modo que las personas se puedan ver entre sí por medio de una cámara Web. Simula la interacción cara a cara.
  - **Audio:** Se utiliza para poder comunicarse con otra persona en forma *sincrónica* y de modo que las personas se puedan escuchar como si estuvieran en el mismo espacio físico.
  - **Chat:** Canal de comunicación textual sincrónico entre varios usuarios, donde cada uno puede comunicarse de manera textual sincrónica.
- **Colaboración entre servicios:** Si bien los servicios de video, audio y el de Chat no interactúan entre sí, es aconsejable que las aplicaciones de conferencia integren todos los servicios en una misma aplicación. Cada una utilizando su propio protocolo (y por ende, su propio Server) de comunicación. De éste modo cuando se crea la conferencia entre todos los interesados, los servicios estarán disponibles al mismo tiempo para brindar soporte a la reunión virtual.

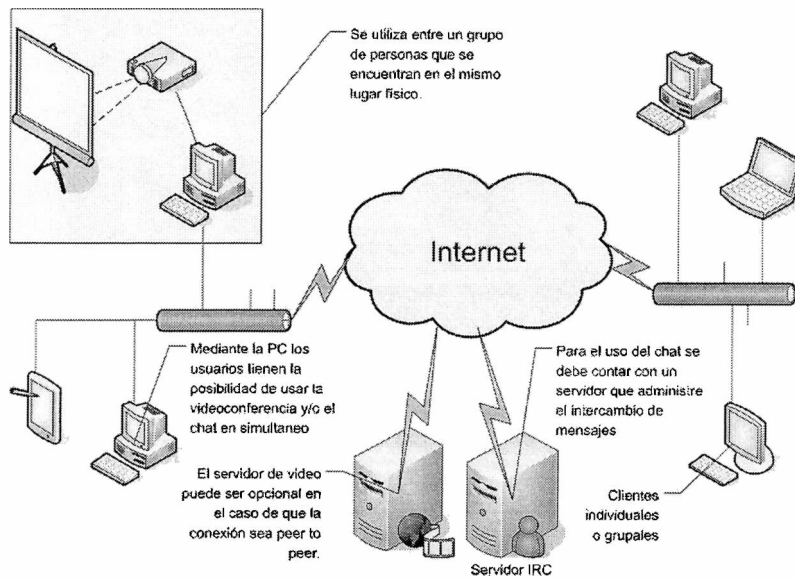


Figura 33: Arquitectura de una conferencia.

- **Detalles de Implementación:** La arquitectura (Figura 33) se basa en la integración del servicio de Chat provisto por un Server dedicado a tal efecto y el servicio de videoconferencia (integrando audio y voz).
- **Aplicación:** Se debe usar este patrón cuando:

  - Se desee generar un sistema integral de comunicación para emular las reuniones cara a cara.
  - Se necesite un sistema de comunicaciones y dado que el audio y el video no son lo suficientemente fiables, se deba recurrir al chat para aclarar ciertos diálogos.
  - Se desee documentar la video/audio conferencia por medio de una herramienta textual.
- **Consecuencias:** Este patrón ofrece los siguientes beneficios:

  - Ofrece una visión de los servicios que se deben integrar si se desea desarrollar un sistema de comunicación que permita las reuniones cara a cara de grupos de trabajo geográficamente distantes.
  - Reduce el tiempo usado en conferencias en donde la calidad el audio o la imagen no son lo suficientemente fiables y se debe recurrir al Chat para aclarar el dialogo.

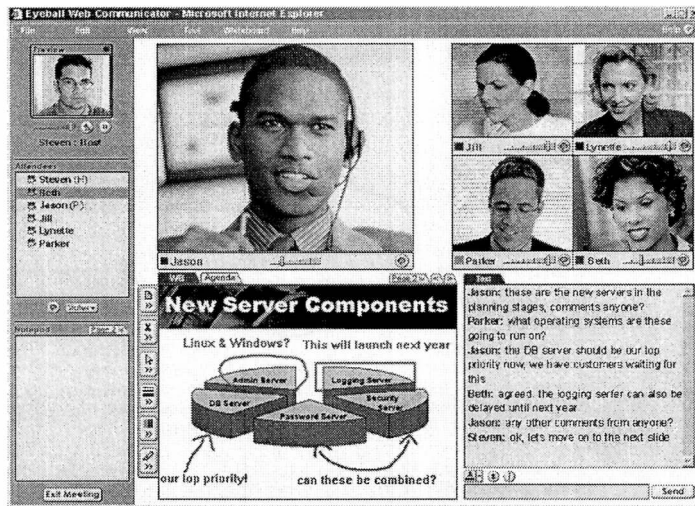


Figura 34: Eyeball Web Communicator

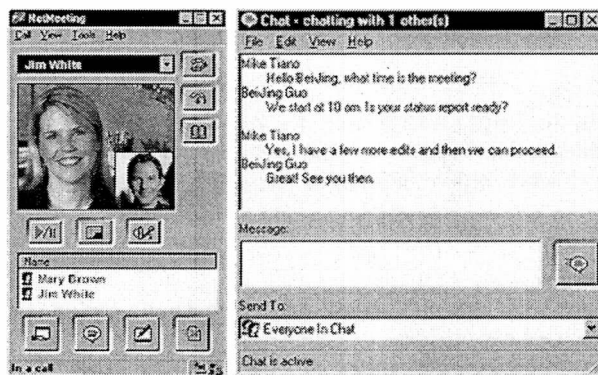


Figura 35: MS NetMeeting videoconferencia y Chat.

- **Ejemplos de uso:** Un ejemplo de este tipo de integración son el conocido el NetMeeting (Figura 35) de Microsoft y el EyeBall (Figura 34) de Eyeball Networks Inc. [8]. NetMeeting ofrece la posibilidad de hacer una conferencia y un Chat entre varias personas. La ventana de video y la de Chat están separadas. Integra además un whiteboard. (Cáp. 4.1.6). Por su lado el EyeBall esta orientado al Web, las aplicaciones están embebidas en la misma ventana. También posee video y Chat. Además de una pizarra whiteboard.
- **Patrones Relacionados:**
  - **Rendezvous:** Informar a usuarios de un sistema de groupware asincrónico sobre la posibilidad de colaborar con otras herramientas de manera sincrónica (groupware sincrónico).



### 5.3.6. Mail Manager

- **Nombre:** *Mail Manager*<sup>40</sup>
- **Intención:** Permitir la utilización de un cliente de mail como un gestor integral de correo utilizando las agendas de contactos de usuarios con sus respectivos datos (Nombre, dirección, teléfono, mail, etc.).
- **También conocido como:** Cliente de Mails. Gestor de Mails.
- **Problema:** La carencia de gestores de mail integrados con agendas de contactos de usuarios (Nombre, dirección, dirección de e-mail, etc.) hace que el envío de e-mail sea improductivo, tedioso y lento. Dado que el usuario debe estar buscando la dirección de e-mail en una agenda independiente y luego copiando dicha dirección en el cliente de e-mail.
- **Escenario:** Si tenemos una cuenta de correo electrónico (o mail), en la cual, a lo largo de los años la dirección de la misma la hemos compartido con un cierto grupo de personas. Dichas personas forman nuestra lista de contactos. El intercambio de mail entre dichas personas es fluido.

Supongamos entonces que tenemos un número, dentro de todo, relativamente bajo de contactos, digamos unos 250. Acordarnos las 250 direcciones de e-mail de los contactos de memoria es casi imposible. Existen muchos clientes de mail que no guardan las direcciones de mail de las personas a las cuales alguna vez les hemos escrito un e-mail.

Para almacenar las direcciones de e-mail que poseemos existen agendas de contactos en las cuales se almacena la información de nuestros contactos. Cada vez que deseamos comunicarnos con alguno de nuestros contactos, tomamos la agenda, miramos su dirección de correo, vamos al cliente de mail y copiamos la dirección en el campo correspondiente para dicha dirección. Si enviamos un mail por día esto no es un trabajo pesado, pero imagínense enviar 50-100 e-mails diarios y a más de un destinatario por cada mail. El trabajo de buscar la dirección del contacto y copiarla en el cliente se torna en verdad un trabajo inhumano. Además cada vez que recibimos un mail de una persona que no está en la lista, tenemos el trabajo de volcar la dirección y los datos de dicha persona en la agenda. En cambio si contáramos con un gestor de contactos integrado al gestor de mail, tanto la recopilación como la búsqueda de la dirección de mail sería casi automática.

- **Fuerzas:** Los clientes de e-mail son apreciados por la facilidad que aportan el envío/recepción y la gestión de los mismos.
  - Las agendas de contactos son muy útiles para mantener la información de los contactos de una persona de manera ordenada y consistente.

---

<sup>40</sup> Este patron es el más utilizado y conocido de todos los patrones del catálogo. Hoy por hoy es muy difícil encontrar un cliente de mail que no tenga una agenda de contactos integrada. Si bien es algo trivial, no deja de ser importante. Es por ello que forma parte del catálogo.

- La carencia de gestores de contactos en los clientes de e-mail hace que la recolección de mail en los mismos se lleve a cabo de manera manual. Lo que trae aparejado, inconsistencia, pérdida de información y tiempo.
  - La integración de un cliente de e-mail con una agenda de contactos agiliza el trabajo de buscar la dirección del contacto al cual se le desea enviar un e-mail.
  - La recolección y administración de contactos por medio de una agenda integrada al cliente de mail hace que la información sea consistente, ordenada y de fácil uso.
- **Solución:** Integrar el servicio de mail para la comunicación (*asincrónica*) con una agenda que gestione los contactos en donde se tienen los datos de las personas, incluyendo su dirección de mail la cual es utilizada por el cliente de mail para enviar los mensajes a dicha dirección.
  - **Participantes:**
    - **Cliente de Mail:** es un servicio que puede transportar a través de Internet todo tipo de información (documentos, presentaciones, audio, programas y demás cosas).
    - **Agenda de contactos pública:** mantienen una lista e información de los contactos. Tanto de usuarios como de grupos de usuarios (agendas corporativas).

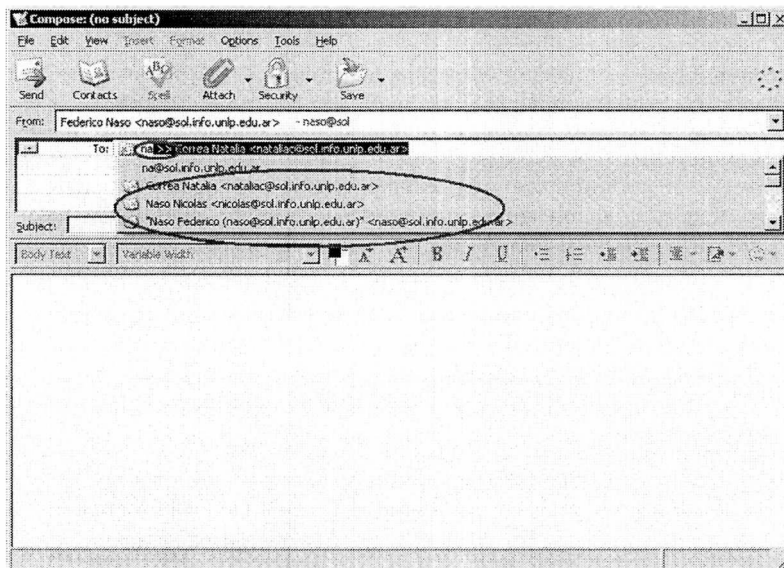


Figura 36: Envío de mail con Thunderbird

- **Colaboración entre servicios:** El cliente de e-mail se encarga del envío/recepción y la gestión de las cuentas de e-mails. El gestor de contactos

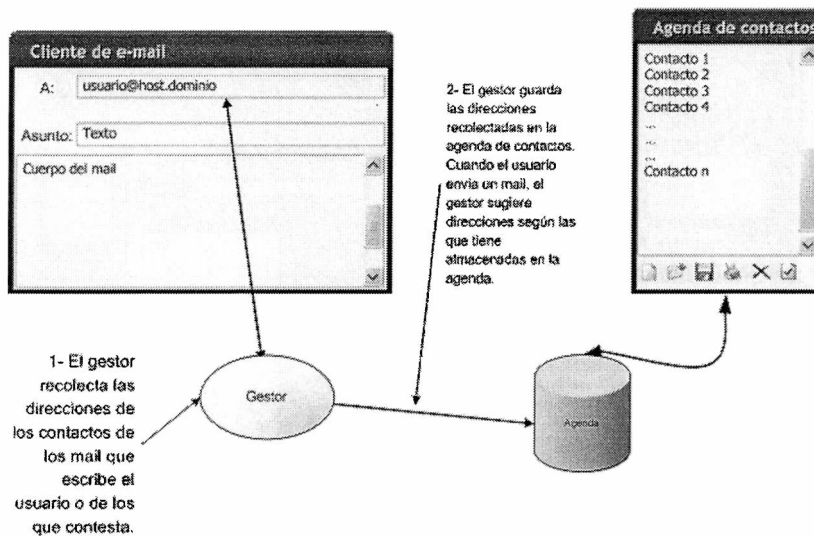


Figura 37: Arquitectura Mail Manager

cada vez que el usuario responde un mail, guarda la información del contacto (nombre y dirección de correo) en su agenda de contactos, es decir, captura la información del usuario al cual se le envía o se recibe un mail. El usuario puede además agregar contacto manualmente. Cuando el usuario desea escribir un e-mail, en el campo "a:" o "para:" pone la dirección de e-mail de la persona destinataria del e-mail (Figura 36). En este momento, como se ilustra en el ejemplo, cuando solo ingresó unas pocas letras de la dirección de e-mail (o del nombre del destinatario) el gestor de contactos busca en su agenda los nombres que mas se acercan a esa dirección y sugiere una lista de posibles candidatos. Si el candidato esta entre ellos, el usuario no tiene más que seleccionarlo y el gestor completará el campo con la dirección de e-mail correcta.

En algunos casos en gestor de e-mail oculta la dirección de e-mail y solo muestra el nombre completo del contacto (en este caso Federico Naso <naso@sol.info.unlp.edu.ar>).

- **Detalles de implementación:** La arquitectura (Figura 37) se basa en la integración de un gestor de contactos que recolecte (1) los contactos de los mail y los almacene en la agenda de contactos. Luego el gestor de contactos sugiere en base a lo que contenga almacenado en la agenda, direcciones de e-mail (2) cuando el usuario desea escribir o componer un e-mail.
- **Aplicación:** Se debe usar este patrón cuando:
  - Se desea construir un cliente de correo que contenga una agenda de gestión de contactos y que sirva de gestor de mails.

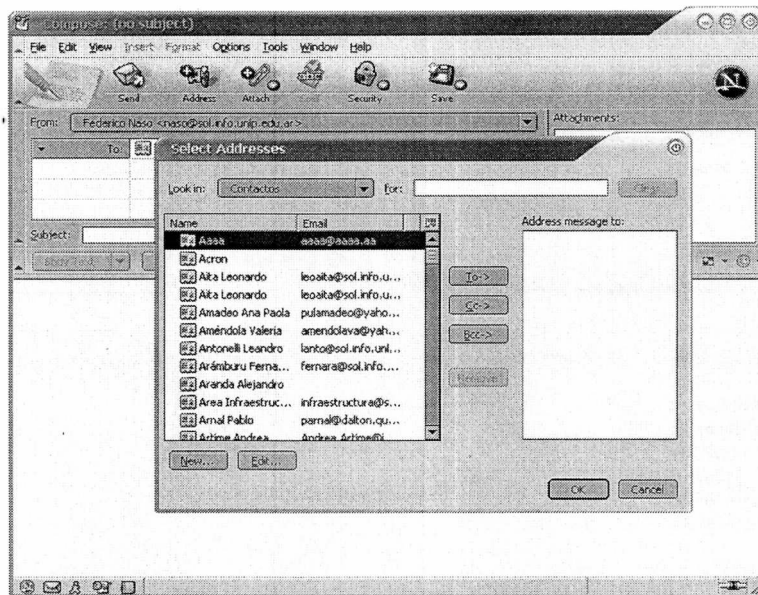


Figura 38: Netscape con su agenda de contactos

- **Consecuencias :** Este patrón ofrece los siguientes beneficios:
  - Minimiza el trabajo de buscar y recordar las direcciones de mail de los contactos y copiarla en el cliente de correo.
  - Ofrecer una visión de los servicios que se deben integrar para soportar la interacción entre las agendas de contactos y el servicio de mail.
- **Ejemplos de uso:** MS Outlook [19] y NetScape [20] (Figura 38) son las aplicaciones más usadas.
- **Otras Implementaciones son:** PocoMail, IncrediMail, Pegasus, Eudora, Yahoo, Outlook Express, etc.
- **Patrones Relacionados:**
  - **Rendezvous:** Informar a usuarios de un sistema de groupware *asincrónico* sobre la posibilidad de colaborar con otras herramientas de manera *sincrónica* (groupware sincrónico).
  - **Notifier Broker:** Notificar a usuarios de sistemas groupware, sobre la disponibilidad de nuevo contenido o actualización del mismo.

#### 5.4. Relación entre los patrones

A continuación se presenta un gráfico (Figura 39) de cómo se encuentran relacionados cada unos de los patrones.

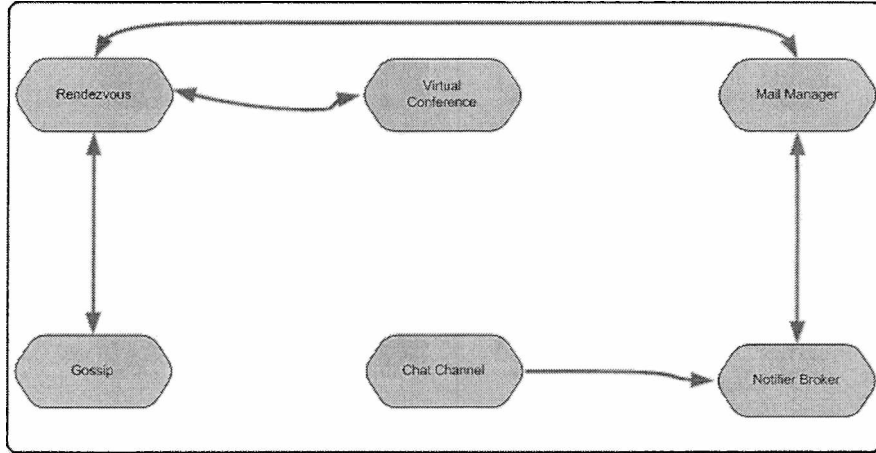


Figura 39: Relación entre los patrones.

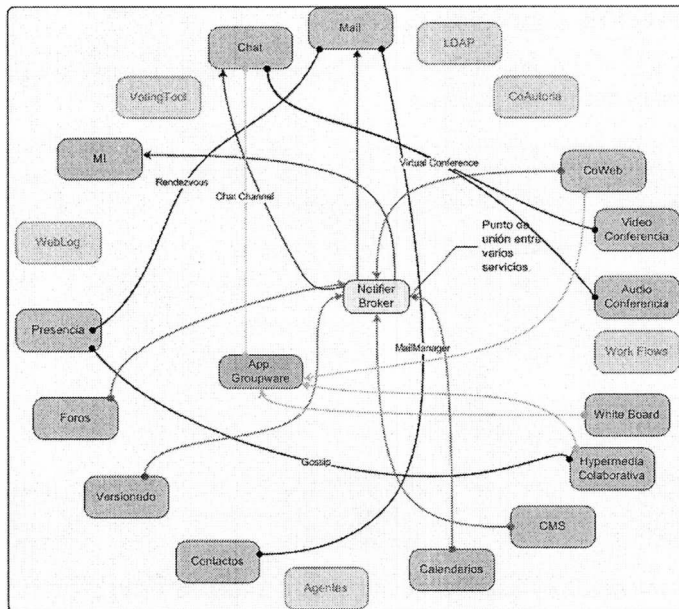


Figura 40: Servicios Integrados.

En la Figura 40, podemos observar, en negro las flechas que marcan la integración entre servicios (en azul) groupware.

El óvalo verde y el naranja indican que el patrón puede integrar múltiples servicios o que en realidad puede ser usado por varios servicios groupware.

Como se puede apreciar, el servicio utilizado en la mayoría de las integraciones es el servicio de e-mail. Hoy en día luego de más de 30 años de la existencia del e-mail sigue siendo uno de los servicios de comunicación más utilizado por las personas. Si bien existen otros servicios de comunicación, no han podido desplazar al e-mail por su gran potencial y su fácil utilización.

# Capítulo VI

Cobertura de Requerimientos

## 6. Cobertura de requerimientos

Este capítulo trata de cómo ésta tesis logra cubrir los requerimientos planteados en el *segundo* capítulo. En este capítulo veremos claramente como, con el catálogo de patrones confeccionado en el *capítulo 5*, se cubren los requerimientos necesarios para la *integración de servicios groupware*.



## 6.1. Primer Requerimiento: Espacio conocido

Por medio del catálogo visto en el *capítulo 5* se logra tener documentado un espacio de soluciones conocido, descriptos en términos atómicos (patrones de integración de servicios). Dado que en dicho catálogo se encuentran documentadas las soluciones a los problemas encontrados en la *integración de servicios groupware*, ese espacio de soluciones puede ser re-usado en otros diseños que contengan problemas similares.

Las soluciones documentadas no son todas las soluciones existentes, sino un subconjunto inicial que forman el punto de partida para documentar otras soluciones halladas.

Por cada patrón documentado, se describe el problema que ataca, el contexto donde se aplica y se detalla una solución. Con lo cual cada vez que un diseñador tiene la tarea de diseñar una aplicación donde se deban integrar servicios groupware, debe buscar en dicho catálogo los problemas que cada una de las integraciones resuelve, el contexto donde se usa y ver si su problema se encuentra dentro de algún otro ya resuelto. De esta manera se aprovecha el espacio de soluciones conocido y se lo reutiliza según su conveniencia.

Existen muchos patrones que han ido apareciendo con el tiempo y que se han convertido en soluciones estándar para diferentes problemas, y por supuesto no limitados al ámbito informático sino que podemos encontrar patrones de arquitectura, de economía, etc.

El espacio conocido plantea por cada problema una solución descrita en términos de patrones de integración. La solución planteada es una guía a tener en cuenta. No significa que en dicha solución estén los pasos a seguir como una receta, sino que propone el líneas generales ideas y "tips" que otros utilizaron anteriormente.

En el caso que la solución no se encuentre entre las propuestas por el catálogo, se debe documentar dicha solución e integrarla al mismo para que forme parte del espacio conocido y pueda ser usada en un futuro en otro diseño que tenga que resolver un problema similar.

## 6.2. Segundo Requerimiento: Documentar la integración

La documentación de la integración de servicios es llevada a cabo mediante el catálogo de patrones, el cual permite una clara interpretación de la integración.

Si nos enfrentamos a un problema dentro del ámbito de la mensajería, lo más adecuado es que comencemos por estudiar patrones comunes en sistemas de mensajería, en lugar de intentar empezar desde cero aplicando patrones. Así descubriremos nuevos patrones que luego de documentados, siguiendo la estructura propuesta, engrosarán nuestro catálogo, patrones que por otra parte se han mostrado efectivos ante el problema en concreto que estamos tratando.

En la documentación se pueden destacar los siguientes ítems:

- La consistencia de la documentación se logra gracias el uso de patrones y a la plantilla propuesta para su respectiva documentación.
- Cada patrón brinda información sobre como ataca el problema, dicha información documentada sirve para futuras integraciones las cuales son llevadas a cabo utilizando el espectro de espacios conocidos.
- Cada integración documentada ataca un problema específico, así como también proveen la solución al mismo. No hay ambigüedad en las soluciones, las mismas atacan un solo y único problema.
- Las relaciones entre las integraciones están dadas por los patrones relacionados. Para un problema dado, es posible que se deba utilizar más de un patrón, y lo más probable es que esos patrones estén relacionados.
- Las diferencias y similitudes entre las soluciones planteadas son detalladas teniendo en cuenta lo que cada una de ellas brinda y aporta a cada uno de los patrones.

El formato de patrones elegido fue validado enviando uno de los patrones, en particular el primer patrón que integra el catálogo, *rendezvous* a un congreso específico en el tema. Dicho congreso es el PloP (Pattern Languages of Programs) [24] que se llevó a cabo entre 8 y el 12 de septiembre del 2004 en Allerton Park, Monticello, Illinois, USA. Dicho patrón fue aceptado y se discutió su formato e importancia en un Writers' Workshop Groups [35].

### 6.3. Tercer Requerimiento: Análisis del problema

Diseñar una arquitectura basada en patrones de diseño es una tarea compleja que reporta muchos beneficios, pero prácticamente imposible sin un estudio previo del dominio del problema que estamos tratando.

El mero hecho de utilizar patrones de diseño "porque sí" y comenzar a realizar su implementación sin tener en cuenta el dominio de nuestro modelo, es un error gravísimo y degenera en la práctica de ir esparciendo patrones por las diferentes partes de un proyecto sin ningún criterio general.

Estos patrones no representarán ningún concepto de nuestro dominio, probablemente no reflejarán ningún significado a alguien que no esté familiarizado con nuestro proyecto, y dificultarán notablemente la extensibilidad de nuestro proyecto obligando a una refactorización continua.

Antes de abordar un proyecto con patrones, hemos de analizar minuciosamente qué patrones nos pueden ser útiles, cuáles son las relaciones entre los diferentes componentes de nuestro sistema, cómo podemos relacionar los patrones entre sí de modo que formen una estructura sólida, cuáles son los patrones que reflejan nuestro dominio, etc.

Cabe aclarar que cada patrón es en si un análisis del problema en cuestión y aporta una solución a dicho problema y vías de implementación.

Una integración efectiva se debe llevar a cabo mediante un análisis exhaustivo del problema en cuestión:

- Las soluciones documentadas anteriormente proveen, de una manera atómica, los criterios que se deben tener en cuenta en las integraciones de servicios.
- Los ejemplos ya probados, brindan información sobre las consecuencias y los beneficios de las integraciones.
- El análisis de lo que debe proveer la solución, se realiza estudiando en detalle el problema y que es lo que el mismo implica.
- El contexto donde se debe utilizar el problema es estudiado teniendo en cuenta el espacio conocido.

El las integraciones se debe tener en cuenta la cohesión y el acoplamiento, se debe lograr un equilibrio entre los mismos, se debe llegar a una integración una integración con una alta cohesión y un bajo acoplamiento. De este modo nos aseguramos que la solución de la integración propuesta podrá ser utilizada en otras integraciones similares.

La alta cohesión nos asegura que los servicios integrados serán utilizados en conjunto para resolver el problema. Es decir que cada uno por su lado tendrá el mismo objetivo. El bajo acoplamiento nos asegura que cada servicios es lo mas independiente posible del otro. Con lo cual el cambio de uno de ellos afecta, mínimamente, al otro.

# Capítulo VII

Experiencias de Uso

## **7. Experiencias de Uso**

En éste capitulo vemos con ejemplos basados en prototipos como son utilizados de manera efectiva algunos de los patrones vistos en el catálogo del capítulo 5.

## 7.1. Desarrollos

La experiencia de uso del catálogo está dada por la aplicación de dos patrones del mismo (Rendezvous y Chat Channel) en el desarrollo de aplicaciones prototípicas.

### 7.1.1. Rendezvous

El patrón rendezvous (Visto en 5.3.1) se utilizó en dos oportunidades. Se desarrolló un cliente de mail (7.1.2) y un cliente de news (7.1.2). En las dos herramientas de comunicación antes mencionadas, es posible ver, por medio del servicio de presencia de usuario, el estado de los usuarios (Online/Offline en éste caso particular) que se encuentran trabajando en un mismo espacio de trabajo colaborativo.

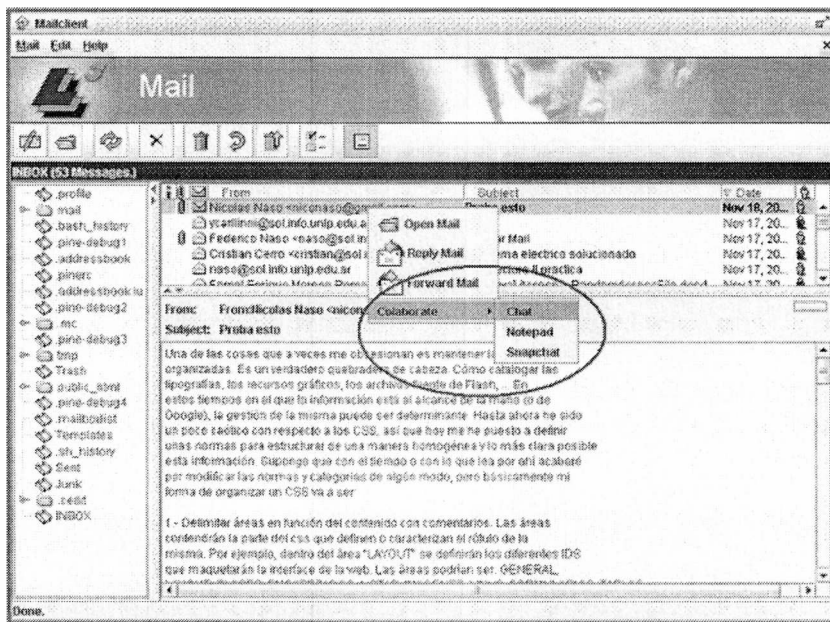


Figura 41: Colaboracion en el Mail

Tal como lo indica el patrón a través de un servicio asincrónico como es el cliente de mail o de news, es posible invitar a aquellos usuarios online a colaborar por medio de alguna herramienta de trabajo sincrónica (Figura 41), como lo son en éste caso el chat, el notepad y el snapchat, los cuales veremos en detalle más adelante.

En el ejemplo <Nicolas Naso> se encuentra online y se lo indica con un icono de una persona de color verde. Al hacer clic con el mouse, el menú contextual nos ofrece las herramientas de colaboración sincrónicas para trabajar en conjunto con el usuario.

Los participantes en la instanciación de éste patrón son:

- **Servicio de Groupware Asincrónico:** los servicios de groupware asincrónico están dados por los clientes de mail y de news.
- **Servicio de Groupware Sincrónico:** los servicios de groupware sincrónico están dados por el cliente de chat, el notepad colaborativo y el snapchat, los cuales veremos en detalle más adelante.
- **Servicio de Presencia:** el servicio de presencia es el que indica en el servicio de groupware asincrónico qué usuarios se encuentran online en ese momento para colaborar.

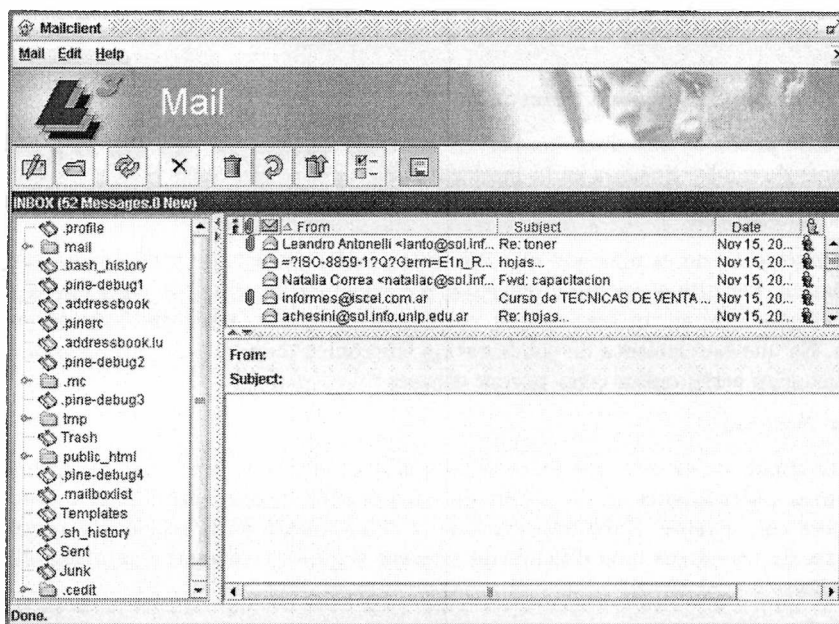


Figura 42: Mailclient

### 7.1.2. Los prototipos de Rendezvous

A continuación se ofrece una breve descripción de que consiste cada uno de los prototipos implementados. En el caso de los clientes de mail y de news, éstos implementan el patrón rendezvous.

#### Servicio de Groupware Asincrónico

- MailClient

Esta aplicación, *mailclient* (Figura 42) es una implementación de un cliente de mail estándar. Posee toda la funcionalidad para el envío y recepción de mail a usuarios. Este prototipo solo permite trabajar con una sola cuenta de mail por vez y utiliza el protocolo IMAP4 para poder ver los mails en el servidor.

- NewsClient

El *newsclient* es un cliente de news (Ver 4.1 Derivados del Mail) que permite ver y enviar news a grupos de noticias a los que estemos subscriptos. Utiliza el protocolo estandar para la lectura de los news (NNTP) como para el envío de los mismos (SMTP).

Tanto el cliente de mail como el cliente de news, integran el servicio de presencia de usuario que nos permite ver el estado de los mismos.

### Servicio de presencia de usuario

Para el servicio de presencia de usuario se utilizó el servicio provisto por el framework sobre el cual se desarrollaron las aplicaciones.

### Servicio de Groupware Sincrónico

A continuación se describen las herramientas sincrónicas que se integran a los clientes de mail y de news en la implementación del rendezvous.

- SimpleChat

El simple chat no es más que una implementación de chat de las más simples y básicas. Permite el envío de mensajes por parte de los usuarios y los mismos son visualizados en un área común por todos los participantes de la sesión de chat. Es una herramienta de colaboración sincrónica dado que es necesario que los usuarios estén online para poder chatear.

- Notepad

Un notepad, no es más que un sencillo editor de notas compartido, donde los usuarios participantes de la sesión de colaboración pueden escribir notas de manera concurrente. Cabe destacar que la colaboración en la edición de texto es una de las tareas más difíciles de resolver y aún se encuentra en etapa de estudio.

- SnapChat

SnapChat se verá dentro del patrón Chat Channel dado que además de ser una de las herramientas sincrónicas integradas en los clientes de mail y de news, es la implementación de dicho patrón.

### 7.1.3. Chat Channel

Por su parte, en la implementación del patrón Chat Channel (Visto en 5.3.4) se integró un servicio de Chat con un servicio de edición de documentos (Texto, Dibujo e Hipermedia) colaborativa, ésta aplicación nos es nada más ni menos que un *snapshot* el cual veremos con más detalle en el punto 7.1.4.

Los participantes en la instanciación de éste patrón son:

- **Aplicación Colaborativa:** la aplicación colaborativa en éste caso es un servicio de snap de trabajo grupal donde los participantes pueden realizar marcas en el contenido sobre el que se esté trabajando.
- **Chat:** éste participante está dado por un cliente de chat simple.



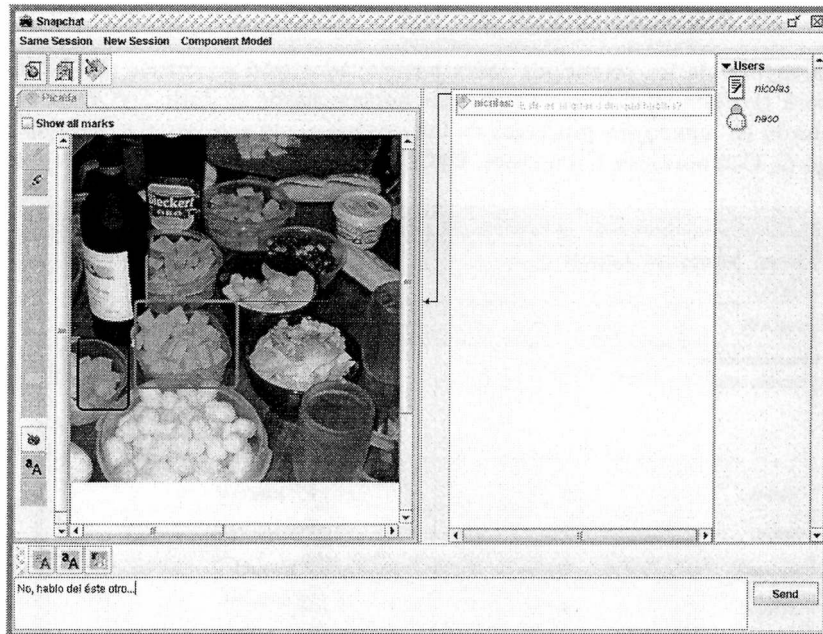


Figura 43: Snapchat

#### 7.1.4. Los prototipos de Chat Channel

- SnapChat

El Snapchat (Figura 43) no solo se integra como una herramienta sincrónica más, sino que además es una instancia del patrón Chat Channel. En ésta implementación se procede a integrar dos servicios groupware, el chat (Ver 4.2) como servicio de comunicación sincrónico y un espacio de edición (Ver 4.5) donde los participantes pueden mostrar y editar documentos.

La principal ventaja de tener asociada una herramienta de chat a la pizarra, radica en que se puede entablar una discusión sincrónica acerca del contenido de algún documento mostrado en la pizarra.

La discusión se realiza creando referencias sobre alguna parte del documento. Es decir, cuando un usuario desea realizar algún comentario de alguna imagen o párrafo mostrado en el documento, simplemente lo selecciona y activa el comando *marcar* donde podrá escribir y enviar un mensaje relacionado a la sección del documento marcada. El resto de los usuarios verá un nuevo mensaje en el Chat con una referencia al área del documento está haciendo referencia.

Todas las aplicaciones antes mencionadas fueron desarrolladas utilizando DyCE, la cual permitió el reuso de funcionalidad ya provistos por el framework tales como: registro e inicio de sesión de usuarios, ambiente de trabajo compartido, etc.

## 7.2. El Framework DyCE

El desarrollo de los prototipos ántes mencionados está soportado por el framework DyCE<sup>41</sup>, el cuál está totalmente implementado en Java. DyCE ha sido empleado en numerosos proyectos de investigación, para desarrollar un amplio rango de Componentes Groupware. DyCE resuelve una serie de puntos a tener

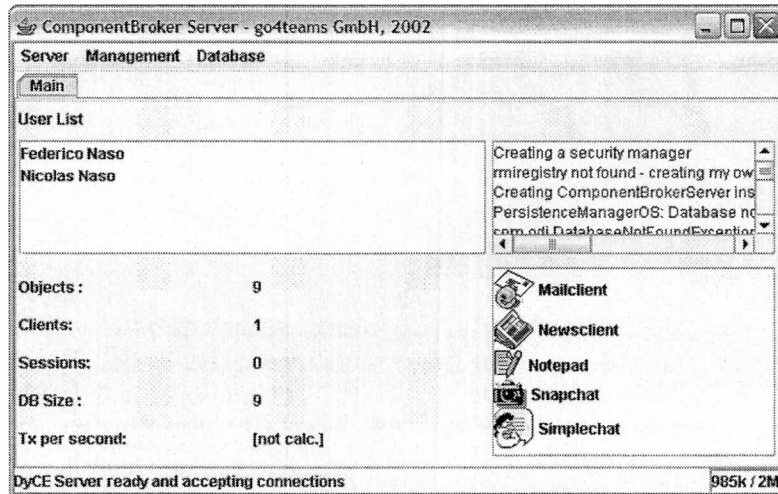


Figura 44: DyCE Component Server

en cuenta si se quiere desarrollar Componentes Groupware como las descritas anteriormente. Algunos de esos puntos son la replicación de los objetos compartidos, el manejo de sesiones de usuarios, la notificación de sucesos por medio de eventos, la persistencia de datos, resolución de conflictos, etc.

En la figura 44 podemos ver parte de cómo el servidor tiene registrados algunos componentes groupware y algunos usuarios.

### 7.2.1. ¿Por que usar un framework?

Un framework es una aplicación semicompleta reusable, que puede ser especializado para producir aplicaciones personalizadas. Los beneficios primarios de un framework son la modularidad, reusabilidad, extensibilidad, e inversión de control que ellos proporcionan desarrolladores. Los frameworks refuerzan la modularidad encapsulando los detalles de implementación.

La modularidad de los frameworks ayudan a mejorar la calidad del software, localizando el impacto en los cambios de diseño e implementación, que reducen el esfuerzo que se requerido para entender y mantener el software existente. En nuestro caso se utiliza un framework para poder abstraerse de los problemas de replicación de objetos y resolución de los potenciales conflictos que suelen

<sup>41</sup>DyCE: Dynamic y Collaboration Environment).

ocurrir en las aplicaciones groupware y poder así concentrarse en la integración de las mismas.

---

# Capítulo VIII

Conclusiones

## **8. Conclusión**

En este capítulo se ven los trabajos relacionados de ésta tesis, cómo se diferencia dicha tesis con estos trabajos. Se hace una revisión de la contribución y se plantea el trabajo a futuro.

### 8.1. Alcance del catálogo

En realidad en las aplicaciones groupware podemos encontrar muchas otras integraciones interesantes de servicios. Como por ejemplo: calendarios con servicios de votación que sirven para votar que día se desea gestar una reunión entre personas entre varias opciones a elegir. Con el servicio de votación los usuarios eligen una de las opciones de un poll. Pero el alcance de ésta tesis no es hacer un estudio de todas las integraciones existentes, sino hacer hincapié solo en las integraciones más frecuentes de servicios groupware. Y de ellas armar un catálogo inicial de patrones de integración. Este catálogo sirve como punto de partida para llevar a cabo nuevas integraciones.

### 8.2. Integraciones a medida

Existen integración de servicios groupware ad-hoc que, si bien son muy interesantes, no respetan la *regla de tres* con lo cual no se pueden considerar patrones ni ser parte de ningún catálogo de patrones. Un ejemplo son los sistemas de workflow, específicos para resolver cierto conjunto de tareas. Podemos encontrar workflows integrados a calendarios donde hay tareas que se deben ejecutar en ciertas fechas. O workflows donde vía FTP guardan documentos en un servicio de CMS. Pero son implementaciones muy específicas llevadas a cabo con un fin muy particular.

### 8.3. Acoplamiento de servicios

En muchas aplicaciones groupware, distintos servicios se integran de manera tan estrecha que no permiten la evolución de los mismos por separado. Por ejemplo, la integración de awarenes de acción (que hace cada usuario) se mezcla con la implementación de la funcionalidad principal del sistema. (Un ejemplo de esto es la edición colaborativa de diagramas UML). El uso de patrones de integración motiva y asiste al desarrollo de groupware como la integración de servicios autónomos.

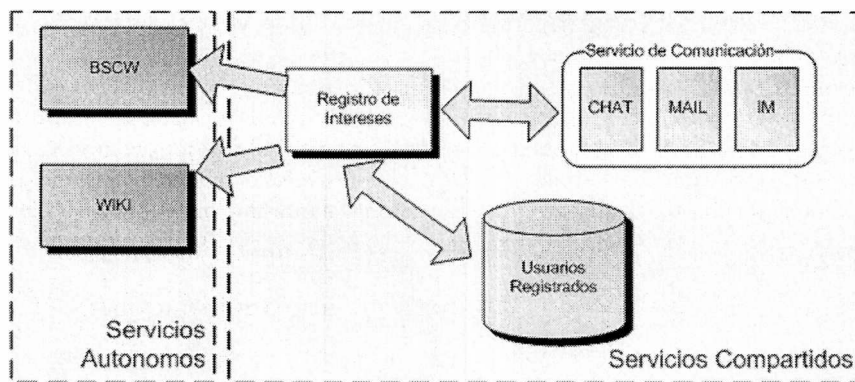


Figura 45: Servicios Autonomos

En cambio cuando los servicios no están integrados de manera estrecha, es decir,

cuando contamos con servicio autónomos, los mismos pueden ser reutilizados en múltiples integraciones. Por ejemplo en la Figura 45 podemos ver que un servicio de manejo de contenido (BSCW o Wiki) utilizan el mismo servicio para notificar (mediante e-mail) los eventos que se estén llevando a cabo (Por ejemplo: "Una nueva versión de un documento fue cargada al sistema") según el registro de interesados que éste posea.

#### 8.4. Trabajos relacionados

Existen algunos trabajos relacionados en lo que concierne a la investigación y la documentación y armado de catálogos de patrones de groupware.

Uno de ellos es el trabajo que se está realizando en Alemania a cargo de Till Schümmer para su PHD, denominado *patterns 4 groupware* [22]. En este proyecto la idea es documentar y armar un catálogo de componentes o conceptos que comúnmente son utilizadas para el trabajo cooperativo Ej.: repositorios, userlist, calendarios, rooms, telepointer, floor control, etc. y clasificar a las mismas en distintas familias. Ej.: comunicación, awareness, protocolo, Roles, etc. De ésta forma cada vez que se necesite desarrollar alguna aplicación cooperativa, con la ayuda de este catálogo, se facilita la tarea de tener que seleccionar qué componentes o qué concepto se deben tener en cuenta para implementar en la solución que se desea llevar a cabo.

El trabajo de Schümmer está orientado a la búsqueda de patrones de componentes o conceptos groupware (como por ejemplo el floor control), no a la *integración de servicios* como se lo describe en esta tesis. Trata a las componentes como partes separadas que proveen funcionalidad propia y que son parte de una aplicación más grande con el fin de llegar a una solución específica.

Así por ejemplo el patrón de integración *rendezvous* (visto en el capítulo 5) utiliza las componentes de userlist, presence indicador, awarenes, instant messaging, y electronic mail que se especifican en el catalogo de Schümmer.

En [22] se puede ver un approach del catálogo. En la figura (Figura 46) se puede ver un mapa del mismo en donde se muestran las componentes groupware, las relaciones entre las mismas y la división en familias. Cada familia agrupa patrones orientados a resolver problemas en áreas similares.

Otro de los trabajos es el que realizó Robert Slagter para su PHD [70]. Slagter, propone una estructuración de servicios groupware en componentes modulares llamados GSM (Groupware Service Modules). Esta estructura permite la cooperación entre usuarios seleccionando módulos de componentes groupware.

Los componentes modulares son la base fundamental de las aplicaciones groupware, entonces utilizando la idea de Slagter, las aplicaciones groupware son diseñadas y desarrolladas mediante estos componentes modulares, los mismos se componen en módulos más grandes que brinda mayor funcionalidad y permiten tener soluciones más específicas según el contexto deseado. Estos componentes son parte de un framework desarrollado en C#.

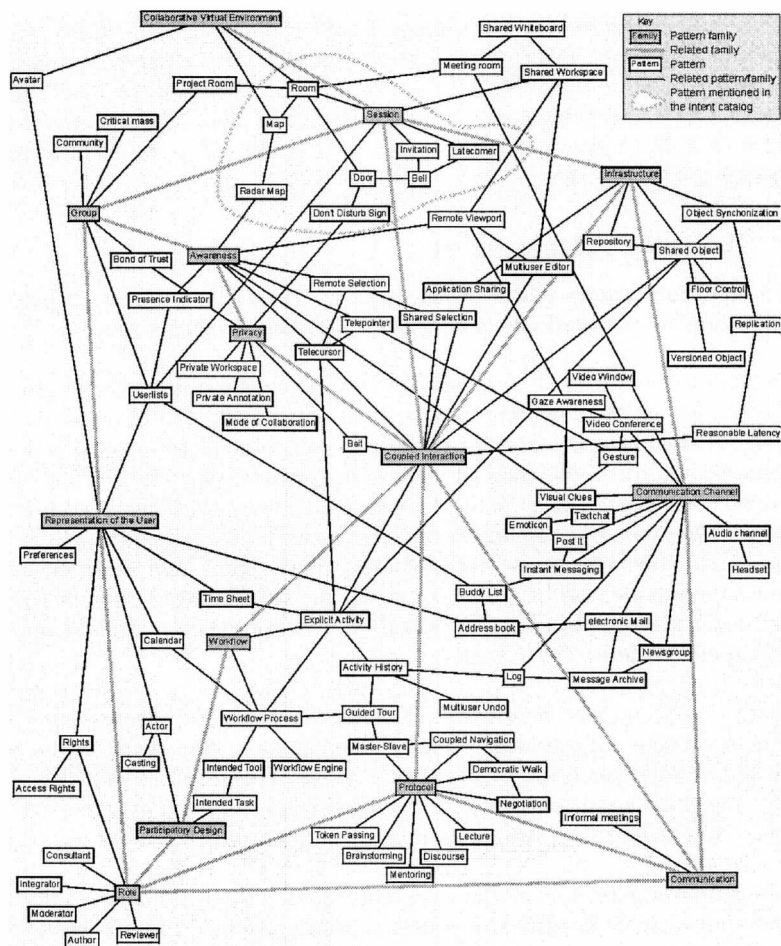


Figura 46: Mapa de Patrones

En los dos casos anteriores los trabajos planteados si bien están relacionados a ésta tesis, se enfocan en distintos aspectos. En el primer caso son patrones sobre qué servicios se debe utilizar según el contexto. Y en segundo caso, se enfoca en componentes como cajas negras que pueden componer. Por otro lado estos trabajos son trabajo de un mayor alcance dado que son trabajos para tesis doctorales.

### 8.5. Validación del catálogo

Le escritura del catálogo fue validada tomando uno de los patrones de integración (*rendezvous*) y enviando dicho patrón a uno de los congresos más importantes sobre patrones (PloP 2004) [24]. Se utilizó para la escritura y validación de dicho patrón la metodología de *Shepherding* [29] que es la única forma existente de validar la calidad de los mismos. Esta metodología es una suerte de revisión



por medio de personas especializadas que brindan consejos al autor. Luego el autor vuelca los consejos de los *Shepherds* al patrón, lo cual conlleva a aumentar la calidad de patrón. Es este caso los *Shepherds* asignados a esta patrón fueron *Doug Lee* y *Joe Yoder* dos integrantes activos de la comunidad de patrones. Este procedimiento se lleva a cabo un par de veces sobre el mismo patrón y luego se discute el patrón en un *workshop* de escritores, donde es evaluada la calidad del mismo. Por una cuestión de tiempo solo se sometió uno de los patrones [60] a éste proceso y para la escritura del resto se tomó éste como punto de partida.

## 8.6. Revisión de la contribución

Haciendo un balance del contenido de esta tesis, de la contribución planteada, y del trabajo relacionado, podemos observar que se logra contribuir con:

- Una revisión del estado del arte de los servicios groupware actualmente utilizados y aplicaciones que implementan tales servicios.
- Un catálogo que presenta el estado del arte en integración de servicios de groupware. Como así también los usos que se le dan y las aplicaciones más conocidas que los implementan.
- Para cada forma de integración conocida, se documentó un método de integración de efectividad estudiada y probada. Esto se consiguió gracias a que los patrones, por definición, documentan soluciones que han sido efectivamente implementadas.
- Validación del catálogo por medio de la metodología de Sheperding. La validación de la escritura y calidad de los patrones está dada porque los mismos fueron sometidos a dicho proceso.
- Un conjunto de prototipos de muestran la aplicación de los patrones de integración.

### 8.6.1. Contribución extra

Además, y como contribución extra, que no estaba planificada en la introducción se logró:

- Una estructura o plantilla para documentar los patrones de integración de servicios basada en las que se utilizan en los patrones de diseños pero donde se introducen algunas mejoras que ayuda a entender y potenciar la integración de los servicios groupware. Esta plantilla puede ser reutilizada en la documentación de nuevas integraciones de servicios.

## 8.7. Trabajo Futuro

Como trabajo futuro se está pensando en dos cosas a tener en cuenta:

- La extensión del catálogo de patrones de integración de servicios. Ya que quedó demostrado que es de gran ayuda a la hora de tener que diseñar una aplicación colaborativa. Los patrones aquí catalogados son solo los patrones básicos e iniciales que existen en la actualidad. El trabajo y por

ende el desarrollo de aplicaciones groupware es cada vez mas amplio con lo cual los patrones irán apareciendo a la par del crecimiento de estas aplicaciones. Esto trae aparejado el crecimiento del catálogo cada vez que se documenta un nuevo patrón.

- El desarrollo de una aplicación, orientada al Web para facilitar la documentación, el estudio y el catalogado de los patrones. Es decir alguna aplicación que facilite la búsqueda y la documentación de nuevos patrones. Dicha aplicación debe soportar colaboración, con lo cual sería una aplicación del estilo CoWeb como las vistas en el *capítulo 5*.
- Validar el resto de los patrones del catálogo tal como se validó el patrón *rendezvous*. Para ello se pretende enviar el catálogo a uno de los congresos específicos de este tema. En [12] se pueden encontrar una lista de congresos especializados en patrones.
- Compartir el catálogo con una comunidad en donde se estudien patrones groupware. Una de esas comunidades es la que formó Till Schümmer y se llama *patterns 4 groupware* [22]. En este proyecto la idea es documentar y armar un catálogo de componentes o conceptos que comúnmente son utilizadas para el trabajo cooperativo.

# Capítulo IX

Referencias

## Referencias

- [1] Ants. <http://ants.etse.urv.es>. Last accessed: Jul/07/2005.
- [2] Arroz a banda. <http://www.arrozabanda.net/>. Last accessed: 07/07/2005.
- [3] Bscw. [www.fit.org](http://www.fit.org). Last accessed: Jul/07/2005.
- [4] Coast (cooperative application system toolkit). [www.opencoast.org](http://www.opencoast.org). Last accessed: 07/07/2005.
- [5] Conversationboard. <http://www.crew.umich.edu/brinck/CSCW92.html>. Last accessed: Jul/07/2005.
- [6] Coweb. <http://minnow.cc.gatech.edu/edutech/14>. Last accessed: Jul/07/2005.
- [7] Edmz. <http://www.edmz.org>. Last accessed: 07/07/2005.
- [8] Eyeball. [www.eyeball.com/](http://www.eyeball.com/). Last accessed: Jul/07/2005.
- [9] Groupkit. <http://www.cpsc.ucalgary.ca/grouplab/projects/GroupKit.html>. Last accessed: Jul/07/2005.
- [10] Groupwise. <http://www.novell.com/products/groupwise/>. Last accessed: Jul/07/2005.
- [11] Habanero. <http://www.isrl.uiuc.edu/isaac/Habanero>. Last accessed: Jul/07/2005.
- [12] Hillside. <http://hillside.net/>. Last accessed: Jul/07/2005.
- [13] Kent beck. <http://c2.com/cgi/wiki?KentBeck>. Last accessed: Jul/07/2005.
- [14] Lotus notes. <http://www.lotus.com>. Last accessed: Jul/07/2005.
- [15] Lotus sametime. <http://www.lotus.com>. Last accessed: Jul/07/2005.
- [16] Micro workflow. <http://micro-workflow.com/>. Last accessed: Jul/07/2005.
- [17] Microsoft sharepoint portal server. <http://www.microsoft.com/cmsserver>. Last accessed: Jul/07/2005.
- [18] Ms exchange server. <http://www.microsoft.com/exchange/default.asp>. Last accessed: Jul/07/2005.
- [19] Ms outlook. <http://www.microsoft.com/outlook/>. Last accessed: Jul/07/2005.
- [20] Netscape. <http://www.netscape.com/>. Last accessed: Jul/07/2005.
- [21] Nota diario clarin 15/10/2004. <http://www.clarin.com/diario/2004/10/15/conexiones/t-849925.htm>. Last accessed: Jul/07/2005.

- [22] Patterns for groupware. <http://www.groupware-patterns.org/>. Last accessed: Jul/07/2005.
- [23] Pequeños enigmas. <http://markelo.f2o.org/>. Last accessed: 07/07/2005.
- [24] Plop 2004. <http://hillside.net/plop/2004/>. Last accessed: Jul/07/2005.
- [25] polls4free. <http://polls4free.com/pollgens.html>. Last accessed: Jul/07/2005.
- [26] Presence awareness. <http://www.ietf.org/rfc/rfc2778.txt>. Last accessed: Jul/07/2005.
- [27] Ray tomlinson. <http://openmap.bbn.com/tomlinso/ray/firstemailframe.html>. Last accessed: Jul/07/2005.
- [28] Rfc822. <ftp://ftp.rfc-editor.org/in-notes/rfc822.txt>. Last accessed: Jul/07/2005.
- [29] Shepherding. <http://hillside.net/shepherding.html>. Last accessed: Jul/07/2005.
- [30] Smalltalk. <http://www.smalltalk.org/>. Last accessed: Jul/07/2005.
- [31] Talk. <http://www.bris.ac.uk/is/selfhelp/documentation/talk-i1/talk-i1.htm>. Last accessed: Jul/07/2005.
- [32] Tikiwiki. <http://tikiwiki.org/TikiWiki>. Last accessed: Jul/07/2005.
- [33] Unix. <http://www.bell-labs.com/>. Last accessed: Jul/07/2005.
- [34] Ward cunningham. <http://c2.com/cgi/wiki?WardCunningham>. Last accessed: Jul/07/2005.
- [35] Writers' workshop groups. <http://hillside.net/plop/2004/workshop.html>. Last accessed: Jul/07/2005.
- [36] Vissers C. A., Ferreira Pires L., Quartel D. A. C., and Van Sinderen M. J. *The architectural design of distributed systems: Reader for the design of Telematics Systems*. Enschede, The Netherlands: University of Twente., 2002.
- [37] Aim. <http://www.aim.com>. Last accessed: March/15/2004.
- [38] Christopher Alexander. *The Timeless Way of Building*. Oxford University Press, New York. USA, 1979.
- [39] Christopher Alexander, Sara Ishikawa, and Murray Silverstein. *A Pattern Language : Towns, Buildings, Construction*. Oxford University Press, New York. USA, 1977.
- [40] Peter Coad, D. North, and M. Mayfield. *Object Models: strategies, patterns and applications*. Prentice Hall, Englewood Cliffs, 1995.

- [41] C. A. Ellis and S. J. Gibbs. Concurrency control in groupware systems. In *SIGMOD '89: Proceedings of the 1989 ACM SIGMOD international conference on Management of data*, pages 399–407, New York, NY, USA, 1989. ACM Press.
- [42] Clarence A. Ellis, Simon J. Gibbs, and Gail Rein. Groupware: some issues and experiences. *Communications of the ACM*, Volume 34 , Issue 1:39–58, 1991.
- [43] Amos Fiat, Richard M. Karp, Michael Luby, Lyle A. McGeoch, Daniel D. Sleator, and Neal E. Young. Competitive paging algorithms. *J. Algorithms*, 12(4):685–699, 1991.
- [44] M. Fowler. *Analysis Patterns: Reusable Object Models*. Addison-Wesley, 1997.
- [45] Erich Gamma, Richard Helm, Ralph Johnson, and John Vlissides. *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley, Toronto, Ontario. Canada, 1995.
- [46] Fernando D. Garcia, Diego De Sogos, Federico Naso, Ricardo Tesoriero, and Alejandro Fernández. Un framework para la construcción de aplicaciones de comunicación textual sincrónica. In *Proceedings of VII Congreso Argentino de Ciencias de la Computación - El Calafate 16-10 Jan. 2001*. CACIC, 2001.
- [47] M. Roseman D. Webster Greenberg, S. and R. Bohnet. Human and technical factors of distributed group drawing tools. *Interacting with computers*, 1(4):364–392, 1992.
- [48] Jonathan Grudin. Why csw applications fail: Problems in the design and evaluation of organizational interfaces. In *CSCW '88, Proceedings of the conference on Computer-supported cooperative work*, pages 85–93, New York, 1988. ACM Press.
- [49] Jonathan Grudin. Groupware and social dynamics: eight challenges for developers. *Commun. ACM*, 37(1):92–105, 1994.
- [50] Carl Gutwin and Saul Greenberg. Workspace awareness for groupware. In *CHI '96: Conference companion on Human factors in computing systems*, pages 208–209, New York, NY, USA, 1996. ACM Press.
- [51] P. Schulthess H. Christein. Presence awareness. A general Purpose Model for Preceence Awareness.
- [52] Thomas Herrmann, Marcel Hoffmann, Isa Jahnke, Andrea Kienle, Gabriele Kunau, Kai-Uwe Loser, and Natalja Menold. Concepts for usable patterns of groupware applications[39] (abstract only). *SIGGROUP Bull.*, 24(1):20–20, 2003.
- [53] Henri Ter Hofte. *Working Apart Together - Foundations For Components Groupware*. Telemática Institut, Enschede, The Netherlands, 1998.
- [54] Icq. <http://www.icq.com>. Last accessed: March/15/2004.

- [55] Jabber. <http://www.jabber.org/>. Last accessed: March/15/2004.
- [56] Robert Johansen. *GroupWare: Computer Support for Business Teams*. The Free Press, Macmillan Inc., New York, November 1988.
- [57] Yann Laurillau and Laurence Nigay. Clover architecture for groupware. In *CSCW '02: Proceedings of the 2002 ACM conference on Computer supported cooperative work*, pages 236-245, New York, NY, USA, 2002. ACM Press.
- [58] Segerstad Ljungstrand. Awareness of presence, instant messaging and webwho. *SIGGROUP Bulletin Dic2000*, (12), 2000.
- [59] Msn messenger. <http://messenger.msn.com/>. Last accessed: March/15/2004.
- [60] Federico Naso and Alejandro Fernández. Rendezvous - a groupware pattern. 2004.
- [61] Federico Naso and Alejandro Fernández. Patrones de integración de servicios groupware. page 60, Cali, Colombia, 2005. XXXI conferencia latinoamericana de Informatica (CLEI) 10-14 de octubre de 2005.
- [62] Netmeeting. <http://www.microsoft.com/windows/netmeeting/>. Last accessed: March/15/2004.
- [63] Coad Peter. Object-oriented patterns. *Communications of the ACM*, 35(9):152-159, 1992.
- [64] Steven Poltrock and Jonathan Grudin. Computer supported cooperative work and groupware. In *CHI '94: Conference companion on Human factors in computing systems*, pages 355-356, New York, NY, USA, 1994. ACM Press.
- [65] Steven Poltrock and Jonathan Grudin. Cscw, groupware and workflow: experiences, state of art, and future trends. In *CHI '99: CHI '99 extended abstracts on Human factors in computing systems*, pages 120-121, New York, NY, USA, 1999. ACM Press.
- [66] Pownow. [www.pownow.com](http://www.pownow.com). Last accessed: March/15/2004.
- [67] Linda Rising. Patterns: A way to reuse expertise by linda rising. *AG Communications System(AGCS)*, 37(4), 1999.
- [68] M. Roseman and S Greenberg. Simplifying component development in an integrated groupware environment. 1997.
- [69] Mark Roseman and Saul Greenberg. Teamrooms: Network places for collaboration. In *Computer Supported Cooperative Work*, pages 325-333, 1996.
- [70] Robert Slagter. *Dynamic Groupware Services - Modular Design of Tailorable Groupware*. PhD thesis, University of Twente, 2004.
- [71] Swiki swiki. <http://minnow.cc.gatech.edu/swiki>. Last accessed: March/2003.

- [72] Daniel Tietze. *A Framework for Developing Component-based Co-Operative Applications*. GMD Research Series, 2001. Dissertation at Darmstadt University of Technology.
- [73] Dr. Daniel A. Tietze. Enterprise instant messaging - the dyce instant messenger solution. *go4teams GmbH, Darmstadt*, 2002.
- [74] Webinfo. <http://webinfo.info.unlp.edu.ar/>. 01/11/2005.
- [75] J.M.; Tietze D.A Wessner, M.; Haake. An infrastructure for collaborative lifelong learning. In *Proceedings of the 35th Annual Hawaii International Conference on 7-10 Jan. 2002*, page 9. HICSS, 2002.
- [76] B Wilson. Wscrawl 2.0: A shared whiteboard based on x-windows. In S. Hayne S. Greenberg and R. Rada, editors, *Designing Groupware for Real Time Drawing*. McGraw Hill,, 1994.
- [77] Yahoo messenger. <http://messenger.yahoo.com/>. Last accessed: March/15/2004.



# Agradecimientos

## 9. Agradecimientos

En principio quisiera agradecer a las dos personas que hicieron posible mi vida, mi papá Ricardo y mi mamá Cristina. Sin todo el esfuerzo que ellos hicieron, hacen y se que harán por mi, el apoyo, contención, educación y amor de ellos jamás hubiera logrado nada en la vida. Ellos me enseñaron a ser como soy, ellos me hicieron lo que soy, me dieron todo y más en la vida. A ellos. Gracias Viejos!!!

A mi hermano Nicolás, que digo hermano si más que un hermano es un amigo. Solo por ser él, por estar a mi lado incondicionalmente, por aguantarme los buenos días que a veces no son mucho mejorer que los días que son para olvidar. Porque con su rectitud inquebrantable a veces casi tozuda, me ha demostrado que si uno esta seguro de las cosas las tiene que defender a como dé lugar.

Al LIFIA como un todo, por brindarme la posibilidad de hacer mi experiencia, de trabajar en lo que me gusta, por abrirme sus puertas y su corazón. Por marcarme el camino. Y lo más importante... a su gente, que es como mi segunda familia.

Alicia, quien accedió a dirigir ésta tesis, sin saber pobre en lo que se estaba metiendo. Pobre... :)

Casco, codirector de ésta tesis, mi mentor, mi guía, quien me abrió las puertas al LIFIA y me tendió su mano amiga enseñándome a dar los primeros pasos en lo que espero sea mi futuro de aquí en más. El fué quien confió en mi aun cuando yo no tenia ni la experiencia ni la edad suficiente como para hacer determinadas cosas que es el día de hoy no se como las logre hacer. Gracias Dr.!

A Gustavo, porque me brindo la oportunidad de crecer. Siempre que lo necesité, él estuvo y me dio consejos que hasta el día de hoy recuerdo.

Diego (Mamin) De Sogos, no tengo palabras para agradecerle, el fue quien estuvo a mi lado en las buenas y en las malas. Un amigo de esos que no se consiguen a la vuelta de la esquina. El me ayudo los primeros años en La Plata y si no fuera por él hoy no estaría escribiendo esta tesis.

Coto, porque con sus cambios de humor repentinos (más repentinos y más cambios que los míos), con su gracia y con su forma de ser aporta a mi vida la cuota diaria de alegría. Diego, gracias por ser vos, siempre.

Lea, otro de los amigos que no se van a olvidar fácil. Porque con el y coto, compartimos muchas de las etapas tanto buenas, las cuales me tengo muchos y variados recuerdos y a veces no tan buenas, algo difíciles, sin saber que nos depararía el destino.

Juan, que se puede decir de Juan que nadie sepa? Solo hay que agradecerle por ser él. Por su buen humor constante y su forma de tomarse las cosas en la vida que a veces me gustaría que me diga como se hace.

Richard, por compartir mis inicios en el LIFIA, porque junto a él nos hemos

roto la cabeza y puteado más de una vez con ciertos algoritmos de Smalltak. Pero que siempre, pero siempre conseguía una solución al problema.

Pipa, no me acuerdo bien cuando nos conocimos, fue hace mucho tiempo ya, pero los años pasan y sigue siendo una de las personas que más aprecio le tengo. Se que a veces no se lo digo pero más de una vez me ha sorprendido y creo que a todos con sus logros, la verdad que por ser como es se merece todo lo que logro, logra y logrará.

Diego Mayeux, por ser él, el autentico, un poco colgado, un poco delirante, un poco amigo, un poco hermano. Por las noches de charlas, birra, mate, o solo palabras de por medio, por supuesto, que se extienden en algunos casos hasta el amanecer. Gracias pelado.

Alejandra, por ser la persona que con solo una mirada ya sabe que es lo que me pasa, por entenderme y por escucharme. Por ser un poco madre, un poco hermana, un poco amiga, por ser ella. Ale, no tengo muchas palabras para expresar lo que te quiero. Creo que ya nos conocemos y sabemos como somos una mirada sobra.

Anabella, por todas las veces que la moleste con pedidos y cosas que necesitaba y ella supo otorgar con santa paciencia.

A Lanto por las veces que me depositó en mi humilde Junín natal, por las charlas de vida en cada viaje. Por las veces que me instruyo cuando lo atormentaba (y no crean que ya no lo hago más) a preguntas.

A Analia que me brindo el cobijo de su hogar en un par de oportunidades. Que sin conocerme me trato como un amigo de toda la vida y me abrió los brazos de su familia.

Julieta, por ser la hermana que nunca tuve. Por las cosas que hemos vivido juntos y porque más de una vez sus palabras fueron fuente de inspiración para mi. Una luz más en mi vida. Por entenderme, solo por ser ella casi una hermana.

A Octavio que en poco tiempo me brindo su amistad, lo aprecio un montón, como si nos conociéramos de toda la vida.

Monte, es uno de esos amigos que vale la pena tener y cuidar. Se merece todo lo que tiene y logro en la vidaes un luchador y le doy gracias que me deje ser su amigo.

Diego (Mono), por las noches de parranda por los pubs y diversas recorridas. Por hacerme el aguante esos días que hay ganas de divertirse y no hay mucho quórum.

Diego Barzola, porque en el poco tiempo que llevo de conocerlo me ha demostrado ser un amigo de fierro, incondicional y que se preocupa por mí. Él está en el momento oportuno, no habla mucho ? pero dice las palabras justas. Él

siempre esta, para las buenas y para las malas. Gracias diego por ser mi amigo.

Pablo, otro de los amigos que hace pila de años que lo son, hace mas tiempo que somos amigos que tiempo que tiempo que no lo somos. Tengo una bocha de recuerdos de cosas que hicimos juntos.

Al Seba, que cada vez que voy a Junín me ceba mates con pepas y me pone al tanto de las ultimas novedades del grupo de amigos. Y que nunca me hace ganar un peso a la quiniela!

Cristian, otro de mis amigos de hace años, tiene una forma tan particular de ser que admiro.

Nahuel mi dentista sin matricula :) , por mantener a raya mi terrible bruxismo en los peores momentos de stress al escribir ésta tesis. Se que va a ser el mejor dentista que pueda haber, ya lo he comprobado.

El resto de mis amigos de Junín, que seguro me voy a olvidar de alguno y por no cometer semejante error voy a preferir meterlos en la misma bolsa a todos gracias a todos ellos amigos.

A Juan, "Bichito de Luz", que me roba las IPs dinamicas de la notebook pero me las devuelve ;) que en poco tiempo se gano mi respeto, mi amistad y mi admiración.

A Yane, por integrarse rapidamente en mi vida y por brindarme su amistad. Por estar y por ser ella.

A la gente del CURJ (Sandra, Maria, Javier, y muchísimos amigos más), porque es el lugar donde yo encontré mi destino, es el lugar donde hice mis primeras armas y que hasta hoy llevo en mi corazón.

A mis tíos de sangre, Graciela, Susana y José (cheche) y los políticos, que no por ser políticos son menos tíos, Rubén, Abrahán y Susana.

Natalia, la prima que me enteré que tenia de grande y que me hubiera gustado conocer mucho antes, pero que de todos modos nos entendimos como si el tiempo de conocernos fuera de toda la vida. Por las cosas que hizo por mi y porque encontré un poco de mi sangre familiar derramada y perdida por la vida y que juntos la supimos recuperar. Sos como mi otra hermana.

Agustina, otras de mis primas. Por hacer a veces de hermana menor, por canfiarme algunas cosas como hermano mayor.

A mi madrina laura que siempre se acordó de mí en todo momento y siempre está presente.

A los shepers que mejoraron con su aporte y su santa paciencia el contenido de los patrones que integran el catalogo.

Al negro, porque me hizo entender y reflexionar sobre muchas cosas de la vida.  
Lastima que nos abandono demasiado pronto.

A todos aquellos, que de seguro me estoy olvidando por una u otra razón, a los que estuvieron, los que están a su modo y a los que no, a los incondicionales, a los del alma, a los que estarán, a los que vienen en camino, a todos... como dijo un grande: "GRACIAS TOTALES!!!"



**BIBLIOTECA  
FAC. DE INFORMÁTICA  
U.N.L.P.**

Hoy en día es común encontrar aplicaciones las cuales integran servicios groupware. Un ejemplo de esto son las pizarras compartidas que integran Chat y dibujo colaborativo en una misma aplicación groupware.

Muchos de estos servicios groupware se pueden llegar a combinar brindando nuevos tipos de servicios de más alto nivel, que inter-operen entre si para brindar nueva funcionalidad. Consecuentemente este nuevo servicio integrado será más potente que la suma de las partes.

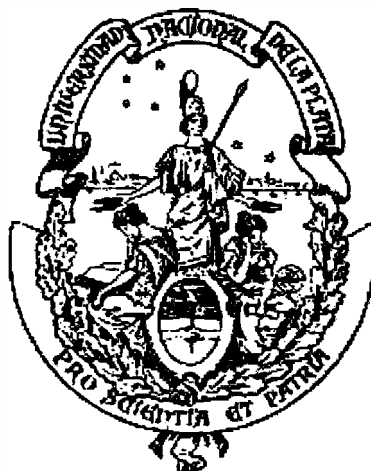
En lugar de diseñar una solución groupware en función de los servicios individuales que brinda, se la diseña como un todo, como un ambiente integral que ofrece una plataforma de colaboración a grupos de trabajo que se encuentran remotamente distribuidos y llevan a cabo interacciones electrónicas.

La integración de servicios groupware se lleva a cabo por medio de ciertos aspectos que son "comunes" a los servicios que se integran. Esos aspectos son el punto de unión entre los distintos servicios. Dado que la integración de servicios groupware está típicamente orientada a la implementación se plantea un problema a causa de que los servicios existentes utilizan tecnologías muy heterogéneas y cada vez que se necesite una solución, la misma, se la deba programar de cero. Como resultado, se desvía fuerza de trabajo a aspectos de implementación, cuando sería más importante dedicarlos a la concepción y diseño de la integración de los servicios.

La falta de asistencia en el diseño de integraciones de servicios groupware, ejemplos y arquitecturas que sirvan de guía y ayuden a realizar integraciones efectivas, conlleva a que cada vez que alguien necesite realizar una integración, la misma, se lleve a cabo desde cero. Con lo cual los problemas que ya fueron resueltos por otras personas previamente, se vuelvan a experimentar una y otra vez.

Una forma de abstraerse de los problemas inherentes de los diseños orientados a la implementación, es la utilización de patrones de diseño de software. Los mismos documentan soluciones ya probadas por medio de arquitecturas mínimas que luego pueden ser implementadas por medio de las tecnologías que se seleccionen.

Teniendo en cuenta lo planteado anteriormente, el objetivo de esta tesis es el estudio de los servicios groupware existentes, el desarrollo y la compilación de un catálogo de patrones de servicios groupware y de patrones de integración de servicios groupware.



Facultad de Informática,  
Universidad Nacional de La Plata  
La Plata, Viernes, 25 de Noviembre de 2005

TES  
05/23  
DIF-02955  
SALA



UNIVERSIDAD NACIONAL DE LA PLATA  
Biblioteca  
50 y 120 La Plata  
catologo.info.unlp.edu.ar  
biblioteca@info.unlp.edu.ar



DIF-02955