



UNIVERSIDAD VERACRUZANA

FACULTAD DE ESTADÍSTICA E INFORMÁTICA

PROTOCOLO DEL TRABAJO:

“DISEÑO E IMPLEMENTACIÓN DE UNA ARQUITECTURA
CONSCIENTE DEL CONTEXTO PARA SISTEMAS
GROUPWARE”

MODALIDAD:

TRABAJO PRÁCTICO TÉCNICO

COMO REQUISITO PARCIAL PARA OBTENER EL GRADO DE:

LICENCIADO EN INFORMÁTICA

PRESENTA:

SANTIAGO DE JESÚS GONZÁLEZ MEDELLÍN

DIRECTORES:

MCA. LUIS GERARDO MONTANÉ JIMÉNEZ

DR. EDGARD IVÁN BENÍTEZ GUERRERO

XALAPA, VER. JUNIO 2014

0.1. Resumen

El trabajo colaborativo es una actividad compleja debido a que un grupo de personas tienen que unir sus esfuerzos para poder llevar a cabo un objetivo realizando tareas que en ocasiones dependen de otras realizadas por otros integrantes del grupo. El trabajo colaborativo asistido por computadora (CSCW) ¹ es el área que estudia los sistemas computacionales que ayudan a los grupos de trabajo a mejorar la coordinación, cooperación y comunicación que hay entre los integrantes del grupo de trabajo.

Entre los sistemas que estudia el CSCW están los Groupware que son sistemas que funcionan como medio para que los usuarios interactúen entre sí y puedan llevar a cabo tareas en conjunto. Para facilitar la interacción de estos usuarios con el sistema y con otros usuarios necesitan tener un grado aceptable de consciencia de la situación que los rodea a ellos como individuos, de los miembros del grupo del que forman parte y al estado de las tareas que se están llevando a cabo. Debido a esto surge la necesidad de dotar a los sistemas con el conocimiento necesario para poder proporcionar tales niveles de consciencia al usuario a partir de esta necesidad se desarrollan sistemas conscientes del contexto los cuales tienen conocimiento de la situación del ambiente en el que operan, así, los Sistemas Groupware Conscientes del Contexto ² tienen ambas características antes descritas: apoyan el trabajo colaborativo de un grupo de usuarios y además conoce el contexto que los rodea y opera en consecuencia a esta información, para eso se crean arquitecturas conscientes del contexto que puedan ser integradas a un groupware y que entre ambos sistemas puedan funcionar como un CAGS.

Actualmente la mayoría de las arquitecturas elaboradas están orientadas a groupwares con ambientes específicos, es decir, usan el contexto de una situación en particular, por ejemplo un sistema para procesos empresariales, tomaría en cuenta datos sobre el giro de la empresa y las actividades que se realizan dentro de la empresa inherentes a las políticas internas de la organización; estos datos no serían de utilidad para otro tipo de situaciones como en un groupware para desarrollo de software que utiliza otro tipo de variables contextuales.

En el presente trabajo se hace una revisión de las arquitecturas conscientes del contexto para poder diseñar una dirigida a sistemas groupware independientemente de el ambiente en el que se desempeñen, la arquitectura parte de un trabajo antes propuesto en el que se divide el manejo de la información contextual en 3 fases: la recuperación de la información, gestión de los datos recuperados y el procesamiento y uso de el contexto obtenido. Inherente a estas tareas están las de implementar una ontología o clasificación de datos contextuales, técnicas de inferencia de datos contextuales y distribución de los resultados obtenidos. Una vez terminado el diseño se implementará la arquitectura con un groupware y se harán pruebas.

¹CSCW por sus siglas en inglés (*Computer Supported Collaborative Work*)

²CAGS por sus siglas en inglés (*Context Aware Groupware Systems*)

Índice general

0.1. Resumen	2
1. Introducción	5
1.1. *	5
1.2. Antecedentes	6
1.3. Definición del problema	7
1.4. Objetivos	8
1.5. Justificación	8
1.6. Alcances y Limitaciones	9
1.7. Método	9
2. Marco Teórico	11
2.0.1. Trabajo colaborativo y Sistemas Groupware	11
2.0.2. Consciencia	13
2.0.3. Consciencia del contexto	17
3. Estado del arte	21
3.0.4. CSCW y Sistemas Groupware	21
3.0.5. Groupware	22
3.0.6. Consciencia contextual	25
3.0.7. Cómputo Consciente del Contexto	26
3.0.8. Sistemas Groupware Conscientes del Contexto	32

4. Análisis	37
4.0.9. Caso de estudio	38
4.0.10. Prototipo	42
5. Diseño	47
6. Implementación	49

Capítulo 1

Introducción

1.1. *

Actualmente la mayoría de las actividades, operaciones o procedimientos que se llevan a cabo en la industria, el entretenimiento y la vida diaria son desarrollados de manera conjunta, un grupo de personas unen sus esfuerzos para poder realizar diversas tareas. Tradicionalmente los sistemas de información son un medio importante en muchas de estas tareas. El área que estudia estos sistemas es el Trabajo Colaborativo Asistido por Computadora o CSCW por sus siglas en inglés (Computer Supported Collaborative Work). Entre los sistemas que estudia CSCW están, en particular, los groupware, sistemas que apoyan a un grupo de trabajo proporcionando comunicación e información a los usuarios sobre la actividad que se están realizando, Ellis [1] define Groupware como sistemas computacionales que apoyan a grupos de personas ocupadas en una tarea en común(u objetivo) y que proporcionan una interfaz para un ambiente compartido. Para hacer que la interacción humano-computadora se lleve a cabo de manera más natural y transparente, se dota al Groupware con la habilidad de percibir y trabajar con datos que describan la situación que rodea al grupo, esto se le conoce como consciencia contextual, una característica que trae consigo el surgimiento de los Sistemas Groupware Conscientes del Contexto (CAGS).

Para que los Groupware Conscientes del Contexto(CAGS por sus siglas en inglés) puedan trabajar con datos contextuales es necesario tener una descripción del ambiente en el que va a trabajar, hace falta especificar los elementos que se deben de tomar en cuenta, por ejemplo, para un sistema de edición simultanea de textos se debe trabajar información como las modificaciones que se han hecho, la fecha de las modificaciones, los permisos de los usuarios tienen para acceder al documento, etc. mientras que para un sistema de guía de turistas se toman elementos como la ubicación del usuario, el lugar(por ejemplo un edificio, o en un espacio abierto), fechas de eventos próximos, etc. Como se puede observar en ambos casos, se usan descripciones diferentes del ambiente del sistema y esto implica tener que desarrollar 2 sistemas completamente diferentes, uno para cada caso en específico.

Esto se vuelve un problema al momento de desarrollar varios sistemas que requieren el procesamiento de información contextual, ya que hay que estar cambiando la especificación del contexto cada vez y, en el peor caso, desarrollar desde cero un sistema para un ambiente completamente distinto de los creados anteriormente.

Para evitar este problema, se propone modelar e implementar una arquitectura orientada a servicios para sistemas groupware conscientes del contexto, cuyo diseño tome en cuenta los aspectos contextuales más generales, esta arquitectura deberá poder utilizarse en cualquier ámbito reduciendo así tiempo de desarrollo, será probada en un juego colaborativo (un videojuego de disparos en primera persona) y se documentarán los resultados para futuras investigaciones.

1.2. Antecedentes

Shmidt [2], uno de los pioneros en el tema, define el trabajo colaborativo asistido por computadora (CSCW) como un área de investigación dirigida al diseño de sistemas de aplicación para una categoría específica de trabajo. Algunos de los sistemas que estudia esta área son los Groupware, sistemas de magnitud organizacional, que permiten la colaboración, comunicación y coordinación de un grupo para alcanzar una meta. Estos últimos tres conceptos (colaboración, comunicación y coordinación) son de suma importancia para el trabajo colaborativo, para que estos sean llevados a cabo de manera eficiente es necesario que los miembros del grupo tengan conciencia de la situación en muchos niveles [3].

En un groupware es difícil que los usuarios tengan conciencia completa de todo el espacio de trabajo en el que participan, por ejemplo, disminuyendo las vías de comunicación como en la conciencia de un juego colaborativo [4], cuando los usuarios juegan en una misma habitación pueden comunicarse directamente con sus compañeros, ver sus expresiones, y percibir sus sentimientos; mientras que, al jugarlo en habitaciones distintas les es más difícil poder captar este tipo de señales. Además de eso, para que el sistema pueda proveer medios de colaboración que permitan a los usuarios comunicarse eficientemente, ellos tienen que interactuar explícitamente con los medios de comunicación que ofrece el sistema, provocando distracciones que afectan la concentración de los usuarios para lograr el objetivo principal. Para dar solución a este tipo de problemas en los groupware, se le proporciona al sistema información del ámbito en el que se ejecuta para poder hacerla consciente del contexto.

El cómputo consciente del contexto es un término discutido por primera vez en el trabajo de Schilit y Theimer [5] como software que se adapta de acuerdo al contexto, esto limita la definición a aplicaciones que son informadas sobre el contexto y se adaptan a él, no dejando en claro qué tipo de adaptación es la que realiza. En investigaciones más recientes, Dey [6], tomando como referencia investigaciones anteriores, define computación consciente del contexto como un sistema que usa el contexto para proporcionar información relevante y/o servicios al usuario, donde la relevancia depende de la tarea del usuario. La definición de Dey se puede ver reflejada en el ejemplo del sistema guía de turistas, donde la información dada por el sistema es de interés para los usuarios y la actividad

que realizan, como por ejemplo, notificar de eventos próximos, o recomendar actividades o lugares a los turistas. En el caso de los videojuegos, el sistema ejecutará instrucciones para ir aumentando la dificultad o el nivel del juego conforme el usuario va incrementando su habilidad, esto cumple la segunda propiedad de la definición de Dey que es ejecutar un comando para adaptarse al contexto.

Se entiende por contexto la situación actual que tiene lugar en una actividad realizada por un sujeto o un grupo de entidades. Estas situaciones están definidas por diferentes elementos que responden a las preguntas ¿quién?, ¿dónde?, ¿cuándo? y ¿qué?. Schilit y Theimer [5] se refieren al contexto como la ubicación, identidades de personas y objetos cercanos, y los cambios en esos objetos. Mientras que la definición de Dey [6] menciona que el contexto es cualquier información relevante sobre las entidades en la interacción entre el usuario y una computadora, incluyéndolos a ambos, una entidad puede ser una persona, lugar u objeto considerado relevante para la interacción entre un usuario y una aplicación.

1.3. Definición del problema

En un ambiente colaborativo existe mucha interacción entre los integrantes, necesitan estar comunicándose constantemente para poder coordinar sus esfuerzos, y en un groupware es aún más difícil ya que el sistema se vuelve intermediario entre los usuarios, para poder hacer estas interacciones más fluidas el sistema necesita saber cómo ayudar a los usuarios y esto se logra dándole datos del contexto en general y logrando que razone y actúe en beneficio del grupo.

Para que una aplicación pueda ser consciente del contexto, debe de ser capaz de adquirir información contextual, gestionarla, y procesarla para obtener resultados que permitan ejecutar un comando o mostrar información para el usuario.

Se han propuesto muchos modelos y arquitecturas para el desarrollo de sistemas conscientes del contexto, el problema de estos sistemas es que están basados en escenarios particulares, es decir, satisfaciendo necesidades específicas del problema; por ejemplo, Meeting Reminder Agent [7], toma el tiempo de distracción de actividades, la ubicación, y el sonido en un campus para avisar al usuario de los lugares donde se están llevando a cabo reuniones, y sugerir de acuerdo con los intereses del usuario, conferencias que se lleven a cabo. Otro caso es Portable Help Desk (PHD) [7] que es una aplicación que toma en cuenta la cercanía de los miembros de un grupo y su disponibilidad para poder brindar apoyo a otros miembros y carecen de generalidad para poder ser aplicados en ambientes con un contexto diferente al que han sido desarrollados. Por último CO2DE [2] que se concentra en la edición colaborativa asíncrona de diagramas y resolución de conflictos que mantiene el contexto individual de los integrantes del proyecto y evita traslapar ediciones manteniendo a cada contexto de los individuos en una rama diferente al proyecto original.

Se puede observar en los tres ejemplos anteriores que cada aplicación usa elementos diferentes del contexto según las necesidades que cubren, y si se quiere integrar el modelo de uno de ellos a

otro ámbito diferente, haría falta integrar las características que el anterior no tenía, lo que supone una mayor carga en el modelado de soluciones. Por lo tanto, servicios fundamentales de contexto genérico son requeridos para hacer de la conciencia del contexto una tecnología factible que puede ser fácilmente incorporada a una variedad de software (Pascoe, J., Ryan, N., & Morse, D., 1999).

Montané [4] propone una arquitectura para apoyar el trabajo colaborativo en los groupware en cuyos módulos se pueden encontrar los elementos anteriores. La primera capa es de adquisición de datos contextuales en la que cada módulo trabaja con tipos diferentes de datos contextuales; en la capa de gestión de contexto se almacena la información obtenida en una base de datos para fácil acceso, aquí se guardarán, actualizarán y recuperarán los datos históricos de la aplicación; en la última capa, que es la de uso de contexto, se encuentran 2 módulos, uno que procesa y razona la información contextual, y otro que entrega los resultados al sistema groupware.

Actualmente se encuentra desarrollada la capa de adquisición de datos, pero para poder implementar la arquitectura en su totalidad, hacen falta desarrollar un modelo de las capas de gestión de datos y de razonamiento contextual, de ahí el surgimiento del presente trabajo.

1.4. Objetivos

El fin principal de este trabajo es diseñar e implementar una arquitectura funcional orientada a servicios para sistemas Groupware consientes del contexto, particularmente en el caso de estudio de los videojuegos colaborativos, partiendo de un modelo propuesto anteriormente [4]. Para poder llevar a cabo lo anterior se proponen los siguientes objetivos

- Revisar el estado del arte correspondiente a la construcción de arquitecturas context-aware.
- Implementar un modelo de consciencia contextual en el ámbito de los videojuegos
- Diseñar una arquitectura orientada a servicios para el uso, adquisición y gestión de información contextual.
- Desarrollar los módulos de cada capa de la arquitectura y realizar pruebas individuales.
- Integrar un método de razonamiento contextual para ser aplicado en la capa de razonamiento.
- Validar la arquitectura con el caso de estudio.

1.5. Justificación

La comunicación, coordinación y cooperación entre grupos colaborativos de trabajo es muy importante, con su eficiencia aumenta el rendimiento de los usuarios en el trabajo que realizan, fo-

mentando la productividad. Con los groupware se mejora la calidad de operación de estos grupos, y añadiendo la conciencia del contexto a este tipo de sistema, la interacción que los usuarios tienen con los dispositivos o aplicaciones se hace de forma más natural y fluida, permitiéndoles concentrarse en la tarea que están haciendo en lugar de detalles de comunicación y evitando la carga de procesamiento de información que el sistema hará por ellos. Con la arquitectura que se va a diseñar, el desarrollo se vuelve menos complejo, evitando que los desarrolladores empiecen desde cero un proyecto que incluya groupware consciente del contexto.

1.6. Alcances y Limitaciones

Se pretende llegar a proponer una arquitectura genérica orientada a servicios e implementarla en un sistema groupware, en este caso el sistema groupware será AssaultCube que es un juego disparador en primera persona, se extraerá información contextual del sistema y se se procesará, en la arquitectura se hará una distribución física y lógica de cada uno de los módulos de la arquitectura y se establecerán formas de comunicación entre ellos. También se propondrá una clasificación de contexto que abarque los elementos necesarios para definir la situación de un ambiente.

El servidor del videojuego Assault Cube está desarrollado para la plataforma Linux, el cliente para Windows, y la arquitectura a desarrollar será programada en C#.

1.7. Método

Para este trabajo se estiman las siguientes actividades:

Fase 1:

Se revisará el estado del arte de las arquitecturas para sistemas conscientes del contexto

Fase 2:

Se hará una investigación de las propiedades y elementos de los sistemas colaborativos e indagar como se puede establecer una comunicación con ellos para poder extraer y enviar los datos contextuales y resultados del procesamiento

Fase 3:

Clasificación de contexto de manera general, para empezar a planear la forma de procesarlo, de esto saldrá un modelo conceptual de contexto en el que se basará el desarrollo de la arquitectura antes propuesta.

Fase 4:

Distribución lógica de módulos de operación que participaran en el procesamiento de la información contextual, producto de esto será un modelo de flujo de datos de los módulos propuestos. Modelo de comunicación basado en servicios y se distribución de módulos en distintos servidores para empezar a estructurar la arquitectura.

Fase 5:

Codificación de módulos para empezar a implementar la arquitectura ya propuesta, se probaran los módulos por separado y en conjunto para determinar la efectividad del producto. Aplicación de la arquitectura a un sistema groupware para realizar un caso de estudio, el sistema es un juego FPS (first person shooter) de código abierto, en particular AssaultCube. Documentación de los resultados y conclusión.

Capítulo 2

Marco Teórico

2.0.1. Trabajo colaborativo y Sistemas Groupware

La mayoría de las actividades que una persona realiza son en grupo [1], para el apoyo a este tipo de actividades existen los sistemas groupware que son aquellos que apoyan a personas ocupadas en una tarea en común, y que proveen una interfaz a un ambiente compartido, [1] cuyo objetivo es facilitar la comunicación, cooperación, y coordinación de los usuarios, según Cruz [8] la comunicación se entiende como el proceso de interacción entre personas que incluye el intercambio implícito o explícito de información; Malone & Crowston [9] definieron coordinación como el manejo de interdependencias entre actividades realizadas por múltiples actores basadas en objetos mutuos que son intercambiados entre actividades; la cooperación ocurre cuando un grupo trabaja para lograr una misma meta [9] con alto grado de tareas interdependientes, compartiendo la información disponible a través de alguna clase de espacio compartido. Existen diferentes tipos de sistemas groupware, [1] define 2 clasificaciones: por espacio-tiempo y por tipo de aplicación. Entre el rubro espacio-tiempo se encuentran aquellos que su funcionalidad permite interacciones cara a cara o interacciones distribuidas, así mismo podemos encontrar aquellos que permiten una interacción en tiempo real o una interacción asíncrona. Además de síncronos y asíncronos agrega un tercer rubro que es la mixta que combina los dos anteriores, y engloba todos en la característica de tipo de comunicación, y para la interacción de los usuarios agrega la característica de saber si el usuario está conectado o desconectado en caso de las interacciones distribuidas, estas las incluye en la categoría de disponibilidad del usuario[10].

Por otra parte cuando se toma en cuenta el tipo de aplicación para clasificar los sistemas groupware, podemos encontrar los siguientes: Sistemas de mensajes que soportan el intercambio asíncrono de mensajes entre miembros de un grupo; editores multiusuario que permite a miembros de un grupo crear y editar un mismo documento al mismo tiempo, pueden ser síncronos o asíncronos; sistemas para soporte de decisiones de grupo y cuartos electrónicos de reuniones, proveen infraestructura para la exploración de problemas no estructurados en un grupo; cómputo para conferencias: proporciona

servicios como medios de comunicación en muchas formas, hay 3 tipos: de tiempo real, teleconferencia, y de escritorio; y por último los agentes inteligentes que son programas computacionales *inteligentes* encargados de tareas específicas.

Además de esta clasificación Ellis[1] propone elementos que tienen que ser tomados en cuenta para poder comparar y describir a los sistemas groupware, entre ellos se encuentran el contexto compartido que es el ambiente en el que va a correr el sistema; un grupo de ventanas que son las que se van a mostrar en diferentes pantallas del sistema y se van a compartir; un tele apuntador que es la capacidad de más de un usuario de mover el apuntador al mismo tiempo; vista, es la porción de contexto o ambiente que se va a mostrar en diferentes ventanas del sistema; sesión que son los tipos de interacción del usuario con el sistema y que ya se mencionaron antes (mismo tiempo, mismo lugar; diferente tiempo, diferente lugar, etc.) y los roles que son los tipos de usuarios que hay y los derechos y permisos que tienen con el sistema.

Shmidt [2] menciona varias propiedades que deben de tener los sistemas groupware:

- Es social; objeto, sujeto, medios, fines, motivos y necesidades, competencias e implementaciones, están mediadas socialmente.
- Integrantes mutuamente dependientes, necesitan cooperar para terminar el trabajo. Diferente a solo compartir el mismo recurso, sujeto A confía positivamente en la calidad y líneas de tiempo del trabajo de sujeto B y vice versa
- Distribución de tareas qué va a hacer cada individuo, cuándo y dónde.
- Distribución física en tiempo y espacio
- Distribución lógica, en términos de control, en el sentido de que los sistemas son semi autónomos en su trabajo parcial.
- Deben aumentar las capacidades mecánicas y de procesamiento de información del individuo.
- Deben combinar las actividades especializadas de múltiples trabajadores dedicados a las diferentes herramientas especializadas.
- Deben facilitar las aplicaciones de múltiples problemas resolviendo estrategias y heurísticas de un problema dado.
- Deben facilitar la aplicación de múltiples perspectivas y concepciones de un problema dado para adaptarse a naturaleza múltiple del ambiente de trabajo.
- Deben apoyar la auto organización, de conjuntos cooperativos al contrario de interrumpir trabajo cooperativo computarizando procedimientos formales.

Para Shmidt [2] el trabajo colaborativo es siempre social, en el sentido en el que el objeto y el sujeto, los fines y los medios, los motivos y necesidades están mediados socialmente, cada elemento

del grupo depende, en parte, del trabajo de algún otro miembro y viceversa, y cuando esto sucede, es importante que ambos elementos del grupo tengan el conocimiento de los avances del otro para poder hacer una estimación de sus propias acciones, esto es, que cada miembro tenga conciencia de lo que pasa a su alrededor para que pueda tomar decisiones adecuadas en cuanto a la actividad que tiene asignada en el grupo.

2.0.2. Consciencia

Gutwin [3] menciona que es importante que los individuos sepan lo que están haciendo los demás ya que pueden usar ese conocimiento para anticipar las acciones de los otros, y ayudarlos con sus tareas, Dourish y Bellotti [11] fueron los primeros en introducir el término awareness diciendo que es el entendimiento de las actividades de otros, que proporciona un contexto para tu propia actividad, aún más, dicen que este contexto es usado para asegurar que las contribuciones individuales sean relevantes a las actividades del grupo como un todo, y para evaluar las acciones individuales con respecto a los objetivos del grupo y progresos.

En un estudio realizado por [12] se mencionan las características que debe de tener la consciencia o el conocimiento, entre ellas se encuentran el tener muchas facetas, es decir, existen diferentes tipos de conocimiento como se explicará más adelante; el conocimiento está fuertemente enlazado a situaciones colaborativas ya que los colaboradores necesitan información para llevar a cabo algunas tareas o para tomar decisiones; el conocimiento en una situación colaborativa puede incrementar los niveles de confianza entre actores lo que los alienta a compartir información. Además de estas características proponen 3 tipos de conocimiento, o consciencia: Consciencia social, Consciencia de las tareas y Consciencia del espacio de trabajo; estas tres categorías se pueden identificar con las preguntas de la siguiente tabla:

Cuadro 2.1: Preguntas para entender los tipos de consciencia.

Tipo de consciencia	Preguntas
Consciencia Social	¿Qué debo de esperar de otros miembros del grupo? ¿Cómo voy a interactuar con el grupo? ¿Qué rol voy a tomar en este grupo? ¿Qué roles van a tomar los demás miembros del grupo?
Consciencia de las Tareas	¿Qué sé de este tema y la estructura de la tarea? ¿Qué saben los demás? ¿Qué se necesita para completar la tarea? ¿Cómo serán evaluados los resultados? ¿Qué herramientas u objetos se necesitan para completar esta tarea? ¿Cuánto tiempo se necesita y cuánto tiempo hay disponible?

Continúa en siguiente página

Cuadro 2.1 – *Continúa de página anterior*

Tipo de consciencia	Preguntas
Consciencia del espacio de trabajo	¿Qué hacen los demás miembros del grupo para completar la tarea? ¿Dónde están? ¿Están activos en el espacio de trabajo? ¿Qué harán? ¿Qué hacen actualmente? ¿Qué han hecho? ¿Qué harán después? ¿Cómo los puedo ayudar?

Por otro lado [10] identifica seis tipos diferentes de consciencia; la consciencia colaborativa, la consciencia contextual, consciencia social, consciencia del espacio de trabajo, consciencia situacional, y consciencia del lugar.

Consciencia colaborativa

Ha sido generalmente aceptado como la percepción de la disponibilidad del grupo que tiene cada uno de los integrantes. Disponibilidad del grupo quiere decir el conocimiento de si las personas están en el mismo lugar físico, o si están conectados o desconectados y los medios de comunicación que tienen disponibles para colaborar entre sí.

Consciencia contextual

La consciencia contextual es fundamental para permitir que un grupo colaborativo tenga conocimiento de lo que está pasando en el espacio virtual del sistema.

Consciencia social

La consciencia social [10] apunta la importancia de entender las prácticas sociales, como los roles de otros y sus actividades, o cómo están otros miembros del grupo contribuyendo a una tarea.

Consciencia de espacio de trabajo

La consciencia del espacio de trabajo se divide en 2 aspectos: uno se enfoca en el lugar y el otro se enfoca en el espacio, otra cosa importante a considerar es la interacción del grupo con los lugares de

trabajo, finalmente la noción del espacio de trabajo trae consigo el nivel de interdependencia de una tarea realizada por el grupo, considerando soporte de actividades paralelas, actividades coordinadas y actividades mutuamente ajustadas.

Conciencia situacional

Está caracterizado por tres niveles cognitivos: en el primero una percepción global del ambiente construido por eventos, acciones, recursos y otros elementos, en el segundo nivel se le da un sentido a lo que está pasando actualmente y en el tercero se construyen escenarios a futuro.

Conciencia del lugar

Puede referirse al conocimiento de los elementos de una ubicación geográfica como pueden ser coordenadas, orientación, distancia, etc. O también el conocimiento de los elementos de un espacio físico, incluyendo clima, topología física del lugar, y atributos físicos.

Por la naturaleza de los sistemas groupware las áreas de más interés son la conciencia social y la conciencia del espacio de trabajo. En otra clasificación (Gutwin, C., Greenberg, S., & Roseman, M., 1996) podemos encontrar 2 tipos de conciencia al contexto: primero conciencia general de las personas en una comunidad de trabajo, y segundo conciencia de las interacciones de otros en un espacio compartido.

Gutwin (Gutwin, Carl, Greenberg, Saul 1996) propone un marco de trabajo que considera elementos que incluyen mecanismos para recolectar información útil para la conciencia de la situación por parte de las personas, y que integran el conocimiento consciente del espacio de trabajo de un grupo.

Cuadro 2.2: Elementos de conciencia de la situación propuestos por Gutwin[3]

Elementos	Cuestiones que responden
Presencia	¿Quiénes están participando en la actividad?
Ubicación	¿Dónde están trabajando?
Nivel de actividad	¿Qué tan activos son en el espacio de trabajo?
Acciones	¿Cuáles son sus actividades y tareas actuales?
Intenciones	¿Qué harán después?, ¿Dónde van a estar?
Cambios	¿Qué cambios están realizando y en dónde?
Objetos	¿Qué objetos están usando?
Extensiones	¿Qué pueden ver?, ¿Cuáles son sus alcances?
Habilidades	¿Qué pueden hacer?
Esfera de influencia	¿Dónde pueden hacer cambios?
Expectativas	¿Qué necesita que haga ahora?

Todos los elementos de la tabla 2.0.2 se pueden clasificar en dos grupos: aquellos que se encargan de saber que está pasando con otras personas, y aquellos que se encargan de saber dónde está pasando. Esta clasificación detalla el perfil individual de cada integrante y sigue las actividades de cada uno. En otro caso, en el trabajo de Montané [4] se propone un modelo de contexto social, donde hay categorías similares, pero tomando en cuenta las relaciones entre los sujetos además de las actividades y estado actual de cada uno de ellos, los elementos que lista en su trabajo están divididos en tres categorías. La interactiva describe a los usuarios de forma individual y sus interacciones con las cosas que los rodean como objetos, tareas, eventos, usuarios o ubicaciones; la cohesiva integra elementos que tienen que ver con la actividad grupal y se pueden encontrar los grupos, roles, metas,

alianzas, actividades y reglas. Y las afectivas que describe cómo se sienten los miembros del grupo al realizar las actividades entre ellas están los sentimientos y los gestos.

2.0.3. Consciencia del contexto

Todos los elementos antes mencionados hacen referencia a la consciencia de parte del usuario del contexto que lo rodea, pero ¿es posible hacer que la consciencia del ambiente pueda ser aprendida por el propio sistema groupware? Existen sistemas groupware que son capaces de reconocer aspectos contextuales que rodean a un grupo de usuarios, estos sistemas son llamados conscientes del contexto, la información contextual más común en los sistemas groupware es la ubicación física del usuario. Al hablar de sistemas groupware conscientes del contexto surgen algunos asuntos importantes que estos han de tomar en cuenta. Para empezar hace falta una categorización formal del contexto en el que van a trabajar, el problema de esto es que los sistemas son desarrollados para que funcionen en ámbitos específicos al problema o situación a la que se le quiere dar solución.

El conocimiento contextual describe una situación, la forma en la que se usan los elementos en un grupo de trabajo, incluyendo los eventos que son manejados por el grupo [13]. Varios autores tienen un concepto de contexto algunos de ellos traslapan en definición con otros, y diferentes elementos son tomados en cuenta para la descripción de contexto. Dey [6] toma varias definiciones de contexto que otros autores habían hecho antes y hace su definición que está enfocada en el contexto en la computación, dice que contexto como cualquier tipo de información que se puede usar para caracterizar la situación de entidades (se entiende por entidad una persona, lugar u objeto) que es considerada relevante para la interacción entre un usuario y una aplicación incluyendo al usuario y la aplicación mismos.

Malik [14] hace referencia a varios problemas que hay en la actualidad inherentes a consciencia contextual, entre ellos están la definición del contexto, ya que contexto es un concepto que abarca todos los posibles parámetros que identifican una situación, las aplicaciones y marcos de trabajo están limitados a definir los parámetros del contexto de su propio ámbito. Otro problema importante es que las arquitecturas no están demasiado desarrolladas aun, están desarrolladas para tareas específicas, hace falta estándares para definir una arquitectura y herramientas, por último otro problema que es de interés para este trabajo es la interpretación del contexto y las adaptaciones del comportamiento del servicio.

Brézillon [13] estudia 3 casos de sistemas groupware conscientes al contexto y describe la forma en que apoya a la consciencia del contexto con los usuarios. SisPro es un sistema que tiene por objetivo facilitar las actividades colaborativas y procesos de aprendizaje y el desarrollo de competencias de trabajo colaborativo. SISCO tiene como tarea la preparación de reuniones, da a conocer a los usuarios los temas de los que se está hablando basados en su agenda individual. CO2DE es un software que permite unir los contextos individuales en un solo diagrama proporcionando una infraestructura de edición colaborativa. Otro caso de sistema groupware muy diferente a los anteriores es Assault Cube [4], un juego FPS (First Shooter Person) de código abierto que soporta las actividades colaborativas,

en este varios jugadores se conectan a un servidor para llevar a cabo actividades específicas de la modalidad del juego que hayan escogido, el juego ofrece mecanismos de conciencia a los jugadores tales como un mapa de la ubicación del enemigo, un medidor de vitalidad, una pantalla de mensajes para comunicar al equipo, etc.

En figura 1 Se realiza una comparación tomando en cuenta los sistemas anteriores, para poder realizar las comparaciones se listan algunos aspectos propuestos por Gutwin [3], Montané [4], Dey [6] y se clasifican en rubros más grandes propuestos por Dey en otro de sus trabajos que son ambiente de usuario y ambiente físico, el ambiente computacional, aunque es introducido por Dey, no es tomado en cuenta en ninguno de los sistemas ni en otras clasificaciones, así que es omitido.

		Tabla 1			
		SisPro project	SISCO	CO2DE	AssaultCube
Ambiente de usuario	usuarios	x	x	x	x
	roles				
	influencia		x		
	relaciones				x
	cambios de estado	x		x	x
	alianzas				x
	grupos	x	x		x
	intenciones				
	habilidades		x		x
	movimiento				x
	expectativa				
	sentimientos				
	conciencia de otros	x	x		x
Ambiente físico	identidad				x
	ubicación				x
	lugar	x	x		
	tiempo	x	x		
	clima				
	cambios		x	x	
Actividad	movimiento				x
	objetos de interacción	x		x	x
	objetivos				x
	reglas				
	estado de tareas	x	x		x
	eventos	x	x		x
	expectativas				
interacción		síncrona	mixta	síncrona	síncrona
vista		privada y compartida	pública, compartida	compartida, pública	compartida
sesión		distribuida	distribuida	distribuida	mixta
contexto		actividades colaborativas varias, procesos de aprendizaje	preparación de reuniones	edición en tiempo real de diagramas	videojuego FPS colaborativo

Figura 2.1: Comparación de sistemas groupware

Como se puede observar en la figura 2.1 no todos los sistemas abarcan los mismos aspectos contextuales, cada uno tiene una arquitectura singular creada para satisfacer las necesidades del entorno para el que fueron creadas.

En este trabajo se hace una propuesta de una arquitectura que pueda integrarse en cualquier tipo de sistema. Esta arquitectura tendrá soporte para el ambiente de aplicaciones ya que al ser el

sistema groupware una entidad en el trabajo colaborativo, tiene que tener conciencia de sí misma y de otros sistemas que la rodean, incluyendo servicios, dispositivos, sensores, etc.

Capítulo 3

Estado del arte

3.0.4. CSCW y Sistemas Groupware

El trabajo colaborativo tiene un alto impacto hoy en día, ya que las personas pueden realizar más rápido una tarea compleja si tienen los elementos necesarios para poder llevar a cabo una colaboración eficiente, hay muchas formas de trabajo colaborativo, y el contexto en el que se desarrolla es muy diverso. El área de CSCW se concentra en las actividades colaborativas, y el hecho de que múltiples individuos situados en diferentes escenarios de trabajo y situaciones, con diferentes responsabilidades, perspectivas y propensiones interactúen sean mutuamente dependientes en el conducto de su trabajo tienen implicaciones importantes en el diseño de sistemas computacionales dirigidos a apoyar estos esfuerzos [2].

Shmidt et al. menciona un conjunto de características que se deben de tomar en cuenta para que los sistemas computacionales que estudia el CSCW sean aceptables para los usuarios [2]:

- Las unidades cooperativas son grandes o son parte de conjuntos más grandes.
- Las unidades cooperativas son, por lo general, formaciones de poca duración, emergen para manejar una situación particular y luego se disuelven.
- La afiliación de las unidades cooperativas no son estables y normalmente no determinable. Las unidades cooperativas a veces se intersectan.
- El patrón de interacción en trabajos cooperativos cambia dinámicamente con los requerimientos y restricciones de la situación.
- El trabajo cooperativo es distribuido físicamente en tiempo y espacio.
- El trabajo cooperativo es distribuido lógicamente, en términos de control, es decir, los agentes son semi-autónomos en su trabajo parcial.

- El trabajo cooperativo involucra perspectivas inconmensurables (e.g. profesiones, especialidades, funciones de trabajo, responsabilidades, etc.) así como estrategias incongruentes y motivos discordantes.
- No hay agentes omniscientes en el trabajo cooperativo en escenarios naturales.

3.0.5. Groupware

Como ya se mencionó anteriormente los groupware son sistemas que apoyan el trabajo colaborativo de un grupo de personas, y lo hace brindando comunicación e información relevante al usuario. Una de las taxonomías clásicas de groupware es la de Johansen [15] quien propone 2 dimensiones de los groupware: tiempo y espacio.

	Mismo tiempo	Tiempos diferentes
Mismo Lugar	Interacción cara a cara	Interacción asíncrona
Diferentes Lugares	Interacción síncrona distribuida	Interacción distribuida asíncrona

Figura 3.1: Dimensiones de los groupware por tiempo y espacio [15]

En un estudio más reciente realizado por Cruz [8] hace una recopilación de los atributos pertenecientes a un groupware, que son dinámicas de trabajo en escenarios de colaboración (comunicación, cooperación, y coordinación), dimensiones temporales y espaciales, características grupales (tipos de tareas grupales, características y tamaño), categorías técnicas de aplicaciones groupware (escalabilidad, software y hardware), y categorías complementarias (e.g. usabilidad). A partir de estos atributos propone un modelo socio-técnico de clasificación para groupwares.

En la primera categoría se encuentra el modelo 3C, que se enfoca en la comunicación, la coordinación y la cooperación. Algunas categorías de relacionadas a la coordinación en la literatura revista por Cruz [8] son: planeación, modelos de control, relaciones entre tareas y subtareas, y manejo de la información, ajuste mutuo, estandarización, protocolos de coordinación, manejo temporal, recursos, o artefactos compartidos producidos durante la sucesión de actividades. Entre las categorías de cooperación se pueden encontrar desde producción (co-autoría), almacenamiento o manipulación de artefactos, hasta concurrencia, control de acceso.

Otro aspecto que toma Cruz en su modelo (figura anterior) es tiempo y espacio de donde se observan las siguientes subcategorías: persistencia de sesión, retardo en transmisión de audio/video, reciprocidad y homogeneidad de canales, retardo de envío de mensajes, y espontaneidad de colabo-

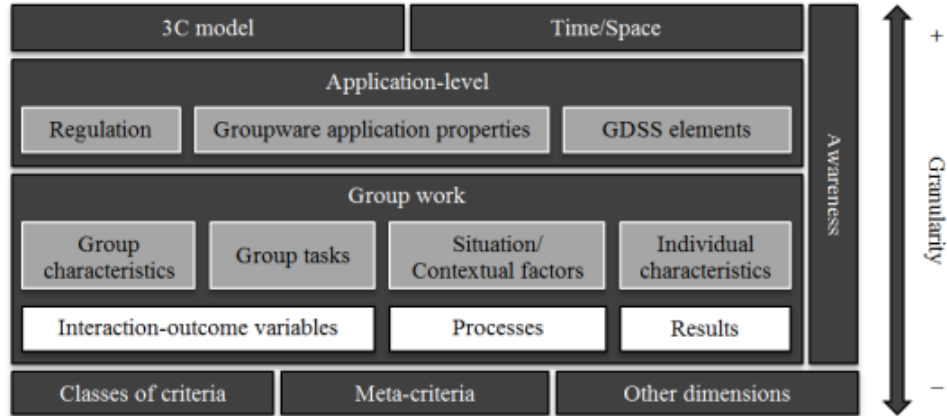


Figura 3.2: Modelo socio técnico de clasificación de groupware según Cruz [8]

ración. En la categoría de nivel de aplicación se identifican tipologías que clasifican a los groupware de acuerdo a su enfoque en el nivel del grupo. Como subcategoría se encuentran la regulación, que son los mecanismos que permiten a los participantes organizarse en un ambiente compartido; en la literatura algunas de las dimensiones de regulación que se encuentran son: arenas(ubicación); actores(roles, lugares, y posiciones); herramientas(regulativas o no); roles(temáticos o casuales); reglas(restricciones, normas o reglas de trabajo); tipos de interacción; escenarios interactivos; y objetos(medios de comunicación y productos de colaboración). Otra subcategoría del nivel de aplicación son las propiedades de la aplicación groupware entre las que se pueden encontrar arquitectura, cualidades funcionales y de calidad, soporte para procesos grupales, interfaz de colaboración(portales, dispositivos o espacio de trabajo físico), relaciones(colecciones, listas, árboles y gráficas), funciones de núcleo, contenido(texto, ligas, gráficas o flujo de datos), acciones soportadas(recibir, agregar, asociar, editar, mover, borrar, o juzgar), identificación, controles de acceso, mecanismos de alerta, componentes inteligentes, indicadores de consciencia, y plataforma, los elementos GDSS incluyen hardware, software y soporte de personas.

La siguiente capa del modelo se enfoca a los grupos, los cuales son definidos como una agregación social de individuos con consciencia de su presencia, conducida por sus propias normas, y soportado por interdependencias de tareas hacia una meta en común en un contexto compartido [16]. Dentro de esta categoría se encuentran las subcategorías: tareas grupales, características del grupo, factores de situación, y características individuales.

Un groupware se puede caracterizar por 3 aspectos complementarios importantes desde la vista del usuario: una descripción de los objetos y las operaciones que se pueden hacer sobre ellos y que están disponibles para los usuarios, una descripción de los aspectos dinámicos del sistema (flujo de control y de datos) y una descripción de la interfaz entre el sistema y los usuarios y entre los usuarios [17]. Este modelo es llamado de coordinación y describe la organización de las actividades realizadas por los usuarios y no por el sistema. Los componentes principales de este modelo ontológico son los objetos y las operaciones. Las operaciones sobre objetos son aspectos importantes que determinan el nivel de contribución de un usuario al trabajo[17], e.g. en un sistema editor de código colaborativo, los

usuarios que realicen más operaciones sobre un archivo (actualizar el archivo, crear archivo, eliminar archivo) son los que más aportan al proyecto, sin embargo la desventaja de esto es que no se sabe si la operación que se hizo produjo resultados positivos, así un usuario que realice varios cambios sobre un archivo puede ser señal de que se equivoca mucho en su redacción, otros aspectos importantes en la colaboración son la interacción con miembros del grupo y el ambiente para resolver problemas, tomar decisiones, mantener la afiliación del grupo, etc.

Antunes[18] presenta un framework detallado para evaluar los sistemas colaborativos bajo desarrollo de acuerdo a variables dadas y niveles de desempeño. Considera dos dimensiones: una define el conjunto de variables de evaluación relevantes y el otro se ocupa de los niveles de desempeño que los usuarios evaluados. El tiempo de evaluación es inherentemente asociado con el proceso de desarrollo. Menciona una lista de métodos de evaluación de groupwares son:

- Evaluación heurística de Groupware: (Backer et al 2002) es basado en ocho heurísticas groupware como una lista de características que los groupware deberían de tener. Los problemas que surjan se apuntan y se corrigen.
- Groupware walkthrough: (Pinelle and Gutwin 2002) un escenario es una descripción de una actividad o conjunto de actividades, que incluye a los usuarios, su conocimiento, el resultado esperado y las circunstancias que los rodean.
- Collaboration Usability Analysis (Pinelle et al. 2003): los evaluadores mapean acciones colaborativas a un conjunto de mecanismos de colaboración, o representaciones granulares de acciones colaborativas básicas, que pueden estar relacionadas con elementos en la interfaz de usuario, los diagramas resultantes capturan detalles sobre componentes de tareas, una noción del flujo a través de ellos y la distribución de tareas.
- Groupware Observational User Testing (Gutwin and Greenberg 2000) los evaluadores observan como los usuarios realizan actividades particulares soportadas por un sistema en un ambiente controlado. Los evaluadores monitorean a los usuarios que tienen problemas con una tarea, o piden a los usuarios pensar en voz alta sobre lo que están realizando
- Human-Performance Model (Antunes et al. 2006): los evaluadores descomponen las interfaces en espacios de trabajo compartidos, definen escenarios críticos enfocados a acciones colaborativas para los espacios compartidos y comparan el desempeño del grupo en escenarios críticos para predecir tiempos de ejecución.
- Enfoque de Manejo de conocimiento (Vizcaíno et al. 2005): se mide si el sistema ayuda a los usuarios a detectar flujos de conocimiento y diseminar, reusar y almacenar conocimiento. El proceso de circulación de movimiento se comprende de 6 fases (creación de conocimiento, acumulación, división, utilización, internalización), la evaluación se realiza contestando preguntas relacionadas a cada área.

Además de estos aspectos inherentes al trabajo colaborativo, es importante que los groupware proporcionen a los usuarios medios con los que puedan articular el trabajo que están realizando de

manera grupal, estos medios deben de apoyar la comunicación, la coordinación y la cooperación de los miembros del grupo. Para que esto se logre con mayor naturalidad es necesario mantenerlos al tanto de la situación del grupo, el estado de las actividades que se están realizando, el objetivo que persigue el grupo, etc. a esto se le llama consciencia.

La consciencia se refiere al conocimiento que tienen los individuos sobre sí mismos y sobre el ambiente que los rodea, y en el caso de trabajo colaborativo, el rol que desempeñan en su grupo y el estado de los demás integrantes.

Para que pueda haber cooperación, los miembros de un grupo deben estar consciencia de las actividades que realizan los demás, creando consciencia del grupo en el espacio de trabajo [8]. El ciclo de colaboración está limitado por la consciencia, que es la percepción del grupo acerca de lo que cada miembro desarrolla, y el conocimiento contextual que tienen sobre qué está pasando entre el grupo [19]. Cruz [8] caracteriza awareness por:

- consciencia espacial y atmosférica
- consciencia de la actividad
- consciencia de los objetos
- consciencia humana
- consciencia presencial
- consciencia influencial
- consciencia de habilidades
- consciencia contextual

3.0.6. Consciencia contextual

Consciencia contextual es el entendimiento de las actividades de los demás usuarios, que proporciona un contexto para las actividades propias [11]. El problema de mantener la consciencia del espacio de trabajo en los groupware gira en torno a la obtención de información útil más que el cómo la utilizan los usuarios [20]. La información a ser recabada se ocupa de quién está trabajando en un contexto compartido, qué están haciendo, dónde están trabajando, cuándo ocurren varios eventos, y como suceden esos eventos [20]. Similar a las interacciones cooperativas entre un grupo de personas, la consciencia contextual aumenta la experiencia que un individuo tiene en cualquier escenario de asistencia, el manejo de acoplamiento de actividades, la coordinación de acciones cooperativas, anticipación de actividades humanas o ambientales tanto como futuras intenciones, y la búsqueda de ayuda son aspectos básicos de una interacción que puede ser mejorada mediante la consciencia contextual [21].

3.0.7. Cómputo Consciente del Contexto

Contexto

Contexto es tradicionalmente la localización, identidad, y estado de las personas, grupos y objetos virtuales y físicos. según pereira [22] el contexto puede ser visto como un conjunto de condiciones e influencias en una situación relevante y que la hacen única y comprensible, esta situación puede referirse a una persona, grupo de personas, objeto físico, entidad computacional, etc. El concepto de modelos mentales tiene una relación muy cercana a la consciencia contextual y situacional [21], al momento de modelar contexto, es necesario distinguir entre los diferentes tipos de información contextual[23], el contexto de las actividades colaborativas pueden ir desde un editor de documentos distribuido, hasta un videojuego de primera persona; por lo que los elementos particulares de dichas actividades cambia muy radicalmente de uno a otro, y con esto surge la necesidad de usar una taxonomía con un alto nivel de abstracción que soporte la diversidad de contexto con los que se trabaja.

En una revisión de la literatura sobre modelos contextuales se revisaron ontologías contextuales, entre las que se podían encontrar los siguientes elementos:

Cuadro 3.1: Unidades contextuales encontradas en los frameworks revisados.

Elementos	Descripción
Objetos o artefactos	Entidades sobre las que los usuarios pueden realizar alguna acción.
Tareas	Acciones asociadas con usuarios, objetos, objetivos y sub-tareas que se deben de llevar a cabo para alcanzar un objetivo.
Eventos	Eventos ocurridos en interacciones[4]
Usuarios	Entidades que pertenecen a una comunidad y que realizan tareas[4].
Ubicaciones	Posición virtual o física en un grupo [4].
Grupos	Colección de usuarios que realizan una actividad[4].
Objetivos	Los objetivos de la comunidad[4].
Alianzas	Subconjunto de usuarios en un grupo [4].
Actividades	Actividades realizadas por un grupo[4].
Reglas	Comportamientos definidos por el grupo [4].
Foco de visión	Dónde están mirando los usuarios[24].
Vistas de sistema, espacios de trabajo	Qué pueden ver los usuarios [24].
Alcance	Alcance de los usuarios[24].

Continúa en siguiente página

Cuadro 3.1 – *Continúa de página anterior*

Elementos	Descripción
Presencia	Presencia de usuarios en el espacio de trabajo[24].
Intención	De qué objetivo es parte una tarea[24].
Habilidades	Capacidad para llevar a cabo un conjunto de actividades con cierto nivel de destreza [25].
Contexto físico	Incluye todas las magnitudes físicas (e.g. tiempo, espacio, temperatura, nivel de luz, nivel de ruido)[23].
Contexto computacional	Información relacionada con el software y hardware de sistemas, e.g. trafico de red, condiciones, estatus de hardware, información pedida por el usuario, requerimientos de memoria[23].
Ambiente	Descripción de la distribución física de los objetos y usuarios en un espacio de trabajo[23].

Todos estos elementos se extrajeron de diferentes modelos entre los que se hizo la siguiente tabla comparativa:

Autor	tipo	físico	objetos	tareas	eventos	usuarios	ubicaciones	grupos	roles	objetivos	alianzas	actividades	reglas	foco de visión	vistas	alcance	presencia	intención	habilidades	notificaciones	contexto físico	ambiente	contexto computacional	curso
Ellis, Clarence, and Jacques Wainer 1994	general		x			x		x	x		x		x											
Montane-Jimenez, L. G., Benitez-Guerrero, E., & Mezura-Godoy, C. (2013, October)	social	x	x	x	x	x	x	x	x	x	x	x	x					x						
Gallardo, J., Molina, A. I., & Bravo, C. (2012, October)	centrado en el usuario		x	x		x	x		x					x	x	x	x	x		x				
(Decauchant, D., Mendoza, S., Sánchez, G., & Rodríguez, J. 2013)	colaborativa			x		x		x	x	x					x				x	x				
(Hayes, J. R., et al., 2013) [MLContext]	general	x	x	x		x	x		x	x	x	x					x		x		x	x	x	
Guermah, H, et al. 2013	e-learning					x																x	x	x

Figura 3.3: Comparación de modelos de contexto[17][4][24]

Alves hace tres definiciones clave para el modelado de contexto y para su propagación [26], la primera es una definición de contexto: Un atributo A_i es una tupla (N, V) , donde N es un nombre representando el atributo (por ejemplo, velocidad) y V es el valor del atributo (e.g., 100); P es un conjunto finito de personas $P_1, P_2 \dots P_n$; t es el rango de tiempo entre dos marcas de tiempo; y A un conjunto finito de atributos $A_1, A_2 \dots A_n$.

Contexto C es una 3-tupla (P, t, A) que representa los atributos que caracteriza la situación de un grupo de personas P durante un intervalo de tiempo t . Por ejemplo suponiendo que Alice está en Nueva York entre Julio 1 y Julio 3. Podemos definir su contexto C como:

$$C = ((Alice), 01/07..03/07, (Ubicacion, NuevaYork))$$

La segunda definición es de agregación, sugerida para reducir la cantidad de información a procesar y reducir la carga de red, define agregación como $(P_1, t_1, A_1), \dots, (P_n, t_n, A_n)$ es un conjunto de contextos y Ac un atributo en común tal que para todo atributo A_i perteneciente a $A_1 \dots A_n$, Ac pertenece a A_i .

$$Aggr(Ac, (P_1, t_1, A_1), \dots, (P_n, t_n, A_n)) = (P, t, A) \Rightarrow \forall P_i \in P_1 \dots P_n : P_i \in P \wedge \forall t_i \in t_1 \dots t_n : t_i \subseteq t \wedge \forall A_i \in A_1 \dots A_n : A_i \in A$$

Por último así define agregabilidad, CP es el contexto actual de una persona P y C_x el contexto de alguien más que el sistema quiere propagar a P .

La función de Agregabilidad $G(CP, C_x)$ representa que tan agregado debe de estar C_x antes de ser transmitido a P , tomando en consideración el contexto actual de CP .

Por lo tanto un conjunto de contextos $C_1 \dots C_n$ es solo propagado a P cuando para todo i perteneciente a $1..n$, $G(CP, C_i) = n$, n siendo un entero. Informalmente la agregabilidad representa el número de mensajes de contexto que deben de ser retenidos antes de su propagación, si se define una función G que siempre regrese 4, el sistema siempre va a agregar cuatro mensajes contextuales antes de propagarlos"

En el trabajo de Ardissono [20] mencionan activity frames para simplificar el contexto de actividades cooperativas, un frame era definido por una 5-tupla (fn, U, O, Oi, T) , donde fn era el nombre del frame, U el conjunto de usuarios involucrados en la actividad, O los objetos asociados al frame, Oi es el conjunto de objetos del frame por medio de inferencias, y T las tareas asociadas a la actividad. Y a su vez las tareas están formadas por $(tn, U, O, Oi, g, P, T, s, d)$, donde tn es el nombre de la tarea, U , O y Oi como dicho antes son los Usuarios, Objetos, y objetos inferidos asociados a la tarea, g es el objetivo, P es el grupo de tareas que deben de estar cerradas antes de iniciar tn , T es el conjunto de tareas hijas, s es el estado(habilitada, deshabilitada, cerrada) y d es la fecha límite para terminar la tarea, nula por defecto.

En el año 2013 se propone MARS que es un modelo contextual de regulación y lenguaje asociado, ayuda a modelar la actividad soportada por herramientas groupware. En este modelo las interacciones toman lugar en un espacio llamado arena, cada interacción es representada por un escenario que describe la forma en que se lleva a cabo dicha interacción, quiénes participan y los objetos involucrados, además se definen condiciones y precondiciones para que este escenario se lleve a cabo; en esta arena están presentes actores que realizan acciones, durante esta actividad se manejan o producen objetos; actores y objetos pertenecen a diferentes familias y desempeñan diferentes papeles o roles. Dado que los miembros de un grupo colaborativo pueden serlo a su vez de otro, se definen vistas que determinan los actores, objetos e interacciones que una arena puede compartir con otra.

Junto con MARS viene un lenguaje de regulación para describir escenarios llamado CORAL¹ que toma en cuenta tres aspectos de los escenarios definidos en MARS especificados en las pre y

¹Collaborative Regulation Language

pos condiciones de la interacción: quién puede participar en la interacción, qué objetos pueden ser manipulados, qué rol tiene un actor u objeto durante la interacción y en casos similares, referencias a otros escenarios. Con esto se define el lenguaje CoRaL con la siguiente sintaxis descrita en notación BNF. Siendo E el conjunto de elementos de cadenas de nombres de arenas, actores, familias de actores, familias de objetos, objetos y roles:

$$\begin{aligned}
 \langle \text{sentencia} \rangle &::= \langle \text{expresión} \rangle \langle \text{operador} \rangle \langle \text{expresión} \rangle ";" \\
 \langle \text{expresión} \rangle &::= \langle \text{palabraReservada} \rangle : \{ \langle \text{elementos} \rangle \} \\
 \langle \text{expresión} \rangle &::= ! \langle \text{palabraReservada} \rangle : \{ \langle \text{elementos} \rangle \} \\
 \langle \text{operador} \rangle &::= :: | \rightarrow \\
 \langle \text{palabraReservada} \rangle &::= "Arena" | \\
 & "Interacción" | "Actor" | "Familiadeactor" \\
 & | "FamiliadeObjeto" | "Objeto" | "Rol" \\
 \langle \text{elemento} \rangle &::= \text{cualquiercadena en } E
 \end{aligned}$$

Un modelo enfocado en el contexto social de aplicaciones groupware clasifica los elementos contextuales en 3 categorías: la interactiva, la cohesiva y la afectiva [4]. En la interactiva se encuentran variables como: objetos, tareas, eventos, usuarios, y ubicaciones; en la cohesiva se pueden observar a grupos, roles, objetivos, alianzas, actividades, y reglas; por último están las variables afectivas en las que se encuentran las emociones y los gestos de los usuarios.

En otro estudio realizado por Skillen [27] se habla de un mecanismo de personalización basado en reglas, en el que se perfila a los usuarios por medio de un método ontológico especificando elementos como la ubicación, preferencias multimedia del usuario, objetos de asistencia que son los involucrados en la interacción, actividades, medios de entrega de información, condiciones de salud, escala de calidad multimedia y formato de interfaz gráfica.

Proceso

Para obtener resultados de un conjunto de variables contextuales se establecen 3 pasos ya descritos antes [4]: recuperar las variables contextuales en un formato legible para la computadora, almacenar, recuperar y actualizar los datos contextuales de una base de datos, e inferir resultados de un conjunto particular de variables asociadas unas con otras.

Adquisición de contexto

Antunes propone un marco de trabajo para evaluar groupwares [18], un enfoque entrada-procesamiento-salida para conceptualizar las relaciones entre el soporte tecnológico y factores relacionados al comportamiento del grupo y contexto de trabajo. Las variables contextuales son factores importantes para describir el comportamiento del grupo, están clasificadas en 5 categorías generales [18]: personales, situacionales, estructura del grupo, características de la actividad o tarea y características tecnológicas. Los procesos grupales están definidas como las características de las interacciones del grupo, incluyendo las decisionales, comunicacionales, e interpersonales. Por último este framework evalúa los resultados de los procesos grupales afectados por el soporte tecnológico, incluyendo los relacionados con las actividades y con el grupo en sí.

Manejo de contexto

Alves [26] sugiere la agregación semántica de información contextual para que sea más ligera y fácil de transportar por la red.

AYPUY es un manejador de recursos creado para ambientes de colaboración distribuidos. Almacena diferentes tipos de recursos, como contratos, historias clínicas, objetos de aprendizaje, etc., y garantiza su acceso cumpliendo con los objetos de confidencialidad, seguridad y escalabilidad de los sistemas que lo utilizan. En este framework un recurso está compuesto de un conjunto de atributos $A = L, D, F$, donde L describe las características lógicas generales del recurso (fecha de creación, autor, tipo de recurso), D establece las características relacionadas con el dominio (e.g., salud, educación) al que pertenece el recurso, y F describe las características físicas de almacenamiento del recurso (e.g., tipo de replicación, cadena de conexión), así AYPUY establece la estrategia de almacenamiento y acceso, adicionalmente los recursos tienen un conjunto de operadores O , que determinan las acciones que se pueden hacer con ellos, y son extensibles para soportar un dominio específico.

AYPUY administra los recursos controlando su acceso con espacios de trabajo (ET), crea un ET general al que pertenecen todos los usuarios del sistema siguiendo un rol específico, si se requiere la especialización o modificación de los roles de acceso a los recursos para todos o un subconjunto de usuarios, se crea otro ET. En un ambiente empresarial, por ejemplo, el ET general representa a la empresa, y un ET1 correspondería a un departamento y ET1.1 a un proyecto en específico.

Uso de contexto

La presentación de información consciente es una parte importante de los groupware, Gross[28] menciona los siguientes puntos importantes que se deben de tomar en cuenta al momento de presentar información al usuario:

- La identificación y el señalamiento de retos sobre la desorganización de contenedores de información consciente son requerimientos centrales para el apoyar la consciencia.
- Los sistemas deben de proporcionar sugerencias que los usuarios puedan sobrescribir, ya sea en un momento específico o como regla general.
- La presentación de información de consciencia debe de ser explorada con la participación del usuario, el tipo de visualización de consciencia debe de ajustarse a la necesidad de información del usuario y a su contexto.
- Modelos de consciencia son importantes para estructurar información de consciencia

Alves [26] propone un modelo genérico para propagación de contexto (Radiator) y necesidades de privacidad de aplicaciones distribuidas conscientes del contexto, enfocado a mejorar la escalabilidad de las aplicaciones y brindar privacidad de la propagación de la información, además agrega que la escalabilidad y la privacidad se pueden asegurar retrasando la propagación de contexto hasta que ciertas condiciones son alcanzadas y entonces agregar los mensajes en niveles sintácticos y semánticos.

En el trabajo de Ardissono [20] se mencionan 2 políticas dependientes del contexto para el manejo de notificaciones que apoya la selección de notificaciones para ser entregadas en base a las actividades actuales del usuario en diferentes niveles de granularidad: colaboración general de la tarea actual del usuario contra tarea llevada a cabo. Estas políticas son ofrecidas por el framework CONRAD (COntext depeNdent awaReness informAtion Delivery, por sus siglas en inglés). Las 2 políticas son las siguientes:

1. el filtro de contexto informa al usuario sobre eventos referentes a los contextos de colaboración en los que está trabajando e ignora los demás
2. el filtro de tareas es más selectivo y filtra las notificaciones basado en la tarea actual del usuario

Con este framework se reduce el nivel de distracción y la carga de trabajo presentada al usuario al momento de obtener información relacionada con su actividad. El objetivo de su trabajo es proveer a usuarios apoyo automatizado para especificar el tipo de información consciente más apropiado basado en la actividad del usuario y ajustar la entrega de las notificaciones y la aplicación de filtros para preferencias individuales de notificación.

Para adaptar servicios cooperativos al contexto del usuario, AYLLU [29] usa el framework AES que adapta la información en diversos contextos. AES funciona de la siguiente manera: una aplicación envía una consulta inicial a AES para que esta la enriquezca, AES obtiene los perfiles o características de la aplicación externa (usuario, contexto, dispositivo, perfiles), e invoca funciones de filtro de acuerdo a los perfiles. El resultado es un conjunto de datos de alta abstracción que es usado para generar una consulta enriquecida que será devuelta a la aplicación externa.

En el framework AYLLU [29] se usa un protocolo de comunicación donde los mensajes son enriquecidos con semántica y un objetivo, como en un sistema multiagentes. Se basa en la creación

de una serie de agentes, que ayuden al usuario a seguir una serie de protocolos de interacción que determinan un servicio cooperativo, cuando un servicio cooperativo se instancia un Agente Manejador de Comunidad (CMA), creado por medio de un agente de fábrica (FA). Los CMA crean agentes de comunidad (CA) los cuales ejecutan protocolos de interacción que son intercambios de mensajes estructurados entre los diferentes CA dentro de un servicio cooperativo. Si se necesita comunicación con el usuario que representa el CA se utilizan agentes asociados a la sesión del usuario. El agente administrador (AA) se encarga de administrar, como crear, actualizar, eliminar, etc. a los usuarios, roles, habilidades, recursos y grupos de trabajo. FA crea los diferentes servicios cooperativos disponibles en la plataforma, para ejecutar un servicio cooperativo se construye un CMA y un grupo de CA, el CMA controla la creación, destrucción e interacción de los CA con el fin de proveer información sobre el estado actual del servicio. Cuando termina el servicio cooperativo los CA y CMA desaparecen. Para cada usuario existen dos agentes: el manejador de sesión (SM) y el agente representante (RA), además cuentan con un agente de interfaz (IA) en cada dispositivo en el que el usuario ejecute un cliente de la plataforma AYLLU; el SM actúa como un puente entre el usuario y todos los agentes CA asociados con los servicios cooperativos de los que el usuario forma parte, el SM controla los mensajes de cada CA en los servicios cooperativos y los transmite al RA, el cual responde en nombre del usuario a diferentes peticiones, en caso de que el RA no pueda manejar las peticiones o la información, se comunica con el IA para mostrar las peticiones al usuario y le solicitará una respuesta o una acción.

3.0.8. Sistemas Groupware Conscientes del Contexto

Estos sistemas combinan las características de los groupware con los beneficios de computo consciente del contexto, ofreciendo una interacción más natural entre los usuarios y el sistema, facilitándoles los medios necesarios para poder realizar su trabajo en conjunto con los demás miembros del grupo. Para poder cumplir con la parte de consciencia contextual, en lugar de crear los mismos groupware con la característica de adaptarse al contexto, se propusieron arquitecturas de sistemas que pudieran acoplarse a los groupware existentes y que, obteniendo datos contextuales de las aplicaciones a las que se les asocia, pudieran ofrecer resultados a partir de la información que se les proporciona, resultados como pueden ser conocimiento de la situación para los usuarios, o comandos de ejecución para el sistema.

Arquitecturas

El trabajo actual se basa en un trabajo previo de Montané [4], en el que, a partir de variables contextuales observadas en experimentos realizados, se propone un modelo conceptual de una arquitectura capaz de trabajar con información contextual. La arquitectura se divide en 3 capas: la de adquisición de datos, que es la que recibe los datos por separado dependiendo de la categoría a la que pertenezcan; la de manejo de contexto, que es la capa encargada de administrar el almacenamiento, recuperación y actualización de los datos contextuales que se guardarán en una base de datos; y la capa de uso de contexto, que tiene 2 tareas principales: razonar los datos contextuales recuperados, y

a partir del resultado obtenido de este procesamiento, entregar información relevante a los usuarios de un groupware o enviar instrucciones de ejecución al sistema para poder adaptarse al contexto de los usuarios.

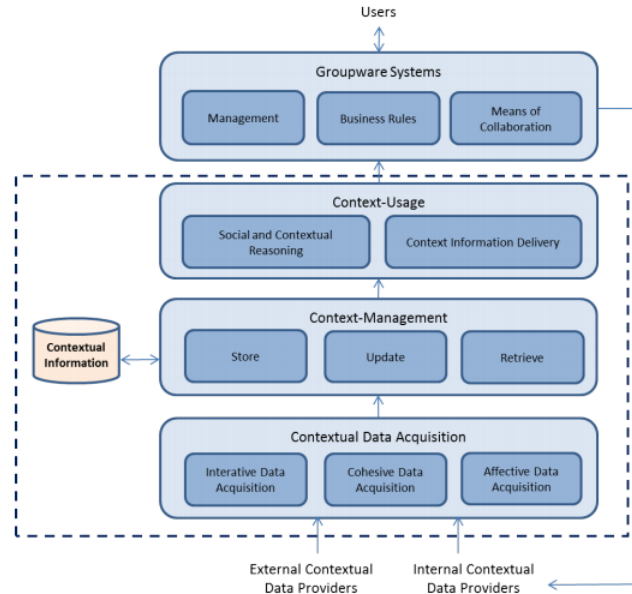


Figura 3.4: Arquitectura para soportar colaboración en groupware conscientes del contexto [4]

Para que una arquitectura sea consciente del contexto debe de cumplir con los siguientes requerimientos[30]:

- Permitir a las aplicaciones acceder a información contextual desde máquinas distribuidas en el de la misma forma en que acceden a la entrada del usuario en una máquina local.
- Soportar la ejecución de diferentes plataformas y el uso de diferentes lenguajes de programación.
- Soportar la interpretación de información contextual.
- Soportar la agregación de información contextual
- Soportar independencia y persistencia de widgets contextuales
- Soportar el almacenamiento del historial de contexto

También para el soporte de consciencia contextual, Elg Ghayam [31] menciona 2 requerimientos que se deben de cumplir: la formalización de contexto, para delimitar los datos contextuales y facilitar la distinción de parámetros de contexto, una categorización de datos contextuales es requerida, más aun, para reducir la complejidad de su manutención, una representación formal también es necesaria. El otro punto son reglas de adaptación: la adaptación de contexto debe de ser vista como un conjunto de reglas que controlan y anticipan el cambio de contexto que puede ocurrir en el ambiente, por lo

tanto, en la construcción de reglas de adaptación, el número de parámetros contextuales es grande y así, no es evidente que no se pueden enumerar todas las posibles situaciones que van a ocurrir. Por lo anterior, un método para construir reglas de adaptación es requerido, que pueda manejar la diversidad de posibles situaciones que se construyen en base de esos parámetros contextuales.

Para poder mejorar la colaboración de los usuarios, la arquitectura debe de permitir que el groupware pueda adaptarse al contexto, y de acuerdo a Abowd[32] hay 3 tipos de adaptación:

- Presentación de información y servicios al usuario. Se refiere a la técnica de interacción que muestra una lista de objetos o lugares cuyos elementos más importantes son resaltados de acuerdo al contexto actual del usuario.
- Ejecución automática de un servicio. En este caso un servicio es automáticamente lanzado si la combinación correcta de condiciones es dada.
- Aumento de la información. Información contextual puede servir para entender mejor el ambiente colaborativo.

Muchas arquitecturas se han propuesto para poder soportar sistemas conscientes del contexto, la siguiente tabla hace una comparación de los elementos y capas de algunas arquitecturas conscientes del contexto, entre las cuales se encuentran la arquitectura base para el presente trabajo que usa un modelo contextual colaborativo clasificado en tres categorías: elementos cohesivos, elementos interactivos y elementos afectivos. La arquitectura de Dey[30] usa widgets para la captura de datos contextuales y servicios de agregación de contexto así como servicios de distribución y razonamiento contextual. En el marco de trabajo de Kamoun [33] se reconfiguran servicios para adaptarlos a situaciones que cambian dinámicamente. Decouchant [25] divide su arquitectura en tres capas: la capa de espacio de trabajo, la de adaptación y la de detección de información contextual. Guerman [34] que propone una arquitectura orientada a sistemas de aprendizaje electrónico. En la figura 3.5 se comparan algunos elementos que poseen dichas arquitecturas, entre ellos se encuentran la presencia de capas como adquisición, manejo y distribución de datos contextuales, persistencia de datos y su reuso, apoyo con vías de comunicación, el uso de widgets como comunicadores entre el sistema y la arquitectura, manejo de sesiones, esquemas conceptuales de colaboración, agregación de datos y la representación de espacios de trabajo como parte de la arquitectura.

Autor	Adquisición de datos	Manejos de contexto	Representación contextual	Razonamiento contextual	Distribución de información	Persistencia de datos	Reusó de contexto	Comunicación	Widgets contextuales	Manejo de Sesión	Esquema de Colaboración	Agregación de datos	Representación de estado de trabajo
Montané E-Jiménez, L.G <i>et. al.</i> (2013) [13]	✕	✕	✕	✕	✕	✕							
Dey, A.K. <i>et. al.</i> (1999) [6]	✕		✕	✕		✕			✕				
Kamoun, E. <i>et. al.</i> (2012) [12]				✕		✕		✕		✕	✕		
Decouchant, D. <i>et. al.</i> (2013) [4]	✕	✕		✕	✕	✕			✕			✕	✕
Guermah, H., <i>et. al.</i> (2013) [10]	✕	✕	✕	✕		✕	✕						

Figura 3.5: Comparación de arquitecturas que soportan consciencia del contexto[4][30][33][25][34]

Capítulo 4

Análisis

Como se vió anteriormente las arquitecturas o marcos de trabajo que soporten consciencia contextual deben cumplir con ciertos requerimientos [30], para poder cubrir estas necesidades se necesita que el marco de trabajo sea accesible por el groupware desde cualquier ubicación, con esto se cubre la distribución de la arquitectura, así la publicación de servicios web consumibles desde el sistema colaborativo se vuelve una solución a este problema, además de que con estos servicios aumenta la compatibilidad con otro tipo de plataformas al enviar sus mensajes serializados en Json o envueltos en una solicitud SOAP. Con estos servicios se hace posible la recepción de información contextual y la emisión de los resultados hacia el groupware. Con la información recibida se mantiene un registro de la actividad del groupware que puede ser útil para análisis futuros como minería de datos o reconocimiento de patrones.

La arquitectura no debe de estar enganchada a un escenario en particular, debe de funcionar tanto para un sistema de guía de turistas como para uno de oficina, es por eso que la definición del modelo contextual se divide en dos partes, un meta modelo compuesto por entidades que describen una actividad mediante relaciones de elementos tales como *actores*, *objetos*, *tareas*, *categorías*, *roles*, *comunidades* y *objetivos* con lo que se modela las interacciones del groupware, una vez modelado el dominio del sistema se instancia a partir del meta modelo, es en esta instanciación donde se guardarán los datos del groupware.

Los datos recibidos deben de ser gestionados en bases de datos y procesados por un motor de razonamiento, el cuál tiene que recibir información contextual y mediante algún proceso tiene que dar como resultado información para el usuario o un comando de ejecución para el groupware. Una vez que se procesa la información esta se tiene que preparar para su distribución en los diferentes dispositivos cliente. Para esto se debe de considerar que al ser las actividades colaborativas los usuarios pueden compartir el contexto en algunas ocasiones, de lo que se desprende la necesidad de agregar la información contextual de usuarios que compartan ciertos atributos establecidos para una situación en particular, es decir, fusionar la información de tales usuarios para un manejo de la información más eficiente.

4.0.9. Caso de estudio

Para el actual proyecto se necesita un groupware al cuál se le pueda acoplar la arquitectura para poder analizar sus datos, en este caso el sistema elegido es un videojuego colaborativo de disparos en primera persona, *AssaultCube*. Este groupware en particular tiene las características de ser distribuido y síncrono según la clasificación de Ellis[1], contiene varios tipos de elementos y los modos multijugador son entre equipos en los cuales se requiere de una buena colaboración para cumplir los objetivos de la actividad, El juego cuenta con un servidor y varios clientes que se conectan a él para poder jugar, la arquitectura tendrá que estar acoplada al servidor ya que este es el que recibe toda la información de las actividades de los clientes. *Assault Cube* tiene diferentes modos de juego, entre ellos capturar la bandera, cuyo objetivo es llegar a la base enemiga y recuperar la bandera que está ahí para regresarla a la propia bas; otro modo de juego es rey de la colina, que tiene por objetivo mantenerse más tiempo en una posición marcada por el juego que el equipo contrario; hay más formas de juego además de estas dos, y en cada una de ellas las actividades son diferentes y poseen objetivos diferentes, sin embargo pueden compartir interacciones como eliminar a un enemigo por ejemplo.



Figura 4.1: Assault Cube

En la Tabla 1 se muestran las interacciones identificadas en el juego para la actividad relacionada al modo de juego de capturar la bandera, estas interacciones están relacionadas directamente con las actividades que se pueden llevar a cabo en el groupware, las interacciones se vuelven irrelevantes cuando no tienen conexión con ninguna actividad, por ejemplo las interacciones del usuario con elementos de configuración en este caso particular, como en la interacción "*Jugador elige modo de juego*". En estas interacciones se encuentran algunos elementos del modelo como pueden ser Actores, Tareas y Objetos, esto nos da la pauta para empezar a diseñar nuestro modelo.

Cuadro 4.1: Tabla de interacciones detectadas en *Assault Cube*

Interacción	Elementos identificados
Jugador se Mueve	Actor: Jugador; Tarea: Moverse
Jugador salta	Actor: Jugador; Tarea: Saltar
Jugador dispara arma	Actor: Jugador; Tarea: Saltar; Objeto: Arma

Continúa en siguiente página...

Cuadro 4.1 – ... *Continúa de página anterior*

Interacción	Elementos identificados
Jugador recarga arma	Actor: Jugador; Tarea: Recargar; Objeto: Arma
Jugador dispara arma	Actor: Jugador; Tarea: Saltar; Objeto: Arma
Jugador cambia arma	Actor: Jugador; Tarea: Cambiar; Objeto: Arma
Jugador obtiene mejora de salud	Actor: Jugador; Tarea: Obtener; Objeto: Mejora de salud
Jugador obtiene protección	Actor: Jugador; Tarea: Obtener; Objeto: Protección
Jugador obtiene munición	Actor: Jugador; Tarea: Obtener; Objeto: munición
Jugador envia mensaje de texto	Actor: Jugador; Tarea: enviar; Objeto: Mensaje de texto
Jugador envia mensaje de voz predefinido	Actor: Jugador; Tarea: enviar; Objeto: Mensaje de voz
Jugador elige arma inicial	Actor: Jugador; Tarea: Elegir; Objeto: Arma predeterminada
Jugador cambia rol	Actor: Jugador; Tarea: Cambiar; Objeto: Rol
Jugador se agacha	Actor: Jugador; Tarea: Agacharse
Jugador se suicida	Actor: Jugador; Tarea: Suicidarse
Jugador es eliminado	Actor: Jugador; Tarea: Ser eliminado
Jugador elimina oponente	Actores: JugadorA, JugadorB; Tarea: Eliminar
Jugador reaparece	Actor: Jugador; Tarea: Reaparecer
Jugador captura bandera	Actor: Jugador; Tarea: Capturar; Objeto: Bandera
Jugador regresa bandera a su base	Actor: Jugador; Tarea: Recuperar; Objeto: Bandera
Jugador ve mapa	Actor: Jugador; Tarea: Ver; Objeto: Mapa
Jugador ve puntuaciones	Actor: Jugador; Tarea: Ver; Puntuaciones

En la lista anterior se pueden identificar ya algunos elementos del modelo del groupware, por ejemplo, jugador como actor; arma, munición y mapa como posibles objetos, y el conjunto de ellos relacionados con una acción como tareas. También a partir de estas interacciones pueden empezar a definirse algunas reglas, por ejemplo establecer una regla que diga que si un jugador capturó una

bandera y está en peligro inminente de ser atacado, el sistema muestre a sus compañeros de equipo la ubicación del abanderado y envíe una señal de auxilio para que ellos puedan acudir a su ayuda.

En las reglas definidas para describir el contexto se muestran tres tipos diferentes:

- Pertenencia(P): Estas reglas verifican si un elemento está contenido o relacionado con otro. Esta regla está definida por $Elemento[\rightarrow | \rightsquigarrow]Elemento$; por ejemplo la pertenencia de un actor a un equipo determinado.
- Comparación(C): comparan el atributo de un elemento con un valor. La sintaxis para esta regla es la siguiente $Elemento.NombreAtributo[= | \neg | < | >]Valor$; por ejemplo que el nombre de un jugador sea "Jesús".
- Ejecución(X): indica que una tarea ha sido llevada a cabo, y si es el caso, con qué objeto se realizó, además se puede puntualizar otras características de la tarea como efectividad, o la afectación que tuvo, y si es necesario se puede añadir una regla de pertenencia en la misma declaración del elemento. $[Elemento|P] - Task(nombre, Elemento)[.nombreAtributo[= | \neg | < | >]valor]$; un ejemplo práctico de esta regla puede ser la ejecución de un disparo efectivo por parte de un jugador perteneciente al equipo rojo con un francotirador.

En donde *Elemento* es un elemento del modelo, el cual es descrito por un meta elemento, un elemento del modelo, y la instancia o instancias quedando la sintaxis de la siguiente forma: $MetaElemento(ElementoListaDeInstancias)$, evidentemente los elementos y las instancias tienen que formar parte del modelo. Así se creó la siguiente gramática libre de contexto que es la que valida el correcto uso de este lenguaje:

- $S \rightarrow ML := XJ$
- $M \rightarrow P$
- $M \rightarrow C$
- $M \rightarrow X$
- $J \rightarrow \&XJ$
- $J \rightarrow \lambda$
- $L \rightarrow \&ML$
- $L \rightarrow \lambda$
- $E \rightarrow m(eiH)$
- $E \rightarrow m(*)$

- $E \rightarrow m(e*)$
- $E \rightarrow m(iH)$
- $H \rightarrow, iH$
- $H \rightarrow \lambda$
- $P \rightarrow EOE$
- $O \rightarrow \Rightarrow$
- $O \rightarrow \rightsquigarrow$
- $C \rightarrow E.aKv$
- $K \rightarrow =$
- $K \rightarrow \neg$
- $K \rightarrow <$
- $K \rightarrow >$
- $X \rightarrow F - t(n, F)Q$
- $X \rightarrow F - t(n)Q$
- $Q \rightarrow .aKv$
- $Q \rightarrow \lambda$
- $F \rightarrow E$
- $F \rightarrow [P]$

Con estas interacciones y el conjunto de reglas definido se propone un conjunto de guiones que describen diferentes tipos de comportamiento dado el caso de estudio; entre los que se encuentra un guion para jugadores novatos, un guion para jugadores inútiles, uno para jugadores con comportamiento sospechoso que podrían considerarse como traidores y uno para jugadores expertos. Estos guiones servirán al momento de ejecutar el sistema *AssaultCube* se estará recibiendo tareas ejecutadas durante las partidas, estas tareas serán comparadas con cada uno de los guiones y se contará la frecuencia de cada una de ellas, cuando las ejecuciones lleguen a el número indicado, el guion se cumplirá y se mandará otra tarea como resultado para que se ejecute en el groupware. Una vez que se cumple un guion, este se reinicia para continuar recibiendo las tareas desde el sistema. Los guiones se definen como sigue.

Guion para jugador inútil:

$Actor(Player*) - Task(Shot)[< 3]$
 $\&Actor(Player*) - Task(Move)[> 10]$
 $\&Actor(Player*) - Task(PickItem)[> 5]$
 $\&Actor(Player*).Attribute(enemyKills) = 0$
 $\&Actor(Player*).Attribute(allyKills) = 0$
 $\&Actor(Player*) - Task(captureFlag)[= 0]$
 $\&Actor(Player*).Attribute(socialPresense) = bad$
 $\&Actor(Player*) \rightarrow Team(red)$
 $:= [Actor(Player*) \rightarrow Team(red)] - Task(sendWarningMessage"Someoneisplayingfool"), Object(UIMessageC$

Guion para jugador novato:

$Actor(Player*).Attribute(enemyKills) < 2$
 $\&Actor(Player*).Attribute(deads) > 2$
 $\&Actor(Player*) - Task(Shot)[> 5]$
 $\&Actor(Player*) - Task(scoreCapturedFlag)[< 1]$
 $\&Actor(Player*) - Task(loseFlag)[> 1]$
 $\&[Actor(Player*) \rightarrow Team(red)] - Task(Shot).Success = true[< 5]$
 $\&[Actor(Player*) \rightarrow Team(red)] - Task(Shot).Dimension = positive[< 5]$
 $:= [Actor(Player*) \rightarrow Team(red)] - Task(sendMessage"NoobPlayerneedshelp", Object(UIMessageConsole)$

Guion para jugador experto:

$Actor(Player*).Attribute(enemyKills) > 5$
 $\&[Actor(Player*) \rightarrow Team(red)] - Task(Shot).Success = true[> 5]$
 $\&[Actor(Player*) \rightarrow Team(red)] - Task(Shot).Dimension = positive[> 5]$
 $\&[Actor(Player*) \rightarrow Team(red)] - Task(Destroy).Bearer = true[> 0]$
 $\&Actor(Player*) - Task(captureFlag)[> 3]$
 $\&Actor(Player*).Attribute(socialPresence) = quitegood$
 $\&Actor(Player*).Attribute(deads) < 2$
 $:= [Actor(Player*) \rightarrow Team(red)] - Task(showPosition, Object(UImap, displayScreen)$
 $\&[Actor(Player*) \rightarrow Team(red)] - Task(sendHelpMessage"stayCloseToHero", Object(UIMessageConsole)$

Guion de jugador traidor:

$Actor(Player*) - Task(captureFlag)[= 0]$
 $\&[Actor(Player*) - > Team(red)] - Task(Shot) - Affects([Actor(Player*) - > Team(red)])[> 3]$
 $\&Actor(Player*).Attribute(socialPresence) = bad$
 $\&Actor(Player*).Attribute(allyKills) > 2$
 $\&[Actor(Player*) - > Team(red)] - Task(Shot).Dimension = negative[> 5]$
 $\&[Actor(Player*) - > Team(red)] - Task(Shot).Dimension = positive[= 0]$
 $:= [Actor(Player*) - > Team(red)] - Task(blockWeaponry) - Affects([Object(Weapon*) - >$
 $Actor(Player*)])$
 $\&[Actor(Player*) - > Team(red)] - Task(sendWarningMessage"thereisatraitorinourlines", Object(UIMessageC$

4.0.10. Prototipo

Para el presente trabajo se hace uso de un meta modelo contextual para el modelado de actividades de sistemas groupware basado un modelo propuesto anteriormente[4], en el cual se pueden encontrar dos categorías de elementos, interactivos y cohesivos. Entre los interactivos se encuentran *actores*, que son los usuarios del sistema, *objetos* que se usan en *tareas* o que son producto de ellas, *categorías* que clasifican a los actores, objetos y tareas según sus atributos, por último están los roles del objeto y del actor los cuales son asignadas a una tarea para establecer el rol que va a tener el actor u objeto involucrado en dicha tarea. Entre los datos cohesivos se encuentran las *comunidades* que son el conjunto de actores con una *actividad* en común, estas actividades pueden tener varias *metas* las cuales se cumplen realizando tareas. Por último se encuentran las reglas que son sentencias en un lenguaje definido para poder inferir las interacciones que suceden en el groupware. En la figura 4.2 se muestra un diagrama de los elementos de este modelo y sus relaciones.

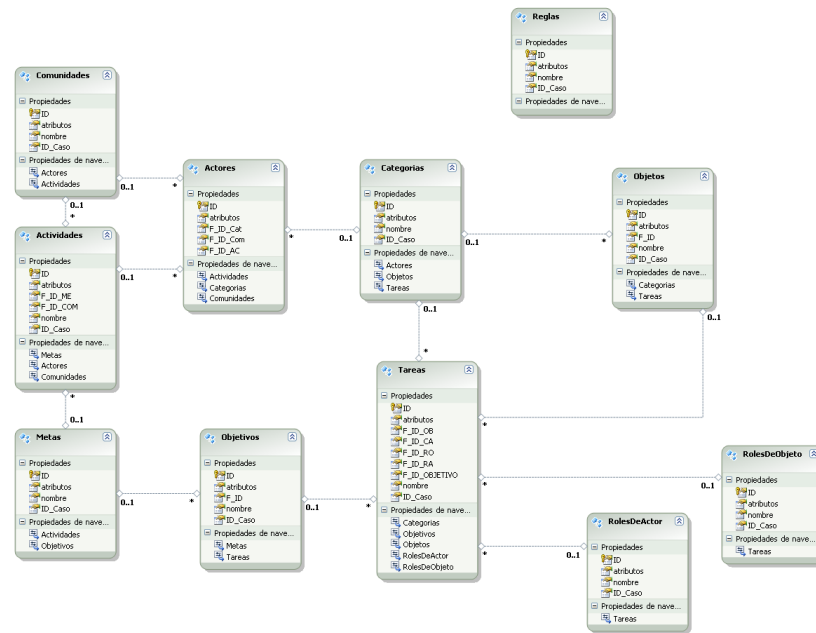


Figura 4.2: Meta modelo contextual

Con este meta modelo contextual se pueden describir groupwares definiendo cada uno de estos elementos a partir de interacciones, una vez dado de alta un caso junto con sus reglas se instancia un modelo que representará al sistema colaborativo y almacenará las variables contextuales que este transmita a la arquitectura. Cabe mencionar que en este metamodelo los elementos cuentan con 3 atributos principales, un identificador del objeto, un nombre descriptivo, y una lista de atributos almacenados en formato JSON, lo que vuelve flexible la forma de registrar casos de estudio.

La arquitectura que se propone mostrada en la figura 4.3, cuenta con tres capas: recuperación de datos, gestión de contexto, y uso de contexto. Para poder acoplar la arquitectura primero se

tiene que dar de alta un modelo del groupware. Para esto se creó una plataforma para registrar casos de estudio en la que se establece el nombre del caso de estudio y todos sus elementos, una vez creado el meta modelo del groupware se instancia el modelo y se administran las interacciones para poder relacionar sus elementos, esto se hace en una segunda plataforma en la que se toman los elementos del meta modelo y se instancia uno más apegado al groupware, ya con este modelo se pueden capturar los datos contextuales que el sistema va a enviar a la arquitectura.

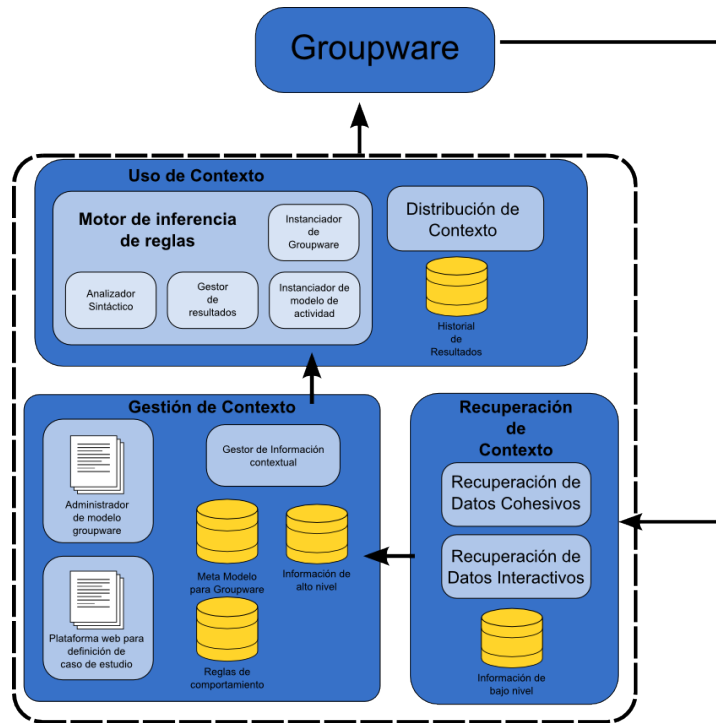


Figura 4.3: Arquitectura propuesta

La arquitectura, similar a la arquitectura en que se basó, se divide en tres capas; la capa de recuperación de datos, la capa de gestión de datos y la capa de uso de contexto. En la primera capa, la de recuperación de datos, se captura información contextual enviada por el groupware por medio de servicios web publicados para la comunicación entre la arquitectura y el sistema, estos datos son enviados en un formato específico para que capas superiores puedan procesarla. Seguido de este módulo está el de gestión contextual, este gestor se encarga de registrar, actualizar y recuperar la información contextual en bases de datos, es a este nivel donde el meta modelo y el modelo del sistema se elaboran y donde los datos enviados desde el groupware se registran y son recuperados para inferir resultados en niveles superiores.

En esta capa se encuentra una plataforma en la que se pueden dar de alta elementos del modelo de actividad.

En la última sección de la arquitectura, uso de contexto, los datos contextuales recuperados son procesados por un motor de inferencia que da como resultado información de interés para los

usuarios o comandos de ejecución para la adaptación del sistema a la situación actual. El núcleo de este motor es un autómata que especifica un lenguaje para definir reglas, mismas que van a ser definidas en las plataformas antes mencionadas, dentro del motor se hará una instancia del modelo del groupware con los datos que envíe el sistema, así se puede comparar su estado actual con las reglas de inferencia con el objetivo de ofrecer resultados coherentes al tiempo en el que el sistema se ejecuta. Los resultados obtenidos son gestionados por un administrador de resultados que almacena los datos en una base de datos para mantener registro histórico del comportamiento del groupware. Una vez obtenidos y almacenados los resultados un módulo de distribución de datos se encarga de enviar la información al groupware con la información de entrega necesaria. Este proceso es iterativo, ya que vive por el tiempo en el que el groupware opera.

Capítulo 5

Diseño

xdfgcvhjbklfghjbknl

Capítulo 6

Implementación

Bibliografía

- [1] C. A. Ellis, S. J. Gibbs, and G. Rein, “Groupware: Some issues and experiences,” *Commun. ACM*, vol. 34, no. 1, pp. 39–58, Jan. 1991. [Online]. Available: <http://doi.acm.org/10.1145/99977.99987>
- [2] K. Schmidt and L. Bannon, “Taking cscw seriously,” *Computer Supported Cooperative Work (CSCW)*, vol. 1, no. 1-2, pp. 7–40, 1992. [Online]. Available: <http://dx.doi.org/10.1007/BF00752449>
- [3] C. Gutwin, S. Greenberg, and M. Roseman, “Supporting awareness of others in groupware,” in *Conference Companion on Human Factors in Computing Systems*, ser. CHI ’96. New York, NY, USA: ACM, 1996, pp. 205–. [Online]. Available: <http://doi.acm.org/10.1145/257089.257282>
- [4] L. Montane-Jimenez, E. Benitez-Guerrero, and C. Mezura-Godoy, “A context-aware architecture for improving collaboration of users in groupware systems,” in *Collaborative Computing: Networking, Applications and Worksharing (Collaboratecom), 2013 9th International Conference Conference on*, Oct 2013, pp. 70–76.
- [5] B. Schilit and M. Theimer, “Disseminating active map information to mobile hosts,” *Network, IEEE*, vol. 8, no. 5, pp. 22–32, Sept 1994.
- [6] A. K. Dey, G. D. Abowd, and D. Salber, “A conceptual framework and a toolkit for supporting the rapid prototyping of context-aware applications,” *Hum.-Comput. Interact.*, vol. 16, no. 2, pp. 97–166, Dec. 2001. [Online]. Available: http://dx.doi.org/10.1207/S15327051HCI16234_02
- [7] J. Anhalt, A. Smailagic, D. Siewiorek, F. Gemperle, D. Salber, S. Weber, J. Beck, and J. Jennings, “Toward context-aware computing: experiences and lessons,” *Intelligent Systems, IEEE*, vol. 16, no. 3, pp. 38–46, May 2001.
- [8] A. Cruz, A. Correia, H. Paredes, B. Fonseca, L. Morgado, and P. Martins, “Towards an overarching classification model of cscw and groupware: A socio-technical perspective,” in *Collaboration and Technology*, ser. Lecture Notes in Computer Science, V. Herskovic, H. Hoppe, M. Jansen, and J. Ziegler, Eds. Springer Berlin Heidelberg, 2012, vol. 7493, pp. 41–56. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-33284-5_4
- [9] T. W. Malone and K. Crowston, “The interdisciplinary study of coordination,” *ACM Comput. Surv.*, vol. 26, no. 1, pp. 87–119, Mar. 1994. [Online]. Available: <http://doi.acm.org/10.1145/174666.174668>

- [10] P. Antunes, V. Herskovic, S. F. Ochoa, and J. A. Pino, "Reviewing the quality of awareness support in collaborative applications," *Journal of Systems and Software*, vol. 89, no. 0, pp. 146 – 169, 2014. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0164121213002756>
- [11] P. Dourish and V. Bellotti, "Awareness and coordination in shared workspaces," in *Proceedings of the 1992 ACM Conference on Computer-supported Cooperative Work*, ser. CSCW '92. New York, NY, USA: ACM, 1992, pp. 107–114. [Online]. Available: <http://doi.acm.org/10.1145/143457.143468>
- [12] F. Belkadi, E. Bonjour, M. Camargo, N. Troussier, and B. Eynard, "A situation model to support awareness in collaborative design," *International Journal of Human-Computer Studies*, vol. 71, no. 1, pp. 110 – 129, 2013, special Issue on supporting shared representations in collaborative activities. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S107158191200050X>
- [13] P. Brézillon, M. Borges, J. Pino, and J.-C. Pomerol, "Context-awareness in group work: Three case studies," in *Proc. of*, 2004.
- [14] N. Malik, U. Mahmud, and Y. Javed, "Future challenges in context-aware computing," in *proceedings of the IADIS International Conference WWW/Internet*, 2007, pp. 306–310.
- [15] R. Johansen, *GroupWare: Computer Support for Business Teams*. New York, NY, USA: The Free Press, 1988.
- [16] D. Pumareja and K. Sikkil, "An evolutionary approach to groupware implementation: the context of requirements engineering in the socio-technical frame," no. TR-CTIT-02-30, pp. 1–27, 2002. [Online]. Available: <http://doc.utwente.nl/38248/>
- [17] C. Ellis and J. Wainer, "A conceptual model of groupware," in *Proceedings of the 1994 ACM Conference on Computer Supported Cooperative Work*, ser. CSCW '94. New York, NY, USA: ACM, 1994, pp. 79–88. [Online]. Available: <http://doi.acm.org/10.1145/192844.192878>
- [18] P. Antunes, V. Herskovic, S. F. Ochoa, and J. A. Pino, "Structuring dimensions for collaborative systems evaluation," *ACM Comput. Surv.*, vol. 44, no. 2, pp. 8:1–8:28, Mar. 2008. [Online]. Available: <http://doi.acm.org/10.1145/2089125.2089128>
- [19] D. Mittleman, R. Briggs, J. Murphy, and A. Davis, "Toward a taxonomy of groupware technologies," in *Groupware: Design, Implementation, and Use*, ser. Lecture Notes in Computer Science, R. Briggs, P. Antunes, G.-J. de Vreede, and A. Read, Eds. Springer Berlin Heidelberg, 2008, vol. 5411, pp. 305–317. [Online]. Available: http://dx.doi.org/10.1007/978-3-540-92831-7_25
- [20] L. Ardissono and G. Bosio, "Context-dependent awareness support in open collaboration environments," *User Modeling and User-Adapted Interaction*, vol. 22, no. 3, pp. 223–254, 2012. [Online]. Available: <http://dx.doi.org/10.1007/s11257-011-9100-1>

- [21] M. Aehnelt, C. Peter, and P. Müsebeck, "A discussion of using mental models in assistive environments," in *Proceedings of the 5th International Conference on Pervasive Technologies Related to Assistive Environments*, ser. PETRA '12. New York, NY, USA: ACM, 2012, pp. 37:1–37:5. [Online]. Available: <http://doi.acm.org/10.1145/2413097.2413145>
- [22] J. Pereira de Souza, C. Tacla, F. Beal, E. Cabrera Paraiso, and G. Gimenez-Lugo, "An ontology-based agent for context aware software process development," in *Computer Supported Cooperative Work in Design (CSCWD), 2013 IEEE 17th International Conference on*, June 2013, pp. 287–292.
- [23] J. R. Hoyos, J. G. Molina, and J. A. Botía, "A domain-specific language for context modeling in context-aware systems," *Journal of Systems and Software*, vol. 86, no. 11, pp. 2890 – 2905, 2013. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0164121213001696>
- [24] J. Gallardo, A. I. Molina, and C. Bravo, "A framework for the design of awareness support in collaborative situations of implicit interaction," in *Proceedings of the 13th International Conference on Interacción Persona Ordenador*, ser. INTERACCION '12. New York, NY, USA: ACM, 2012, pp. 7:1–7:2. [Online]. Available: <http://doi.acm.org/10.1145/2379636.2379643>
- [25] D. Decouchant, S. Mendoza, G. SÁnchez, and J. RodrÍguez, "Adapting groupware systems to changes in the collaborator's context of use," *Expert Systems with Applications*, vol. 40, no. 11, pp. 4446 – 4462, 2013. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0957417413000742>
- [26] P. Alves and P. Ferreira, "Radiator: Context propagation based on delayed aggregation," in *Proceedings of the 2013 Conference on Computer Supported Cooperative Work*, ser. CSCW '13. New York, NY, USA: ACM, 2013, pp. 249–260. [Online]. Available: <http://doi.acm.org/10.1145/2441776.2441806>
- [27] K.-L. Skillen, L. Chen, C. D. Nugent, M. P. Donnelly, W. Burns, and I. Solheim, "Ontological user modelling and semantic rule-based reasoning for personalisation of help-on-demand services in pervasive environments," *Future Generation Computer Systems*, vol. 34, no. 0, pp. 97 – 109, 2014, special Section: Distributed Solutions for Ubiquitous Computing and Ambient Intelligence. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0167739X1300246X>
- [28] T. Gross, "Supporting effortless coordination: 25 years of awareness research," *Computer Supported Cooperative Work (CSCW)*, vol. 22, no. 4-6, pp. 425–474, 2013.
- [29] M. Arias-Baez, L. Torres-Ribero, A. Carrillo-Ramos, A. Quimbaya, and E. Gonzalez, "Platform based on agents for support to the collaboration of work teams," in *Computing Congress (CCC), 2012 7th Colombian*, Oct 2012, pp. 1–6.
- [30] A. K. Dey, D. Salber, M. Futakawa, and G. D. Abowd, "An architecture to support context-aware applications," 1999.
- [31] Y. El Ghayam and M. Erradi, "Distributed context management in collaborative environment," in *New Technologies of Distributed Systems (NOTERE), 2011 11th Annual International Conference on*, May 2011, pp. 1–8.

- [32] G. D. Abowd, A. K. Dey, P. J. Brown, N. Davies, M. Smith, and P. Steggles, “Towards a better understanding of context and context-awareness,” in *Handheld and Ubiquitous Computing*, ser. Lecture Notes in Computer Science, H.-W. Gellersen, Ed. Springer Berlin Heidelberg, 1999, vol. 1707, pp. 304–307.
- [33] A. Kamoun, S. Tazi, and K. Drira, “Fadycos, a semantic interoperability framework for collaborative model-based dynamic reconfiguration of networked services,” *Computers in Industry*, vol. 63, no. 8, pp. 756 – 765, 2012, special Issue on Sustainable Interoperability: The Future of Internet Based Industrial Enterprises. [Online]. Available: <http://www.sciencedirect.com/science/article/pii/S0166361512001261>
- [34] H. Guermah, T. Fissaa, H. Hafiddi, M. Nassar, and A. Kriouile, “Ontology based context aware e-learning system,” in *ISKO-Maghreb, 2013 3rd International Symposium*, Nov 2013, pp. 1–7.