

**IMPLEMENTACIÓN DE UN MODELO DE SISTEMA DE RECOMENDACIÓN
QUE INTERPRETA DATOS GENERADOS POR HERRAMIENTAS ESTILO
PERFMON.**

**DAVID EUGENIO QUINTANA ZAPATA. 1101334
RODOLFO REY SANTAMARIA. 1101333**

**FACULTAD DE INGENIERIA DE SISTEMAS
ESPECIALIZACION EN PROCESOS PARA EL DESARROLLO DE SOFTWARE
CALI
18 MAYO 2011**

TABLA DE CONTENIDO

INTRODUCCION.....	1
1. OBJETIVOS DE LA INVESTIGACION.....	2
1.1 OBJETIVO GENERAL.....	2
1.2 OBJETIVOS ESPECIFICOS.....	2
2. ALCANCES Y LIMITACIONES	3
2.1 ALCANCES	3
2.2 LIMITACIONES	4
3. REQUISITOS PARA LA IMPLEMENTACION DEL MODELO.....	5
3.1 MINIMIZAR CONSUMO DE RECURSOS DE RED	5
3.2 MINIMIZAR CONSUMO DE RECURSOS MONETARIOS.....	5
3.3 COHESIVIDAD.....	5
3.4 FACIL ACCESO	5
4. SISTEMAS DE RECOMENDACIÓN	6
4.1 ORIGEN, BASE Y DEFINICION	6
4.2 BASE DE CONOCIMIENTO	7
4.2.1 Feedbacks Explícitos.....	7
4.2.2 Feedbacks Implícitos	8
4.3 PROCESO DE CONOCIMIENTO.....	8
5. TIPOS DE SISTEMAS DE RECOMENDACIÓN	10
5.1 RECOMENDACIÓN BASADA EN EL CONTENIDO.....	11
5.2 RECOMENDACIÓN BASADA EN CASOS.....	11
5.3 RECOMENDACIÓN COOPERATIVA.....	12
5.4 RECOMENDACIÓN HÍBRIDA.....	12
5.5 VENTAJAS Y DESVENTAJAS ENTRE TIPOS DE SISTEMAS DE RECOMENDACION.....	13

6.	HERRAMIENTAS DE MONITOREO DE RECURSO	15
6.1	COMANDO PS EN LINUX.....	15
6.2	COMANDO TYPEPERF DE WINDOWS	17
7.	COMPLEMENTOS DE MONITOREO DE LA HERRAMIENTA DE WINDOWS	
PERFMON	19	
8.	IMPLEMENTACION DEL MODELO DE SISTEMA DE RECOMENDACION	21
8.1	ARQUITECTURA DEL MODELO	22
8.2	DESCRIPCION DEL MODELO DE LA SOLUCION	23
8.2.1	Archivo ejecutable	23
8.2.2	Archivo de salida	23
8.2.3	Tarea Programada	24
8.2.4	Transferencia de archivos	24
8.2.5	Suministro de información	25
8.3	TECNOLOGIAS UTILIZADAS EN EL MODELO.....	25
8.3.1	UNIDAD DE RED	26
8.3.2	SAMBA.....	26
8.3.3	BASE DE DATOS DB2.....	27
8.3.4	LENGUAJE DE PROGRAMACIÓN .NET	28
8.4	INTERPRETACION DE LOS DATOS GENERADOS	29
8.5	RECOMENDACIÓN A BASE DE CASOS	30
8.6	APLICACIONES.....	30
9.	CONCLUSIONES	32
10.	BIBLIOGRAFIA	33

LISTA DE ILUSTRACIONES

Ilustración 1 - Casos de Implementación de los sistemas de Recomendación.....	6
Ilustración 2 – Arquitectura de una solución del modelo.	22
Ilustración 3 - Esquema simple del funcionamiento de Bases de Datos en DB2.	28

LISTA DE TABLAS

Tabla 1 - Columnas importantes del proceso PS en Linux.....	17
Tabla 2 – Opciones disponibles para la ejecución del comando typeperf.	18
Tabla 3 - Ejecución del typeperf con los complementos especificados.	24
Tabla 4 - Promedio por componente de datos generados por el typeperf.	29
Tabla 5 - Promedio por componente de datos generados por el typeperf.	29

INTRODUCCION

En la actualidad los administradores de sistemas están siendo sobrepasados por la cantidad de maquinas que deben administrar y mantener en funcionamiento. Esto es debido al aumento, no necesariamente de servidores, sino de la capacidad de computo de estos junto con la combinación de la virtualización que ya casi todos estos están presentando; así pues, en una maquina física tenemos muchos servidores virtuales, los cuales hacen que se aproveche la capacidad de computo al 'máximo'.

Teniendo en cuenta el cambio de tecnología y el incremento exponencial de consumo de recursos de una maquina, es necesario presentarle al administrador de sistemas, soluciones que le faciliten la administración de sus máquinas a través de soluciones terceras que brinden un aprovechamiento al máximo del rendimiento, recursos y monitoreo de la arquitectura que manejan.

Por esta razón, los administradores de sistemas deben basarse en herramientas que les facilite el trabajo de monitoreo y evaluación de las maquinas que tienen a cargo. La dificultad radica en que la mayoría de herramientas de monitoreo disponible tienden a ser costosas o de difícil manejo por la interfaz que poseen.

La idea principal de esta investigación es presentar un estudio sobre los sistemas de Recomendación para el mantenimiento de la calidad del servicio en los servidores, describiendo los aspectos más significativos de su diseño y problemas fundamentales con los que nos encontramos al momento de diseñar un sistema de este tipo. Además, presentarle a las organizaciones, arquitecturas y administradores de sistemas, un modelo de sistemas de recomendación basado en la herramienta Perfmon de Windows, que le permita tomar decisiones que afecten de manera positiva el rendimiento y la utilización de los recursos.

1. OBJETIVOS DE LA INVESTIGACION

1.1 OBJETIVO GENERAL

Proponer un modelo de implementación de un sistema de recomendación que pueda interpretar datos generados por herramientas estilo Perfmon.

1.2 OBJETIVOS ESPECIFICOS

- Elaborar una técnica de implementación de sistemas de recomendación basados en algoritmos de filtrado colaborativo.
- Modelar una solución informática para la captura de la información suministrada por los distintos componentes Perfmon que alimenten el sistema de recomendación.
- Evaluar y especificar en qué casos la implementación de un sistema de recomendación basado en componentes Perfmon es requerida o no para la administración de sistemas.

2. ALCANCES Y LIMITACIONES

Si bien el tema de los sistemas de recomendación y las herramientas Perfmon, no es un tema nuevo, se toma en consideración que la solución de implementar un modelo de administración de sistemas que tenga como base estas dos tecnologías, es requerida siempre y cuando el sistema lo amerite.

2.1 ALCANCES

Entre los alcances del proyecto encontramos que la implementación del modelo propuesto, es una extensión del modelo de virtualización de maquinas. Dicha extensión les permitirá a los administradores de sistemas, escalar el nivel de administración de las mismas.

El modelo es una solución informática de alto nivel, ya que podría ser implementado utilizando tecnología en la nube, permitiéndole a una organización la administración de múltiples sistemas y de forma masiva.

Por otro lado, el modelo propuesto podría ser estructurado y compuesto como una herramienta de usuario final en donde el requisito de tener conocimientos de virtualización de maquinas y servidores, no sería de suma importancia para un administrador de sistemas.

El modelo también ofrece a los usuarios un amplio espectro de posibilidades que le permiten tomar una decisión que afecte de manera positiva el rendimiento y los recursos de las maquinas que administra. Esto es posible gracias a que, además de la información suministrada por la herramienta de monitoreo, los demás usuarios participantes suministran resultados de sus decisiones, manteniendo así un flujo constante de información.

2.2 LIMITACIONES

El modelo propuesto es modificable. Las tecnologías utilizadas para implementarlo pueden variar dependiendo de la necesidad de la organización o usuario que lo requiera. Los medios de almacenamiento, las tecnologías de comunicación para la transferencia de archivos e incluso el uso de la herramienta Perfmon como herramienta de monitoreo, pueden cambiar.

Debido a la flexibilidad que presenta la implementación del modelo, del mismo modo el tiempo de implementación y ejecución del mismo puede variar. Dependiendo de la complejidad de la tecnología escogida, pueden presentarse dificultades debido a la diversidad o heterogeneidad de conocimiento de los participantes en la implementación.

Dada la magnitud del modelo y de la solución, la información o los datos necesarios para alimentar el sistema de recomendación es demasiado extensa. Por lo tanto, para alcanzar el objeto de la investigación de este trabajo, se implementaron teorías y ejemplos que simularán la cantidad de información disponible.

La solución propuesta requiere que las organizaciones, sistemas o administradores de sistemas, cuenten con información precisa, consistente y de alto volumen, de lo contrario, el modelo presentado podría arrojar valores inciertos o un detalle no deseado por el usuario.

Por último, dado el alcance del mismo, el tiempo de implementación del modelo puede ser un limitante cuando las directivas de la organización involucrada, desean obtener resultados de forma rápida y en el menor tiempo posible. Esto se debe a que los sistemas de recomendación requieren de un alto volumen de información, como se menciona previamente.

3. REQUISITOS PARA LA IMPLEMENTACION DEL MODELO

Para la implementación del modelo es necesario cumplir con una serie de requisitos que permitan que el proceso sea óptimo, eficiente y eficaz.

3.1 MINIMIZAR CONSUMO DE RECURSOS DE RED

El modelo utiliza tecnologías que consumen la red del canal de información, por lo tanto, se requiere evitar utilizar tecnologías que excedan el consumo del ancho de banda.

3.2 MINIMIZAR CONSUMO DE RECURSOS MONETARIOS

La adquisición de productos licenciados debe ser mínima para de esta forma garantizar que la implementación del modelo no sea costosa para las organizaciones interesadas en implementarlo.

3.3 COHESIVIDAD

El modelo permite utilizar distintos sistemas operativos, por lo tanto la implementación del mismo no debería ser limitante al momento de escoger el sistema operativo de la maquina a medir.







3.4 FACIL ACCESO

Los usuarios finales interesados en consultar la información del sistema de recomendación, deben tener la posibilidad de accederla a través de cualquier medio o ubicación.

4. SISTEMAS DE RECOMENDACIÓN

El tema de los sistemas de recomendación, es el eje principal del objeto de investigación. El modelo propuesto más adelante, fue concebido con el fin de optimizar las funciones de un sistema de recomendación. La practica en estos sistemas, sugieren que la implementación de esta solución sea dependiendo de la cantidad de información y la fuente de donde sale esta información.

Información

Mucha			
Media			
Poca			
	Poca	Media	Mucha

Fuente de Información

Ilustración 1 - Casos de Implementación de los sistemas de Recomendación.

4.1 ORIGEN, BASE Y DEFINICION

Un sistema de recomendación se define como la solución “[...] capaz de realizar predicciones del hecho que un usuario le gustara o no cierto ítem. Usando el término ‘ítem’ nos referimos a cada objeto, medio o información que un usuario podría acceder. [...] No hay restricciones en que es un ítem, y por supuesto, no hay limitaciones sobre su recomendación”¹.

¹ Traducido de: CHESANI, F., Recommendation Systems. 123 p. 2007.

Aunque el origen de los sistemas de recomendación puede remontarse principalmente al extenso trabajo en ciencia cognitiva, teoría de la aproximación y recuperación de la información, los sistemas de recomendación surgieron como un área de investigación independiente a mediados de la década de los noventa. Por esos años fue cuando los investigadores empezaron a centrarse en problemas de recomendación que dependían explícitamente de las valoraciones de usuario.

Las aplicaciones más conocidas y utilizadas dentro del ámbito de los sistemas de recomendación, son en el comercio electrónico en donde la información suministrada por el usuario va desde el historial de productos adquiridos hasta los gustos especificados de forma explícita.

La forma más común de formular el problema de la recomendación es desde una perspectiva probabilística, en la que la tarea de recomendar es reducida al problema de calcular el valor esperado de un voto (valoración) de usuario sobre algún contenido, dado lo que el sistema conoce del usuario.

El objetivo de esta investigación se centrará en aplicar los sistemas de recomendación en estructuras de virtualización de máquinas utilizando herramientas de monitoreo.

4.2 BASE DE CONOCIMIENTO

Los sistemas de recomendación dependen de la información con la cual son alimentados o la información disponible al momento de realizar los cálculos necesarios. Dicha información es denominada *Feedback* y puede ser de forma Explícita o Implícita.

4.2.1 *Feedbacks Explícitos*

Información suministrada directamente por el usuario. En el caso de las redes sociales y los comercios electrónicos, la información está compuesta por los

comentarios, votaciones, listados y cualquier elemento que denote su preferencia por un "ítem".

4.2.2 Feedbacks Implícitos

Información que se obtiene del comportamiento del usuario dentro de la red social y el comercio electrónico pero depende más del historial de dicho comportamiento, como por ejemplo, el listado de productos adquiridos, visitados y consultados. Esta información no es directamente especificada por el usuario sino dependiendo de sus acciones.

4.3 PROCESO DE CONOCIMIENTO

“El conocimiento es un conjunto de información almacenada mediante la experiencia o el aprendizaje, o a través de la introspección. En el sentido más amplio del término, se trata de la posesión de múltiples datos interrelacionados que, al ser tomados por sí solos, poseen un menor valor cualitativo.”².

El proceso para adquirir este conocimiento es particular a cada entidad y puede ser simulado en equipos de cómputo a través de muchos métodos, algunos desarrollados mediante el uso de inteligencia artificial. En otros casos, el conocimiento debe provenir dependiendo de las necesidades de un usuario que lo requiere. Es ahí donde entran a participar los sistemas de recomendación.

Los sistemas de recomendación utilizan una serie de diversos métodos, desde los tipos de *feedbacks*, como anteriormente mencionamos, hasta información basada en la descripción de posibles casos.

“En estos sistemas los usuarios dan un ejemplo del tipo de producto que están buscando, y el sistema buscará y recomendará productos similares al ejemplo dado. Como es de suponer, la mayoría de las veces el usuario lo que busca no es

² DEFINICION.DE. Definición de Conocimiento. [En línea] <<http://definicion.de/conocimiento/>> [citado el 01 de Julio de 2011].

un producto exactamente igual al producto ejemplo si no un producto parecido a este. Por tanto, estos sistemas deben permitir que los usuarios refinan sus búsquedas declarando o modificando algunas de los atributos del ejemplo dado.”³

Para obtener la información es necesario contar con un historial amplio de información disponible. Una solución tecnológica o un sistema de información que desee implementar los sistemas de recomendación, no debe ser un sistema de información relativamente nuevo. En caso de ser así, es requerido definir una serie de casos que permitan recomendar ítems dependiendo de poca información. Este tipo de sistemas de recomendación se especifican a continuación.

³ BARRANCO, M. J; PÉREZ, L.G. MARTÍNEZ, L. Un Sistema de Recomendación Basado en Conocimiento con Información Lingüística Multigranular. 423 p. 2005. Universidad de Jaén. Jaén, España.

5. TIPOS DE SISTEMAS DE RECOMENDACIÓN

Una forma de clasificar los sistemas de recomendación es según la metodología utilizada para construir el perfil de usuario. Se suelen clasificar por si utilizan preferencias explícitas, implícitas o una combinación de ambas para calcular las recomendaciones.

Se considera que un sistema de recomendación utiliza preferencias explícitas, cuando el usuario expresa sus preferencias conscientemente. Como por ejemplo en el caso de valorar un contenido, donde la valoración consiste en dar un valor numérico dentro de un rango definido.

Los sistemas que utilizan *feedbacks* implícitos deducen las preferencias a partir del comportamiento del usuario y de su historial. Esto permite que en la mayoría de los casos no sea necesario pedir al usuario gran cantidad de información sobre sus preferencias para que pueda ser recomendado.

En general, los sistemas de recomendación utilizan tanto preferencias explícitas como implícitas para construir los perfiles de usuario. De manera que inicialmente suelen utilizar métodos explícitos para obtener la información básica del perfil, para más adelante, a medida que el usuario interactúa con el sistema de recomendación, utilizar preferencias implícitas con el fin de refinar el perfil inicial y mejorar sus predicciones.

La forma más conocida de clasificar a los sistemas de recomendación es según la metodología que utilizan para resolver el problema de la recomendación. En la literatura encontramos básicamente cuatro metodologías distintas:

- Recomendación basada en el contenido.
- Recomendación basada en casos.
- Recomendación basada en filtro cooperativo.
- Recomendación híbrida.

5.1 RECOMENDACIÓN BASADA EN EL CONTENIDO

Consiste en buscar objetos que se adapten lo mejor posible a las preferencias del usuario. El elemento clave de esta metodología de recomendación es la medida de similitud que nos indica cuánto de relacionado está un objeto con un determinado usuario.

Los principales inconvenientes son: la sobre-especialización, que consisten en que las recomendaciones resultantes suelen ser siempre muy similares dado que se basan en la misma información; y que dependiendo del dominio de los objetos a recomendar, es difícil extraer buena información de estos para realizar un cálculo fiable de la similitud.

La ventaja respecto a los demás métodos es que se pueden realizar recomendaciones sin la necesidad de tener un historial previo, por lo que permite realizar predicciones independientemente del historial del usuario.

5.2 RECOMENDACIÓN BASADA EN CASOS

Consiste en recomendar objetos similares a los que el usuario ha valorado positivamente en el pasado. En esta metodología los elementos clave son la medida de similitud entre objetos, y la clasificación de los objetos del historial de usuario según una medida de la relevancia de cada objeto por el usuario.

Los principales inconvenientes son: la sobre-especialización, y la calidad de las recomendaciones va estrictamente relacionada con la cantidad de valoraciones que ha hecho previamente el usuario.

La ventaja principal es que la recomendación es independiente de cada usuario, y por lo tanto soluciona el problema de la poca densidad de información.

5.3 RECOMENDACIÓN COOPERATIVA

Consiste en recomendar objetos a un usuario basándose en las valoraciones de otros usuarios con un perfil similar. Las metodologías de filtro cooperativo se pueden dividir en dos grandes categorías.

- *Memory-based (user-based) collaborative filtering*: Son sistemas que utilizan heurísticos que realizan predicciones en base a toda la base de datos de usuario para generar una predicción.
- *Model-based (item-based) collaborative filtering*: Son sistemas que utilizan un modelo (creado a partir de la base de datos de valoraciones de usuario) para realizar las predicciones. Para construir este modelo se utilizan diversas técnicas de aprendizaje; las más usadas son las técnicas basadas en reglas.

Los algoritmos de filtro cooperativo basado en ítems tienen mejor rendimiento en dominios con gran escasez de datos. La combinación de ambas metodologías de filtro cooperativo (*memory-based y model-based*) puede ofrecer mejores resultados que utilizando únicamente una de las metodologías.

Los principales inconvenientes de estos métodos son: la baja densidad de información disponible, usuarios con gustos especiales son mal recomendados, un nuevo objeto no es recomendable hasta que no sea valorado por un número determinado de usuarios, y finalmente son poco escalables.

La ventaja principal es que solventan los inconvenientes de las metodologías basadas en contenido y en casos. Un ejemplo claro de sistemas que utilizan esta metodología es Amazon.

5.4 RECOMENDACIÓN HÍBRIDA

Para poder resolver las limitaciones de cada tipo de recomendación, en muchas ocasiones, se utilizan sistemas híbridos. Dependiendo de la forma en que

combinan las diversas aproximaciones los sistemas híbridos se pueden clasificar en:

- **Weighted:** sistema donde, dado un objeto, se combinan las diferentes puntuaciones de cada componente mediante una fórmula lineal.
- **Switching:** sistema que escoge un único componente dependiendo de la situación de recomendación.
- **Cascade hybrids:** sistema jerárquico donde el componente secundario refina las decisiones del primario.
- **Feature augmentation:** sistema que consiste en utilizar un componente para generar una nueva característica para cada objeto, aumentando así la información disponible para el componente primario.

5.5 VENTAJAS Y DESVENTAJAS ENTRE TIPOS DE SISTEMAS DE RECOMENDACION.

"Todas las técnicas de recomendación tienen sus puntos fuertes y débiles. De entre todos los problemas, podemos destacar quizás como el más importante el problema de la falta de datos cuando se empieza a utilizar el sistema, conocido en la literatura como el problema del ramp-up. Este término realmente se refiere a dos problemas diferentes, aunque relacionados:

- **Nuevos usuarios:** debido a que las recomendaciones son el resultado de la comparación entre el usuario objetivo y otros usuarios, basándonos solamente en anteriores interacciones de los usuarios con el sistema, un usuario del que se disponen pocos datos resulta difícil de clasificar, y por tanto, de recomendar.

- **Nuevos productos:** *análogamente, un nuevo producto del que no se dispone todavía de suficientes datos de acceso de los usuarios a éste, puede ser complicado de recomendar.*"⁴

⁴ ALBÍN RODRÍGUEZ, A. Sistema de recomendación colaborativo basado en algoritmos de filtrado mejorados. 182 p. 2006.

6. HERRAMIENTAS DE MONITOREO DE RECURSO

La base del modelo propuesto consiste en los datos generados por una herramienta de monitoreo de recursos disponibles en una maquina especifica. Estas herramientas, basan su información en la configuración de software y hardware que poseen, midiendo desde el uso de recursos de un proceso especifico, hasta el consumo de bytes de la red que alimenta la maquina.

Existen muchas herramientas de monitoreo, a continuación describimos aquellas que cumplen con los requisitos, previamente descritos, de implementación del modelo.

6.1 COMANDO PS EN LINUX

Linux es *“un sistema operativo multitarea y multiusuario, por lo tanto, múltiples procesos pueden operar simultáneamente sin interferirse unos con los otros.”*⁵

Cada proceso tiene la característica de simular que es el único proceso en el sistema y que tiene acceso exclusivo a todos los servicios del sistema operativo.

Los programas y los procesos son entidades distintas. En un sistema operativo multitarea, se presenta el caso donde si varios usuarios ejecutan un mismo programa al mismo tiempo, entonces el numero de instancias de ese programa equivale al número de usuarios ejecutándolo. Cada proceso posee un identificador único llamado PID (Process ID).

El comando PS, permite informar sobre el estado de los procesos ejecutados o instanciados en el momento de la ejecución del comando. Este comando está basado en el sistema de archivos /proc, es decir, lee directamente la información de los archivos que se encuentran en este directorio.

Al ejecutar el comando se obtiene un listado, dependiendo de las columnas que se especifiquen. A continuación algunas de las columnas.

⁵ EL RINCON DE LINUX. Descripción de Linux. [En Línea] <http://www.linux-es.org/sobre_linux> [citado el 03 de Julio 2011]

p o PID	Process ID , número único o de identificación del proceso.
P o PPID	Parent Process ID , padre del proceso
U o UID	User ID , usuario propietario del proceso
t o TT o TTY	Terminal asociada al proceso, si no hay terminal aparece entonces un '?'
T o TIME	Tiempo de uso de cpu acumulado por el proceso
c o CMD	El nombre del programa o comando que inició el proceso
RSS	Resident Size, tamaño de la parte residente en memoria en kilobytes
SZ o SIZE	Tamaño virtual de la imagen del proceso
NI	Nice, valor nice (prioridad) del proceso, un número positivo significa menos tiempo de procesador y negativo más tiempo (-19 a 19)
C o PCPU	Porcentaje de cpu utilizado por el proceso
STIME	Starting Time, hora de inicio del proceso
S o STAT	Status del proceso, estos pueden ser los siguientes: R runnable, en ejecución, corriendo o ejecutándose S sleeping, proceso en ejecución pero sin actividad por el momento, o esperando por algún evento para continuar T sTopped, proceso detenido totalmente, pero puede ser reiniciado Z zombie, difunto, proceso que por alguna razón no terminó de manera correcta, no debe haber procesos zombies

	<p>D uninterruptible sleep, son procesos generalmente asociados a acciones de IO del sistema</p> <p>X dead, muerto, proceso terminado pero que sigue apareciendo, igual que los Z no deberían verse nunca</p>
--	---

Tabla 1 - Columnas importantes del proceso PS en Linux⁶

Una vez el comando PS genera la información, podría utilizarse un comando adicional que escriba la información en un archivo de texto. Este archivo de texto representaría información suficiente sobre el estado de los procesos de una maquina Linux en un determinado tiempo.

6.2 COMANDO TYPEPERF DE WINDOWS

“El typeperf.exe es un proceso asociado a Microsoft® Windows® Operating System de Microsoft Corporation, hace parte de las librerías que presenta este sistema operativo. Este proceso se considera seguro. No es probable que represente un daño para su sistema.”⁷.

No se considera que este proceso sobrecargue la CPU. Sin embargo, la ejecución de demasiados procesos en su sistema puede disminuir el rendimiento del equipo donde se ejecuta.

- Esta herramienta se utiliza para escribir datos de rendimiento en la ventana de comandos o en un formato de archivo de registro admitido.
- Mostrar todos los contadores que estén disponibles en ese momento en un equipo local o remoto en particular.

Las opciones que presenta son:

```
typeperf { [counter ...]> | -cf | -q [object] | -qx [object] } [opciones]
```

⁶ ABOUT.COM. About "Linux PS" [En Línea]:

<http://linux.about.com/od/commands/l/blcmdl1_ps.htm> [Citado 15 Junio 2011]

⁷ UNIBLUE SYSTEMS LIMITED. Typeperf.exe. [En línea]

<<http://www.processlibrary.com/es/directory/files/typeperf/28668/>> [Citado 15 Junio del 2011]

Parámetros:

[counter ...]> Contadores de rendimiento que se supervisarán.

A continuación describimos las opciones disponibles para obtener una mayor cantidad de información al ejecutar el comando.

-?	Muestra la ayuda contextual.
-f	Formato de archivo de salida. De forma predeterminada es CSV.
==cf	Archivo que contiene contadores de rendimiento que se supervisarán; uno por línea.
-si <[[hh:]mm:]ss>	Tiempo entre muestras. El valor predeterminado es 1 segundo.
==o	Ruta del archivo de salida o base de datos SQL. De forma predeterminada es STDOUT.
-q [object]	Muestra contadores instalados (sin instancias). Para listar contadores de un objeto, incluya un nombre de objeto tal como Processor
-qx [object]	Muestra contadores instalados (con instancias). Para listar contadores de un objeto, incluya un nombre de objeto tal como Processor.
==sc	Cantidad de muestras para recopilar. De forma predeterminada se hará el muestreo hasta que se presione CTRL+C.
==config	Archivo de configuración que contiene opciones de comandos.
==s	Servidor al que supervisaré si no se especifica un servidor en la ruta de contadores.
==y	Responde Sí a todas las preguntas sin preguntar antes.

Tabla 2 – Opciones disponibles para la ejecución del comando typeperf.

7. COMPLEMENTOS DE MONITOREO DE LA HERRAMIENTA DE WINDOWS PERFMON

Al ejecutar el comando `typeperf.exe` con la opción “\?”, obtenemos un listado de todos los complementos disponibles que permiten realizar una medición de algún proceso o recurso del equipo. Entre los comandos disponibles encontramos:

- *\Proceso(explorer)\% de tiempo de usuario*
Este complemento se encarga de medir el tiempo de ejecución iniciado por un usuario, de un proceso determinado. Por cada proceso que se encuentra ejecutado, el `typeperf` puede medirlo utilizando este comando pero reemplazando el valor de “explorer” por el nombre del proceso.
- *\Cola de impresión(Microsoft XPS Document Writer)\Trabajos en cola*
Este complemento permite conocer los trabajos en cola en una cola de impresión de alguna impresora instalada en el equipo. Por cada impresora instalada, el comando se repite pero el valor del nombre de la impresora cambia. Además de conocer los trabajos en cola, también pueden medirse los trabajos errados, impresos y cancelados.
- *\Interfaz de red(Controladora Fast Ethernet)\Paquetes recibidos descartados*
Este complemento permite conocer el comportamiento de las interfaces o controladoras de red disponibles en la maquina. Además de medir los paquetes recibidos descartados, también es posible monitorear la información de bytes recibidos, total de bytes, paquetes enviados, paquetes recibidos, ancho de banda, entre otros.
- *\TCPv6\Errores de conexión*
Este complemento permite monitorear el comportamiento del modelo TCPv6. Entre sus funciones se encuentra medir, conexiones activas,

conexiones establecidas, conexiones reinicializadas, segmentos recibidos, segmentos enviados, entre otros.

- *\\Battery Status(ACPI\PNP0C0A\1_0)\Discharge Rate*

Este complemento permite medir el estado de la batería disponible en el equipo. En caso de que el equipo monitoreado no tenga una batería instalada o en funcionamiento, los valores presentados asimilan un comportamiento de una batería sin descarga o trabajando al 100%. Algunos de los valores que pueden ser medidos con otras opciones del complemento son: Velocidad de carga, carga, voltaje.

- *\\Disco físico(0 C: D:)\% de tiempo de disco*

Este complemento mide el comportamiento de una partición o disco físico del equipo. Entre las opciones que pueden ser monitoreadas se encuentran: Longitud promedio de la cola de disco, promedio de segundos de disco/transferencia entre otros.

La lista de complementos que permiten realizar un monitoreo del equipo varía dependiendo de la configuración del mismo, pero las variables estándares son las mismas para todos. Para ver un listado completo de los complementos disponibles, es necesario ejecutar el comando **“typeperf.exe \?”**.

8. IMPLEMENTACION DEL MODELO DE SISTEMA DE RECOMENDACION

Teniendo en cuenta los requisitos previamente descritos, la información que permite conocer la herramienta de monitoreo, la base teórica que se conoce sobre los sistemas de recomendación y el objetivo de la investigación, se puede llegar al siguiente diseño propuesto para el modelo de sistema de recomendación.

8.1 ARQUITECTURA DEL MODELO

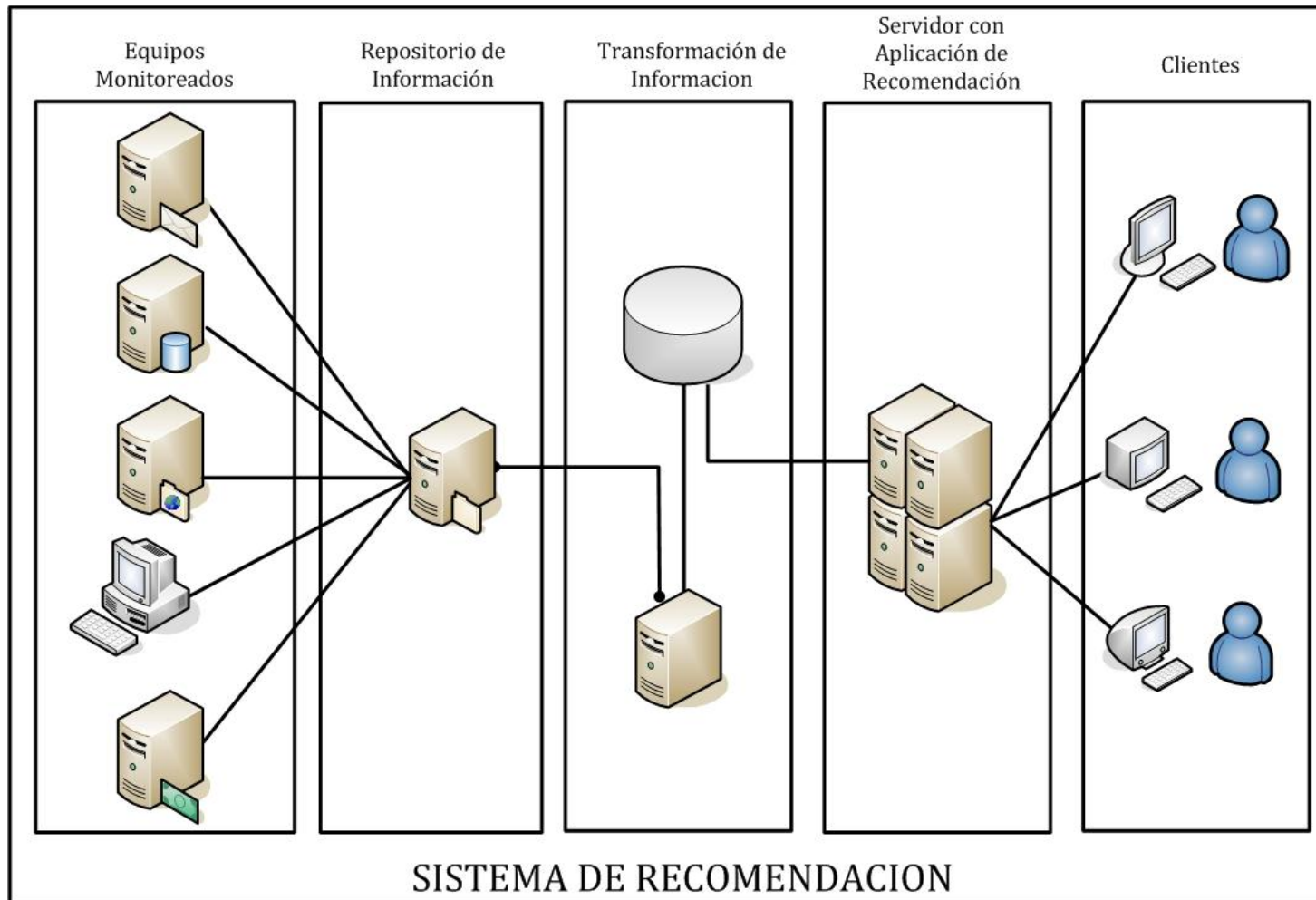


Ilustración 2 – Arquitectura de una solución del modelo.

8.2 DESCRIPCION DEL MODELO DE LA SOLUCION

El modelo está diseñado con el fin de optimizar la obtención de datos generados por las herramientas de monitoreo. Basándose en la arquitectura previamente descrita, se define el modelo de la siguiente forma:

8.2.1 Archivo ejecutable

En los servidores donde se desee monitorear el comportamiento de ciertos recursos o procesos, se genera un archivo ejecutable que se encargue de capturar los datos generados por el Perfmon.

Un ejemplo de este proceso se describe a continuación.

Se crea un archivo de extensión “.bat” que contenga la siguiente ejecución del comando:

- `typeperf -cf counters.txt -si 3 -sc 5 -o salida.csv -y.`

Donde “salida.txt” implica el archivo que contendrá el resultado de la ejecución del comando; -cf indica que los contadores están en un archivo que en este caso es “counters.txt” que contiene los complementos a medir. La opción “-sc” indica cuantos registros calculara, en esta caso 5; la opción “-si” indica el tiempo de monitoreo por acción, en esta caso 3 segundos.

Dentro de “counters.txt” se encuentran:

Indizador de búsqueda(SystemIndex)\Velocidad de combinación L1 - Promedio

Indizador de búsqueda(SystemIndex)\Combinación L1 - Recuento

Indizador de búsqueda(SystemIndex)\Velocidad de combinación L1 – Última.

8.2.2 Archivo de salida

Al ejecutar él “.bat” se obtiene la siguiente información en el archivo de salida:

Hora est. Pacífico, Sudamérica	Velocidad de combinación L1 - Promedio	Combinación L1 - Recuento	Velocidad de combinación L1 - Última
7/8/11 11:11:51 AM	3290713	1	3290713
7/8/11 11:11:54 AM	3290713	1	3290713
7/8/11 11:11:57 AM	3290713	1	3290713
7/8/11 11:12:00 AM	3290713	1	3290713
7/8/11 11:12:03 AM	3290713	1	3290713

Tabla 3 - Ejecución del typeperf con los complementos especificados.

8.2.3 Tarea Programada

Este archivo ejecutable contendrá entre sus funciones, escribir en un archivo de texto el resultado del monitoreo y una vez terminado el proceso, almacenar el archivo en una ubicación específica.

Para asegurar que el archivo se ejecute un número de veces en un determinado intervalo, se configura como una tarea programada.

El archivo plano que genera la tarea programada es almacenado en un repositorio centralizado, donde la comunicación con el sitio pueda ser a través de una cola de mensajería, una carpeta compartida, un registro dentro de una base de datos, entre otros.

8.2.4 Transferencia de archivos

Cuando el sistema interpretador de datos detecte que existe una nueva fuente, o en términos de la solución, un nuevo ítem, el sistema de información se encargara de procesar, transformar y registrar el dato para que sea interpretado por el sistema de recomendación.

Dicha transformación de la información generada se encuentra descrita en el punto 8.4 INTERPRETACION DE LOS DATOS GENERADOS.

Una vez la información se encuentra transformada y transferida al sistema de recomendación, una interfaz de usuario le permitirá al mismo crear marcas o

pautas que le permitan consultar la información que considere relevante o de forma rápida.

8.2.5 Suministro de información

El sistema de recomendación tendrá pautas previamente definidas que dependiendo de la información suministrada y de los posibles casos registrados, le recomiende al usuario, en este caso el administrador del sistema, posibles soluciones o medidas a tomar para optimizar el rendimiento de las maquinas a su cargo.

Por otro lado, el usuario suministrara información relevante de si la recomendación obtenida fue positiva o no, asegurando así obtención de la información de forma explícita por parte del usuario. Dicha información entrara al sistema de recomendación y será utilizada para futuras recomendaciones a otros usuarios.

8.3 TECNOLOGIAS UTILIZADAS EN EL MODELO

En el modelo propuesto se considera como mejor opción la técnica del Sistema de Recomendación Basado en Conocimiento, esta elección se realiza gracias a la naturaleza de los datos que se trabajan y al tipo de usuario al cual está dirigido esta aplicación.

Las ventajas que obtenemos con esta elección son:

- Sensibilidad a cambios en las preferencias.
- Puede incluir características que no tienen relación estricta con el elemento a recomendar.
- Capacidad de relacionar las necesidades del usuario con los productos a recomendar.⁸

Las desventajas que presenta haber escogido esta técnica son:

- No posee una capacidad de aprendizaje.

⁸ ALBÍN RODRÍGUEZ, A. Sistema de recomendación colaborativo basado en algoritmos de filtrado mejorados. 182 p. 2006.

- Se necesita conocimiento para interpretar la información que se está manejando.⁹.

Para garantizar el correcto funcionamiento del modelo, se utilizaron las siguientes tecnologías como planteamiento de la solución y de la arquitectura propuesta.

8.3.1 UNIDAD DE RED

Una unidad de red es un emulador de disco duro, funciona con algunas de las características de un disco duro externo. Se utiliza para prevenir problemas informáticos, ya que permite tener la información guardada externamente al computador, en un servidor especialmente dedicado a eso. Además, se puede utilizar como sistema de backup.

Como todos los discos duros, tiene una capacidad limitada, pero a diferencia de éstos, se le puede fijar un tamaño máximo de archivo al que admitir, sin importar de qué tipo sea éste. Se puede configurar de modo que el sistema transmita los datos cifrados y sólo se pueda acceder mediante una contraseña. Se puede acceder a estos servidores desde cualquier navegador conectándose a la dirección virtual pertinente. En algunos casos, también se puede acceder mediante un programa FTP seguro. El sistema operativo, en el cual ha sido nombrado lo ve como otra unidad o partición.

Algunas de las ventajas de esta tecnología son: reducción de costos, seguridad de la información, disponibilidad.

8.3.2 SAMBA

Samba configura directorios Unix y GNU/Linux (incluyendo sus subdirectorios) como recursos para compartir a través de la red. Para los usuarios de Microsoft Windows, estos recursos aparecen como carpetas normales de red. Los usuarios

⁹ ALBÍN RODRÍGUEZ, A. Sistema de recomendación colaborativo basado en algoritmos de filtrado mejorados. 182 p. 2006.

de GNU/Linux pueden montar en sus sistemas de archivos estas unidades de red como si fueran dispositivos locales, o utilizar la orden smbclient para conectarse a ellas muy al estilo del cliente de la línea de órdenes ftp. Cada directorio puede tener diferentes permisos de acceso sobrepuestos a las protecciones del sistema de archivos que se esté usando en GNU/Linux.

Por ejemplo, las carpetas home pueden tener permisos de lectura y escritura para cada usuario, permitiendo que cada uno acceda a sus propios archivos; sin embargo, deberemos cambiar los permisos de los archivos localmente para dejar al resto ver nuestros archivos, ya que con dar permisos de escritura en el recurso no será suficiente.

8.3.3 BASE DE DATOS DB2

“DB2 Universal Database es el sistema de administración de bases de datos que brinda una plataforma de base de datos flexible y efectiva en costos para construir aplicaciones robustas de negocios on demand”. Asimismo DB2 UDB promueve sus recursos con un amplio soporte para estándares abiertos y plataformas de desarrollo populares, tales como J2EE y Microsoft.NET. La familia de DB2 UDB también incluye soluciones adaptadas para necesidades específicas, tales como business intelligence y herramientas avanzadas.

- DB2 automatiza algunas tareas de administración de base de datos. Esto conlleva a que los administradores de bases de datos están enfocados en problemas propios del negocio.
- DB2 ofrece compresión de almacenamiento, por lo cual se necesita menos hardware para almacenar los datos, y así se reducen las necesidades de consumo de energía así como también menos espacio de disco.
- IBM ofrece un conjunto integrado de soluciones de administración de datos que facilitan la colaboración entre analistas, arquitectos, desarrolladores y administradores cuando trabajan con datos.¹⁰

¹⁰ IBM. Liberese de los costos de su base de datos. [En Línea] <<http://www-01.ibm.com/software/mx/db2/lowerdatabasecosts/>> [Citado el 16 de Junio de 2011]

Un esquema simple que muestra cómo funciona la base de datos db2 es el mostrado en la siguiente ilustración:

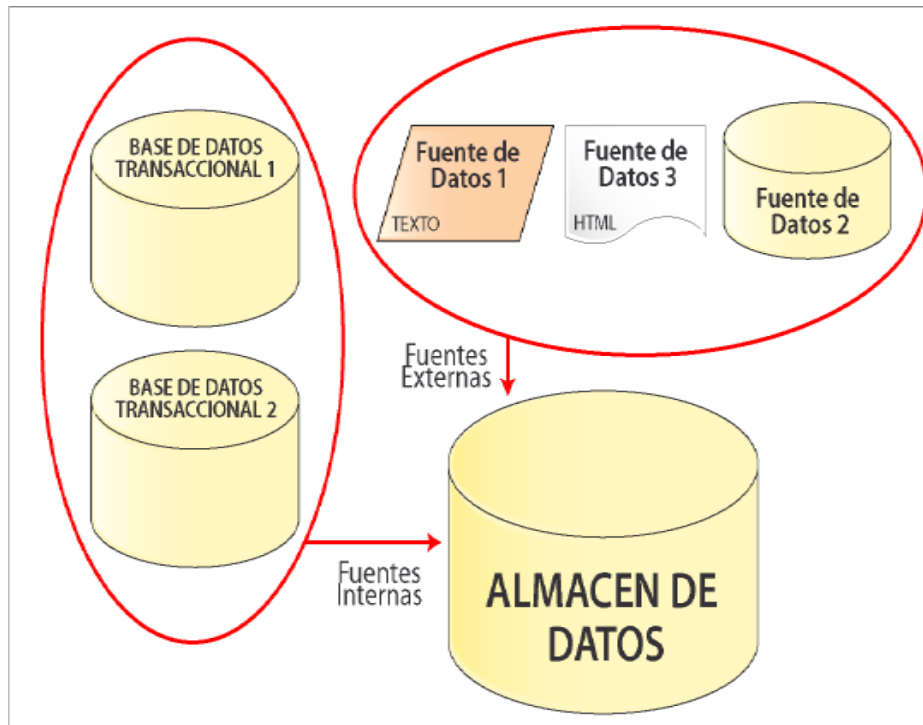


Ilustración 3 - Esquema simple del funcionamiento de Bases de Datos en DB2.

8.3.4 LENGUAJE DE PROGRAMACIÓN .NET

La escogencia de este lenguaje de programación es por la facilidad que brinda en la transformación de archivos e información. Por otro lado, el modelo está altamente orientado al sistema operativo de Windows y la tecnología .NET es la tecnología en programación más reconocida actualmente. El lenguaje .NET es utilizado en la solución del modelo, para desarrollar dos aplicaciones claves en el proceso: transformación de información y consulta de información.

8.4 INTERPRETACION DE LOS DATOS GENERADOS

Teniendo la información generada en el archivo “.csv” y conociendo de antemano los contadores que medirá la solución del modelo, se definen las estructuras de las tablas que contendrá la base de datos.

Como la información generada esta diferenciada por contador y fecha, entonces la alimentación de la misma será utilizando estos dos valores como llaves para agrupar la información. Como cada contador arroja valores numéricos, estos mismos valores pueden ser utilizados para definir un promedio en cantidad de datos en un intervalo de tiempo.

Por ejemplo, tomando como base la información previamente generada se obtienen los datos presentados en las siguientes dos tablas:

Hora est. Pacífico, Sudamérica	Velocidad de combinación L1 - Promedio	Combinación L1 - Recuento	Velocidad de combinación L1 - Última
7/8/11 11:11:51 AM	3290713	1	3290713
7/8/11 11:11:54 AM	3290713	1	3290713
7/8/11 11:11:57 AM	3290713	1	3290713
7/8/11 11:12:00 AM	3290713	1	3290713
7/8/11 11:12:03 AM	3290713	1	3290713

Tabla 4 - Promedio por componente de datos generados por el typeperf.

Se puede calcular que para el día “7/8/11” se obtuvo el siguiente resultado:

Hora est. Pacífico, Sudamérica	Velocidad de combinación L1 - Promedio	Combinación L1 - Recuento	Velocidad de combinación L1 - Última
7/8/11	3290713	1	3290713

Tabla 5 - Promedio por componente de datos generados por el typeperf.

Esto se obtiene de sumar los valores para cada columna y dividirlos por el número total de registros. Cada valor diario es un registro dentro de la base de datos.

8.5 RECOMENDACIÓN A BASE DE CASOS

Una vez se obtienen los datos y se registran los promedios diarios, se definen los posibles casos que se presentan en la toma de datos. Estos casos se definen dependiendo del correcto funcionamiento del equipo. Por ejemplo, si el equipo en su estado normal consume alrededor de 1,5 Gigabytes de RAM, de las 4 que posee, entonces se define el valor como “valor bandera” para realizar las recomendaciones.

Entonces, en el caso de obtener un valor superior al valor bandera entonces el sistema de recomendación, utilizara los casos definidos para enviarle la información al usuario. Para el ejemplo, las recomendaciones establecidas para ofrecerle al usuario podrían ser:

- Revisar el consumo de procesos en ejecución.
- Liberar procesos que no sean imprescindibles para el funcionamiento de la maquina.
- Asegurarse del correcto funcionamiento de las tarjetas de memoria disponible.
- Revisar procesos en ejecución que no sean necesarios o que no hayan sido inicializados por el usuario o por el sistema.

Cada una de estas recomendaciones poseen valoraciones que el usuario deberá establecer una vez solucione el problema o la advertencia generada. En caso de que a algún usuario le haya funcionado la recomendación dada, deberá especificar cual fue para así mismo cuando a otro usuario se le presente el caso, las sugerencias presentadas por el orden según la valoración que contengan.

8.6 APLICACIONES

Debido a el alcance del modelo sugerido, es posible identificar en que situaciones la aplicación del modelo es optima. El modelo nace con el fin de poder identificar que maquinas o servidores requieren de atención por parte del administrador del

sistema en cuanto a ejecución de procesos y consumo de recursos, además de permitirle administrar múltiples máquinas virtuales en distintas ubicaciones.

Una aplicación del modelo está basada en tecnologías en la nube, donde el administrador podrá monitorear un gran volumen de máquinas virtualizadas sin importar la ubicación de las mismas. Del mismo modo, con tecnología en la nube, las organizaciones que implementen el modelo pueden ofrecer a sus clientes un servicio administrador por ellos donde le hagan llegar al cliente las recomendaciones para optimizar el rendimiento de un producto.

Además de las aplicaciones mencionadas, las organizaciones en constante crecimiento pueden implementar la solución propuesta del modelo, como un modo de mantener una modernización tecnológica constante en cuanto a recursos de hardware o software de sus máquinas.

9. CONCLUSIONES

El modelo propuesto es una aproximación al uso de sistemas de recomendación que basándose en una información generada por los complementos de la herramienta Perfmon, le permitirá al usuario final de la solución, en este caso el administrador de sistemas, tomar una decisión dependiendo de las sugerencias arrojadas.

La solución propuesta está abierta a cualquier modificación que permita optimizar el consumo de recursos del modelo. Cambios en la infraestructura de red, lenguajes de programación, métodos de transferencia de archivo, motor de base de datos u obtención de la información que modifiquen el concepto principal del modelo, son aceptados.

El modelo no es exclusivo a un sistema operativo o lenguaje programación, pero vale recordar que el sistema operativo y la herramienta utilizada fueron escogidos por las ventajas en interfaz grafica, rendimiento en tecnologías de virtualización de maquinas y el hecho de que muchos de los clientes finales del modelo están familiarizados con el sistema operativo Windows.

Dada la magnitud del modelo, la cantidad de datos requeridos para optimizar los resultados y la cantidad de tiempo necesario para implementar una solución en base al modelo, se opto por simular una solución a manera de ejemplo, pero es claro que, en caso de ser necesario, una profundización en el modelo es completamente posible.

BIBLIOGRAFIA

- ABOUT.COM. About "Linux PS" [En Línea]: <http://linux.about.com/od/commands/l/blcmdl1_ps.htm>
- ALBÍN RODRÍGUEZ, A. Sistema de recomendación colaborativo basado en algoritmos de filtrado mejorados. 182 p. 2006.
- ALBÍN RODRÍGUEZ, A. Sistema de recomendación colaborativo basado en algoritmos de filtrado mejorados. 182 p. 2006.
- BARRANCO, M. J; PÉREZ, L.G. MARTÍNEZ, L. Un Sistema de Recomendación Basado en Conocimiento con Información Lingüística Multigranular. 423 p. 2005. Universidad de Jaén. Jaén, España.
- DEFINICION.DE. Definición de Conocimiento. [En línea] <<http://definicion.de/conocimiento/>>
- EL RINCON DE LINUX. Descripción de Linux. [En Línea] <http://www.linux-es.org/sobre_linux>
- HERRERA-VIEDMA, E.; PORCEL, C.; HIDALGO, L. Sistemas De Recomendaciones: Herramientas para el Filtrado de Información en Internet. Depto. de Ciencias de la Computación e Inteligencia Artificial. Universidad de Granada. 14 p. 2007
- IBM. Liberese de los costos de su base de datos. [En Línea] <<http://www-01.ibm.com/software/mx/db2/lowerdatabasecosts/>>
- RAMÍREZ CRUZ Y., ANAYA SÁNCHEZ H., GIL GARCÍA R., COBOS CASTILLO Y., "Un enfoque híbrido al Reconocimiento de Nombres de Entidades para el español". 235 p. 2004.
- UNIBLUE SYSTEMS LIMITED. Typeperf.exe. [En línea] <<http://www.processlibrary.com/es/directory/files/typeperf/28668/>>