

# **Detección de ocupación de participantes y asignación eficiente de ítems de trabajo en un ambiente de Gestión de Procesos de Negocios.**

**Juan J. Moreno**

Universidad Católica del Uruguay, Montevideo, Uruguay, [jmoreno@ucu.edu.uy](mailto:jmoreno@ucu.edu.uy)  
INTEGRADOC, Montevideo, Uruguay, [jmoreno@integradoc.com](mailto:jmoreno@integradoc.com)

**Marcelo Cordini**

Universidad Católica del Uruguay, Montevideo, Uruguay, [mcordini@ucu.edu.uy](mailto:mcordini@ucu.edu.uy)

**Cristian Mastrantonio**

Universidad Católica del Uruguay, Montevideo, Uruguay, [cmastran@ucu.edu.uy](mailto:cmastran@ucu.edu.uy)

**Martín Palatnik**

Universidad Católica del Uruguay, Montevideo, Uruguay, [mpalatni@ucu.edu.uy](mailto:mpalatni@ucu.edu.uy)

## **RESUMEN**

La disciplina de Gestión de Procesos de Negocios (BPM por sus siglas en inglés, Business Process Management) ha permitido optimizar considerablemente los procesos de las organizaciones, incorporando en algunos productos modelos de asignación eficiente de los ítems de trabajo a los participantes. Sin embargo, actualmente no se considera la ocupación y eficiencia de esos usuarios al momento de la asignación, lo que constituye una oportunidad de mejora y optimización para estas herramientas.

Este trabajo presenta el resultado unificado de tres investigaciones con un objetivo común: proveer un modelo completo para representar y preveer la ocupación de los usuarios de forma que la misma sea utilizada en una asignación óptima de los ítems de trabajo en un ambiente BPM. La metodología incluyó un análisis exhaustivo del estado del arte. Posteriormente un equipo compuesto de varios investigadores, desarrolló la solución al problema planteado. Este trabajo incorporó instancias de validación y verificación, comprobando su efectividad y factibilidad, incluyendo un prototipo utilizando una herramienta de BPM de clase mundial y código abierto.

**Palabras Clave:** Gestión de Procesos de Negocios, ocupación, asignación, ítem de trabajo, usuario.

## **ABSTRACT**

Business Process Management (BPM) discipline has allowed organizations to considerably optimize their business process, including in some products the functionality needed to efficiently assign work items to participants. However, nowadays these solutions do not consider user business and participant's efficiency when work items are assigned, what constitutes a major optimization and improvement opportunity for these tools.

This work presents the unified result of three researches with a common objective: providing a complete model to represent and forecast user business in order to optimize work item assignment in a BPM environment. The methodology included an exhaustive state of the art analysis. After this stage the solution was built by a team of several researchers in order to solve the described problem. This work included several validation and verification stages to probe its feasibility and effectiveness, including a prototype using a world class open source BPM tool.

**Keywords: BPM, occupation, assignment, work item, user.**

## **1. INTRODUCCIÓN**

La participación de los humanos en la automatización y gestión de procesos de negocio (BPM – Business Process Management) ha quedado relegada en los últimos tiempos en las investigaciones, enfocándose más en los procesos en sí, y en la integración de aplicaciones. “Los procesos no hacen el trabajo, las personas lo hacen” (en inglés “Processes don't do work, people do”) afirma Jonh Seely en el artículo “Las Personas son la compañía” (en inglés “People Are the Company”) (Seely, 2007), donde destaca el valor de las personas en las organizaciones, y afirma que los métodos informales, improvisados e inspirados de las personas para resolver problemas, pueden solucionar algunos que los procesos formales no pueden predecir.

En los sistemas de BPM es muy común que un mismo trabajo o documento en determinado estado pueda ser asignado a muchos participantes potenciales. Si se quisiera que esta asignación se realice a la persona que pueda prestarle atención lo antes posible y no a alguien que se encuentre saturado de trabajo, sería necesario saber cuál es el nivel de ocupación de un usuario de forma automática. También es útil este mecanismo para no interrumpir al usuario que está muy ocupado afectando su trabajo y su comportamiento.

Se pueden manejar varios elementos para definir el nivel de ocupación de un usuario, tanto pertenecientes al sistema de gestión de procesos de negocios (BPMS – Business Process Management System) como a cualquier otra actividad que el individuo pueda estar realizando. Por ejemplo, la cantidad de ítems de trabajo<sup>1</sup> que tenga asignado el usuario, la cantidad de operaciones realizadas en determinado tiempo, la cantidad de tareas que le han sido asignadas en determinado tiempo, la cantidad de letras por segundo que esta tecleando en una herramienta de Ofimática, la cantidad de sitios webs que visita, la cantidad de aplicaciones abiertas que tiene y la cantidad de llamados recibidos, entre otros.

Para distinguir que significa estar ocupado de no estar ocupado, se definen métricas que permiten comparar su nivel de ocupación con otros usuarios, y con otros estados de ese usuario en el tiempo.

Los problemas a resolver son los de adquirir los datos referentes a la carga de trabajo tanto dentro del ambiente de BPM como fuera de él, almacenar y analizar los datos recogidos de forma de inferir información y realizar predicciones, y por último, a partir de los datos de la carga de trabajo almacenados, de la información inferida y de las predicciones realizadas tomar decisiones de asignación del trabajo más inteligentes y llevarlas a cabo.

Estos problemas fueron abarcados en el marco de varias investigaciones de la Facultad de Ingeniería y Tecnologías de nuestra universidad, hallando soluciones innovadoras y completas a los mismos, las que se describen en este trabajo. En las primeras secciones se describen los subproblemas descritos, las soluciones encontradas para ellos y los mecanismos de validación y verificación empleados para asegurar su correctitud. Finalmente se resumen las conclusiones obtenidas del trabajo y el trabajo futuro que queda planteado.

## **2. DEFINICIÓN DEL PROBLEMA**

Detectar la ocupación de los usuarios del Sistema de Gestión de Procesos de Negocios automáticamente a partir de lo cual realizar una asignación más eficiente de los ítems de trabajo.

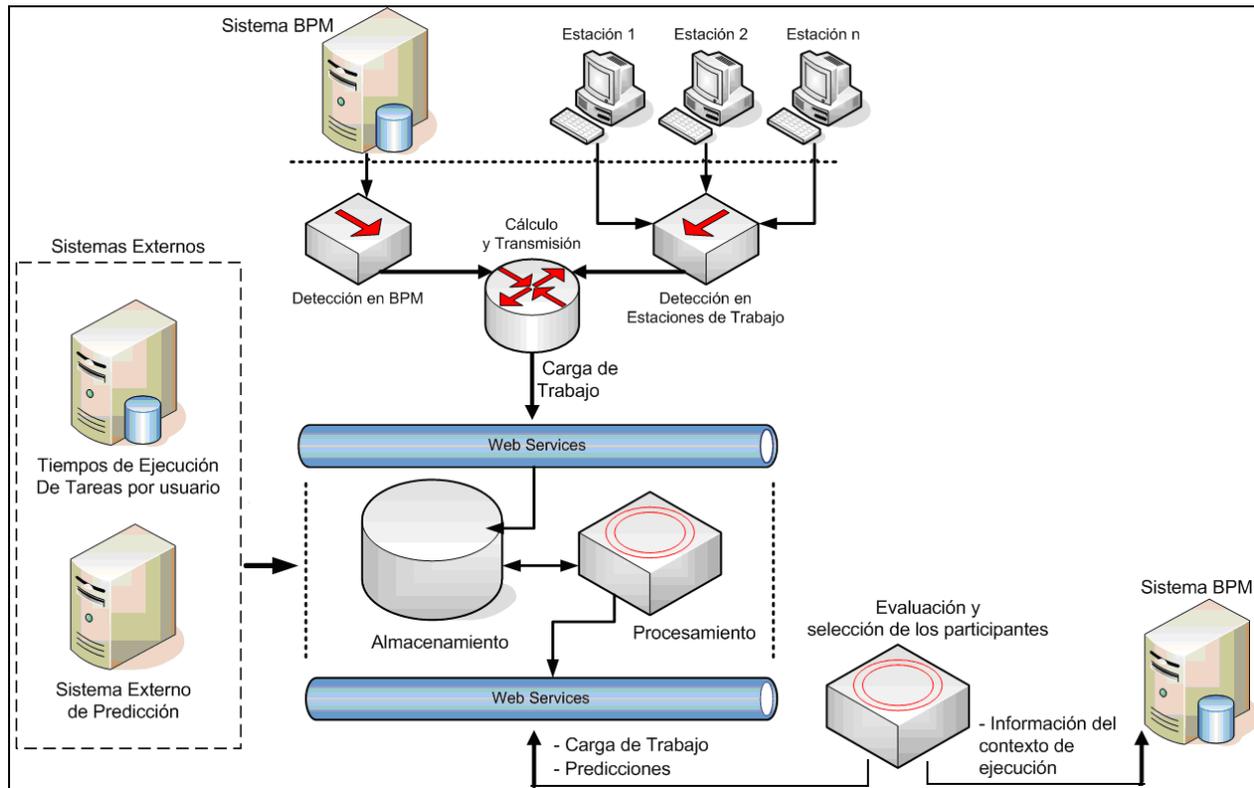
## **3. SOLUCIÓN PROPUESTA**

El problema se divide en 3 subproblemas, resueltos en módulos independientes e interconectados entre sí. El primero toma información de los usuarios y sus actividades en los Sistemas BPM, llamado **Módulo de Detección**, que además incorporará funciones para extraer información de los usuarios en las diversas acciones que realizan

---

<sup>1</sup> El concepto de ítem de trabajo es genérico en la disciplina de BPM, pudiendo corresponder tanto a tareas, actividades o cualquier otro elemento que sea asignado a un participante. En este trabajo se utilizarán estos términos como sinónimos.

con el computador (Detección en Estación de Trabajo<sup>2</sup>). En segundo lugar está el **Módulo de Cálculo y Transmisión**, que se encargará de recibir la información obtenida por el módulo anterior, calcular el puntaje de ocupación de los usuarios y transmitirlo al tercero: **Módulo de Asignación**. Éste es responsable de determinar si es posible optimizar la forma en que se asigna el trabajo en el BPMS (Figura 1).



**Figura 1 - Arquitectura de la Solución**

### 3.1 MÓDULO DE DETECCIÓN

Este módulo solicita información de las actividades de los participantes del sistema BPM al Módulo de Detección en BPM y al Módulo de Detección en Estación de Trabajo, para calcular la ocupación de los mismos.

La ocupación de un usuario va a depender de los ítems de trabajo que tenga asignados por el sistema BPM y de las actividades en general que esté realizando en este momento en su puesto de trabajo. Por lo tanto, cuando la ocupación de un usuario es solicitada, se podría tomar la cantidad de ítems de trabajo asignados que tiene y la cantidad de actividades que está realizando y asignarle un índice de ocupación.

A partir de lo planteado surgen algunos problemas, como por ejemplo que no es posible relacionar de antemano un número de ítems de trabajo con un índice de ocupación, y que dentro de una misma organización no todas las tareas o actividades son iguales. Por lo tanto, el modelo tiene que considerar que en determinadas organizaciones y procesos tener cierta cantidad de ítems de trabajo pendientes es demasiado y en otras puede llegar a ser normal. Para solucionar esto, se debe tener un Número de Comparación que sea propio de una organización, que sea definido por un experto del negocio o que se calcule de forma automática. Esto último se puede enfocar a la situación actual de las cargas de trabajo en la organización, tomando el promedio de la cantidad de ítems de trabajo asignados actualmente y comparando con la cantidad que tiene el usuario.

Por otro lado, el modelo debe ser capaz de discriminar las tareas por su costo, para que a la hora de calcular la ocupación de un usuario se tenga en cuenta la cantidad de tareas asignadas y la dificultad de las mismas.

<sup>2</sup> Se refiere al lugar donde el usuario trabaja con su respectiva computadora. Se abreviará como WS.

En (Aldaz, 2004), se describe como recomendar que decisiones tomar dentro de un sistema de BPM. La solución propuesta por Aldaz adaptó Razonamiento Basado en Casos (CBR – Case Based Reasoning) para los flujos de trabajo (Workflow), tomando como origen una instancia de proceso, y como destino un patrón, de manera que una función de similitud compare las variables de interés de la instancia con la de los patrones. En los patrones se definió variables con valores asociados y puntajes para cada decisión posible a tomar y para cada participante en esa actividad. A partir de esto, cuando se toma una instancia de proceso, se le asocia un patrón y el mismo recomienda qué decisión tomar y el usuario idóneo para la tarea. Así se elimina el problema del mapeo entre una instancia de proceso y toda la base de casos, lo cual es inapropiado en un ambiente de producción.

Para aplicar esta solución a esta investigación, en vez de tener el puntaje de que posible decisión tomar, dentro de cada patrón se puede tener un puntaje con la dificultad de realizar una tarea. De esta manera, cuando un usuario tiene una tarea pendiente de determinada instancia de proceso, este mecanismo le asociará un patrón mediante la función de similitud. Este patrón contendrá un puntaje de dificultad de la tarea a realizar y será utilizado en el cálculo de la ocupación del usuario.

### 3.1.1 CÁLCULO DE CARGA EN BPM

El Módulo de Detección de BPM envía al componente de CBR las tareas pendientes de los usuarios y las instancias de procesos de cada una de las tareas que recogió del Sistema BPM. El CBR alimentado por los patrones de tareas BPM envía el costo de cada una de las tareas que los usuarios tienen pendiente al componente de Cálculo de Carga en BPM. Obtenido el costo total de cada uno de los usuarios se calculará el promedio de todos los usuarios, en orden de obtener la relación con el Número de Comparación.

### 3.1.2 CÁLCULO DE CARGA EN LA ESTACIÓN DE TRABAJO

Una alternativa para calcular la carga en la Estación de Trabajo es analizar las acciones que realiza en el computador, que representadas como interacciones del humano con el computador, generan eventos que pueden ser estudiados en diferentes niveles de abstracción. Se destacan los eventos de Interfaz de Usuario porque nos brindan información útil sobre el comportamiento del usuario y son fáciles de capturar y analizar (Hilbert, 1999). Estos eventos pueden ser capturados de diversas formas y a partir de ello inferir las actividades que los usuarios están realizando. Por ejemplo, las técnicas de detección de secuencias buscan secuencias predefinidas de eventos en los eventos UI (User Interface) generados por los usuarios. Cuando se detecta una secuencia, se esta detectando un evento de más alto nivel, como el cumplimiento de una tarea o de un objetivo. Por lo tanto, analizar los eventos UI generados por el usuario podría ser otra forma de detectar la ocupación del mismo, ya que podemos inferir cual es la tarea que esta realizando. Existen varios proyectos (ejemplo Dragunov, 2005; Rath, 2008), en los cuales a partir de interacciones que realiza el usuario con el computador identifican que tarea está realizando.

La aplicación de las metodologías utilizadas en estos proyectos dejaría a disposición la información capturada de manera que se pueda detectar la tarea que el usuario esta realizando a un nivel de abstracción similar a la solución para el BPM, por tareas. Esto permite aplicar la misma metodología que se aplicó para el Cálculo de Carga en BPM: utilización de patrones que retorne el costo de la tarea y la aplicación de un Número de Comparación para medirse ante las tareas actuales de los otros usuarios de la organización.

Cuando llega una solicitud de ocupación de un usuario, el sistema consultará los eventos que ha generado el usuario últimamente y a partir de ellos inferirá las tareas que está realizando. Luego, con las tareas obtenidas se aplicará la función de evaluación de CBR para obtener el patrón correspondiente a esas tareas, y de esta forma el costo de realizar esa tarea. Luego se calcula el promedio de los costos de todos los usuarios de manera de tener un Número de Comparación (al igual que el Cálculo de la Carga en BPM), y por último se calculará la carga en la Estación de Trabajo con el Número de Comparación.

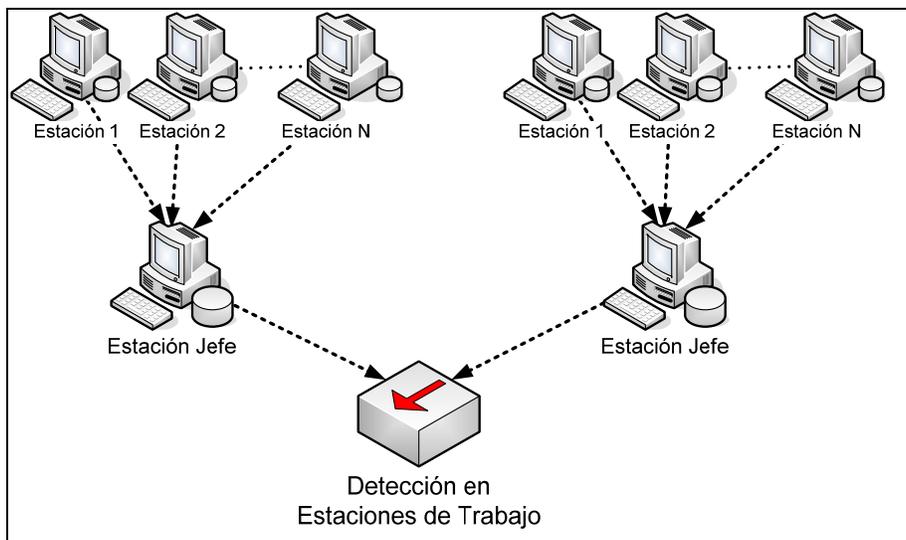
### 3.1.3 COMPONENTE DE DETECCIÓN EN BPM

Tiene como objetivo solucionar el problema de adquirir información relevante de las actividades y acciones de los usuarios en un Sistema BPM. Lo primero que se tiene que definir es de donde se va obtener la información necesaria de las actividades de los usuarios, y que técnica se va a utilizar.

Los BPMS almacenan la información de los procesos, las rutas definidas, los datos de los usuarios, los roles, las tareas, los permisos, en bases de datos propias desde donde también se obtiene información de las distintas instancias de proceso y de las tareas pendientes de los usuarios. De ellas se recolectará información sobre las tareas que el usuario tiene asignadas. Este tipo de solución es totalmente independiente del BPMS y no es necesaria configuraciones especiales para recolectar esta información.

### 3.1.4 COMPONENTE DE DETECCIÓN EN ESTACIÓN DE TRABAJO

Cada Estación de Trabajo deberá almacenar los eventos producidos y cuando sea solicitada la ocupación de los usuarios, el módulo solicitará a la Estación de Trabajo correspondiente los eventos de interacción que se hayan producido. En un ambiente de trabajo donde el número de estaciones sea grande, esta solución puede no ser adecuada en caso de que todos transmitan al mismo tiempo al mismo punto. Se propone jerarquizar la transmisión de estos datos en niveles de “Jefes”.



**Figura 2 - Detección en Estación de Trabajo Jerarquizada**

Así, las estaciones transmiten su información a una Estación “Jefe” y ésta al Módulo de Detección (Figura 2). La cantidad de niveles de “Jefes” variará en cada organización. Controlando la carga de transmisión generada, se detectarán los puntos de congestión a partir de los cuales agrupar o separar niveles para resolverlo.

## 3.2 MÓDULO DE CÁLCULO Y TRANSMISIÓN

El objetivo de este módulo es obtener un modelo que permita representar la carga de trabajo de los usuarios en un sistema de BPM & Workflow, con el fin de realizar predicciones a partir de la misma. Además, debe almacenar información acerca del tiempo de ejecución de las tareas para realizar predicciones también sobre este punto.

### 3.2.1 SOLUCIÓN PROPUESTA

Se definen cuatro componentes enfocados a proveer una solución completa para este módulo: modelado, predicción, procesamiento y comunicación. Respecto al primer componente, la información que debe “modelarse” proviene en principio de dos fuentes de datos heterogéneas. En primera instancia se encuentra la información acerca de la carga de trabajo de los participantes, que se calcula en el Módulo de Detección, la cual es utilizada en tiempo real aunque sobre la misma también se deben realizar predicciones. En segundo término, se encuentra la información acerca de tiempos de ejecución de las tareas por usuario, que es provista por un sistema externo y sólo es necesaria para realizar predicciones.

En función de lo anterior se decide diseñar una base de datos temporal para almacenar la información diaria de carga de trabajo (utilizada en tiempo real), y un Data Warehouse para los datos históricos sobre los que realizar predicciones (Figura 4).

En el caso del modelo para almacenamiento de información temporal, se escogió un modelo de Estrella. Si bien el modelo resultante consta de una tabla de hechos y de solamente una tabla de dimensión para la información de los participantes, esta solución garantiza un alto nivel de flexibilidad en la solución, un diseño adecuado para consultas rápidas y un mejor desempeño que otros modelos (Browning, 2001; Broda, 2007).

Para el almacenamiento de información histórica, la mejor aproximación resulta ser el diseño, desarrollo e implementación de un Data Warehouse mediante tecnologías abiertas, que facilite la tarea de almacenamiento y extracción de datos relevantes acerca de carga de trabajo y tiempo de ejecución de tareas (Figura 3).

Los Data Warehouse están optimizados para soportar la carga de información de a grandes cantidades, y lo están también para realizar consultas grandes, complejas e impredecibles que acceden a muchas filas por tabla (Browning, 2001). Otro punto de interés es la posibilidad que ofrecen los mismos para mantener información histórica a fin de poder realizar comparaciones y análisis sobre la misma, y la habilidad también para recolectar datos de diversas fuentes que quizás no están siendo consideradas y pueden ser potencialmente útiles, o de diversos sistemas bajo diferentes esquemas (Jansen, 2006 y Sugandhi, 2004).

A partir de buenas prácticas, de recomendaciones de diversos autores (Browning, 2001; Jansen, 2006; Sugandhi, 2004; y Lechtenbörger, 2001), y de un exhaustivo análisis de las mismas, se definió el modelo de datos solución para soportar la información mencionada y realizar la implementación del Data Warehouse propiamente dicho.

Para el componente de predicción de carga de trabajo y tiempo de ejecución de tareas por usuario, la mejor aproximación consiste en definir una interfaz mediante la cual un sistema experto de predicción pueda obtener datos históricos almacenados en el modelo, garantizando así un alto grado de precisión en las predicciones arrojadas, delegando esta tarea a un componente de software especializado a estos efectos.

En base a esta decisión, y dado el modelo definido en la presente investigación para almacenar datos históricos, resulta una buena elección entonces beneficiarse del hecho de que los datos históricos son almacenados en un Data Warehouse, y hacer uso de los mecanismos de extracción y exposición de datos que el mismo brinda mismo brinda (ejemplo: Cubos OLAP - Online Analytical Processing) para facilitar la tarea de interconexión con el sistema de predicción.

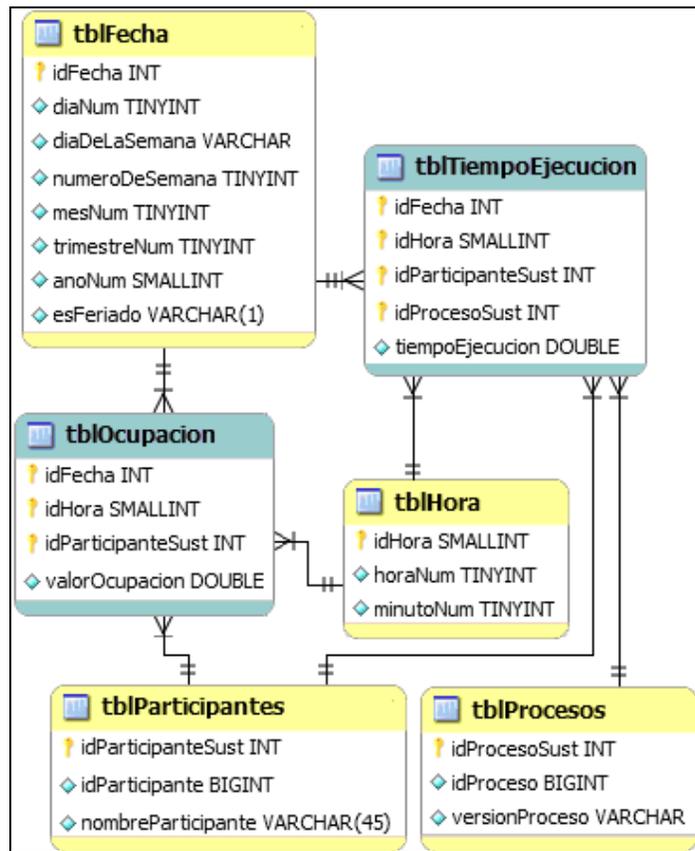


Figura 3 Modelo de datos para almacenar información histórica.

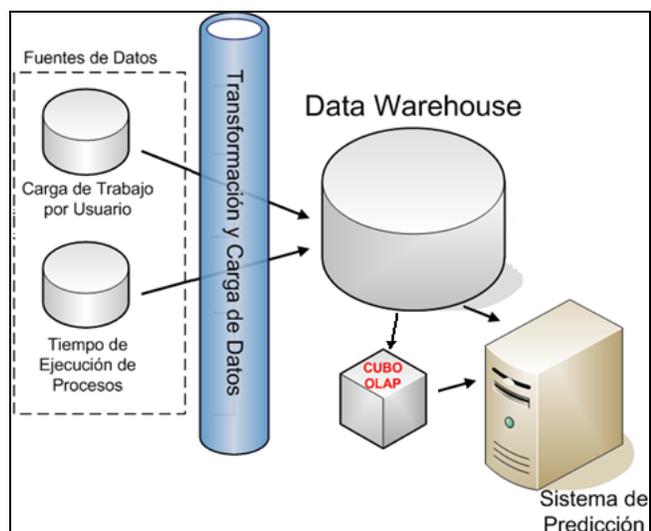


Figura 4 Modelo del Data Warehouse empleado.

Respecto al componente de comunicación, el cual se encarga de controlar que este proceso se desarrolle con normalidad y fluidez entre los módulos que componen la solución, se optó por implementar un mecanismo de comunicación sincrónico y abierto: Servicios Web (Web Services). Así se desacopla la solución de los demás sistemas y módulos, además de proveer un canal de comunicación abierto, empleando este estándar de facto actual para comunicaciones de estas características a través de Internet. Fueron definidas también ciertas primitivas de comunicación, necesarias para permitir la interacción y cooperación entre los módulos correspondientes al modelo solución, que a su vez utilizan determinados mensajes para que el proceso de comunicación sea exitoso. Estos mensajes fueron definidos mediante mensajes XML (eXtensible Markup Language) intercambiados en el momento que se invocan las primitivas definidas.

Por último el componente de procesamiento debe responder en tiempo y forma a las peticiones recibidas desde los otros módulos, y en él es donde se concentra la lógica de la solución correspondiente a este módulo. Invoca servicios del Data Warehouse para que se extraigan los datos de la base de datos temporal y sistemas externos, solicita servicios de predicción al sistema experto correspondiente y solicita información adicional al Módulo de Detección en caso que sea necesario. Este componente redefine el concepto de ocupación o carga de trabajo, para sintetizar la carga de trabajo de los participantes de forma que se asimile en gran medida a la carga de trabajo real y se adecue al contexto, el ambiente del problema y las necesidades del Módulo de Asignación. Si bien el Módulo de Detección mencionado anteriormente envía periódicamente valores acerca de la ocupación de los participantes, esta información es una captura del estado de los mismos en un tiempo  $t$ , y no considera situaciones como por ejemplo que el participante esté realmente ocupado pero en el momento de calcular su ocupación por alguna razón resulte baja, por lo que podría retornar un resultado equivocado.

### **3.3 MÓDULO DE ASIGNACIÓN.**

Para optimizar la asignación del trabajo en los BPMS se obtuvo un modelo-solución genérico que permite optimizar la forma en que los trabajos son asignados a los participantes en sistemas de BPM. Para garantizar una correcta automatización, el modelo tiene en cuenta factores como la confianza de los usuarios y la disminución de la pérdida de control sobre el sistema de los mismos.

#### **3.3.1 SOLUCIÓN PROPUESTA**

Existen varias investigaciones que han derivado en modelos de asignación inteligente de ítems de trabajo en entornos BPM. En (Kumar et al., 2002), (Velo and Macêdo, 2007; Veloso, 2006) y (Shen et al., 2003) se presentan modelos que se basan en determinar puntajes de adecuación de participantes a tareas en función de parámetros calculados a partir del BPMS. En (Reijers et al., 2007) se estudia la emulación del comportamiento de insectos para la distribución del trabajo en BPM. A su vez en (Ha et al., 2006) se utiliza la teoría de filas de espera para estimar la carga de trabajo y entonces decidir las asignaciones. Por último en (Geppert et al., 1998) se utiliza un enfoque de subasta para distribuir el trabajo. Todos estos modelos redefinen el modelo de asignación basado en roles. Esto genera un problema dado que los BPMS comerciales se basan en el modelos de roles. En cambio, el modelo presentado a continuación optimiza la asignación de trabajo en los BPMS y continúa utilizando el modelo de roles característico de los BPMS comerciales. La solución de este modulo se divide en tres componentes: integración, evaluación, y almacenamiento.

#### **3.3.2 COMPONENTE DE INTEGRACIÓN.**

Se encarga de integrarse con el BPMS. Es el que logra que el resto de la solución sea independiente del BPMS y totalmente reutilizable. Se integra con el componente de predicción y modelado independizando la solución de él.

#### **3.3.3 COMPONENTE DE EVALUACIÓN.**

Aquí se encuentra la lógica del proceso de distribución del trabajo, incorpora la *inteligencia* de la solución. Dado un evento este componente realiza las operaciones necesarias. En el caso de ser una asignación inicia el proceso de evaluación y selección de participantes utilizando el rol de los participantes para saber cuáles con los participantes aptos y una función de puntaje para determinar quién se encuentra en mejores condiciones.

#### **3.3.4 COMPONENTE DE ALMACENAMIENTO.**

Almacena las bitácoras relacionadas a la solución y que son esenciales para una correcta automatización.

### 3.3.5 FUNCIÓN DE PUNTAJE

Basada en parámetros recibidos de los demás componentes: carga de trabajo, estimación del tiempo de ejecución de las tareas y promedio de carga en el tiempo estimado de ejecución de las tareas. Se calcula para cada participante el puntaje que definirá a quién se asigna la tarea. Cada parámetro se uniformiza del 1 al 10 (mejor).

### 3.3.6 FUNCIÓN DE PUNTAJE LINEAL

La función de puntaje lineal toma estos tres parámetros y les asigna un coeficiente a cada uno de acuerdo a la importancia que tiene. Existen varios métodos actualmente (Reijers et al., 2007; Veloso and Macêdo, 2007; Kumar et al., 2002), que utilizan este enfoque pero con diferentes parámetros.

$$Puntajetotal(p) = puntajecarga(p) * a + puntajeproductividad(p) * b + puntajecargapromedio(p) * c$$

Estos coeficientes (a, b y c) pueden ser decididos por la organización donde se utilizará la asignación automática. Sin embargo, aquí se predefinen en función de la importancia que tienen para los autores. Se utilizarán coeficientes de forma tal que  $a + b + c = 10$  y que en consecuencia el puntaje total se encuentre en el intervalo 1 a 100.

Para el primer coeficiente (a) se toma un valor alto en relación al resto. Esto se debe a que como se describió en una sección anterior la carga de trabajo es un dato preciso porque es actual, es un dato que representa la realidad instantánea del sistema, por lo que constituye una foto del mismo en un determinado momento de lo que sucede en el sistema. Por ello es que le asigna un valor de 5.

Para el segundo coeficiente (b) se toma un valor intermedio en relación al resto. El valor es menor que el de la carga de trabajo porque la productividad se calcula en base a predicciones pero es mayor que el de la carga promedio de trabajo porque la productividad es un valor muy importante sobre el participante y aporta mayor información para una asignación eficiente que la carga promedio de trabajo. Por esto se le asigna un valor de 3.

Para el tercer coeficiente (c) se le asigna 2, en base a lo definido previamente, obteniendo la función siguiente:

$$Puntajetotal(p) = puntajecarga(p) * 5 + puntajeproductividad(p) * 3 + puntajecargapromedio(p) * 2$$

### 3.3.7 FUNCIÓN DE PUNTAJE BASADO EN LÓGICA DIFUSA

Basándose en (Shen et al., 2003) y considerando que la lógica difusa puede ayudar al experto de la empresa donde se utilice el sistema de asignación automática a describir las reglas de asignación se decidió incluir una alternativa basada en lógica difusa. Este sistema funciona también como una función que a partir de las 3 entradas puntaje de carga de trabajo, puntaje de productividad y puntaje de carga promedio obtiene una salida que es el puntaje global. La diferencia es que aquí en lugar de utilizar coeficientes se define un conjunto de reglas difusas.

Cabe destacar que al igual que en la función de puntaje anterior, se definió un sistema difuso con las reglas y conjuntos difusos que consideraba el autor, pero se espera que estos sean adaptados en cada organización.

Para cada variable de entrada (puntaje) se definieron tres conjuntos difusos (alto, bajo y medio) y tres conjuntos para la variable de salida: alto, bajo y moderado. También se definieron las siguientes reglas:

**Tabla 1: Reglas de la función de Puntaje basada en Lógica Difusa**

Premisas	Conclusiones
Si carga = alta y (productividad >= media o cargaPromedio >= media)	asignación = alta
Si carga = baja	asignación = baja
Si carga = media y productividad <= media y cargaPromedio <= media	asignación = baja
Si carga = media y (productividad = alta o cargaPromedio = alta)	asignación = media

En las reglas nuevamente se le dio prioridad al puntaje de la carga de trabajo. A diferencia con la función de la sección anterior aquí se tiene una expresividad mayor. Por ejemplo con la segunda regla se define sencillamente, incluso para alguien no informático que si la carga de trabajo es baja la propensión a asignarle sea baja.

#### **4. CONCLUSIONES**

Se planteó una solución novedosa al problema de la distribución de trabajo en los BPMS basándose no solo en la carga de trabajo interna al BPMS sino que también en la carga externa a él y teniendo en cuenta también la carga de trabajo futura y la productividad de los participantes. A su vez se innovó atacando también el problema desde el punto de vista de la interacción humano-computador y definiendo aspectos de la solución que contribuyan a una correcta automatización aumentando la confianza de los usuarios en el mecanismo.

En segundo lugar, se definió una solución más adecuada que la planteada en investigaciones académicas relacionadas ya que intenta aprovechar las características actuales de los BPMS apoyándose y no reemplazando la asignación basada en roles características de los BPMS comerciales. A diferencia del resto de las investigaciones se planteó la solución como abierta y modificable de acuerdo a las necesidades de las organizaciones donde sea implementada y extensible mediante sistemas externos como sistemas de predicción. A su vez, se planteó una solución escalable que permite responder a las peticiones de forma rápida, eficiente y segura.

En tercer lugar, se validó la solución implementándola en un BPMS comercial y de difusión mundial como es el JBoss JBPM, realizando el pasaje de la solución al prototipo de forma óptima y mostrando que la solución es válida y que puede ser implementada en BPMS disponibles en el mercado.

Se realizaron experimentos para validar la solución. Se midió por una muestra estadística que el tiempo de recolección de datos para calcular la ocupación de los usuarios estaba optimizado. Se validó mediante una muestra estadística que el modelo era seguro para realizar predicciones. Finalmente se obtuvo que tanto la implementación basada en lógica difusa como la implementación basada en la función lineal reducen ampliamente los tiempos de ejecución en relación a una implementación de selección aleatoria de los participantes. Incluso se encontró que la solución basada en la función lineal es levemente superior que la función basada en lógica difusa, lo cual no es concluyente, pero permite afirmar que ambas tienen un comportamiento aceptable. Por último se obtuvo también que las asignaciones realizadas por la solución son muy similares a las que realizaría un humano teniendo en cuenta posiblemente factores similares a la hora de asignar.

#### **5. TRABAJOS FUTUROS.**

Se propone completar el ciclo de optimización en la asignación de recursos, mediante la medición objetiva de los resultados, haciendo uso de herramientas de BAM (Business Activity Monitoring) y BI integradas al BPMS.

En relación a la detección reactiva de ocupación de usuarios, se propone la investigación en profundidad de técnicas de análisis de datos en el área de BPI (Business Process Intelligence) para mejorar la solución planteada en este documento. Otra línea es investigar la relación entre las tareas que un participante está realizando en la estación de trabajo y las que tiene asignado en el BPM (las tareas pueden o bien ser las mismas, una incluir a otra, o pueden no tener relación alguna); lo relevante es, en el caso que exista la mencionada relación, investigar cómo afecta este aspecto en el cálculo de ocupación. A partir de la detección de tareas en la estación de trabajo, es viable además desarrollar una investigación que, mediante la implementación de una herramienta de Workflow Mining aplicada a las tareas capturadas, permita identificar como realmente se están ejecutando los procesos y cuando esta ejecución diste del curso normal planificado.

Relacionado con el modelado y predicción, se propone perfeccionar el mecanismo de predicción a utilizar. En la presente investigación se ha hecho uso de un sistema experto, que aplica y funciona correctamente en situaciones genéricas y normales. Sin embargo, para obtener una mejor calidad en el resultado de las predicciones sería ideal tener en cuenta los factores, características y naturaleza de los sistemas de BPM y obtener una representación más afinada. Sería de gran utilidad contar con un sistema que en base a una cierta cantidad de atributos conocidos, y teniendo en cuenta las características de los flujos de trabajo, el ambiente de ejecución y otros parámetros, pudiese realizar estimaciones de carga de trabajo más acertadas y aproximadas al BPM en sí mismo.

En referencia a la asignación efectiva de tareas, se propone en primera instancia optimizar el modelo-solución de

asignación del trabajo agregándole más características de la asignación del trabajo, como ser prioridad en la ejecución de las tareas y experiencia en tareas relacionadas o similares. En segundo término se propone utilizar mecanismos de aprendizaje por refuerzo para optimizar los parámetros de la función lineal o del sistema difuso. Finalmente se plantea extender la solución al mecanismo basado en “push”, haciéndola genérica a todo BPMS.

Si bien aún no existen aplicaciones en producción que implementen los conceptos vertidos en este trabajo, dado el carácter realista y concreto de los mismos, se espera que las mismas existan en breve.

## REFERENCIAS

- Seely, John; Estee Solomon Gray: The People Are The Company, <http://www.fastcompany.com/magazine/01/people.html>, [Consulta: 29/10/07]
- Hilbert, David (1999). “Extracting Usability Information from User Interface Events” *Department of Information and Computer Science, University of California, Irvine, CA.*
- Aldaz, Guillermo (2004). “Algoritmos de extracción de conocimiento para procesos automatizados” *Memoria de Grado, FIT, Univesidad Católica del Uruguay (UCUDAL).*
- Dragunov, Anton (2005). “Tasktracer: A Desktop Environment To Support Multitasking Knowledge Workers” *In IUI '05: Proceedings of the 10th international conference on Intelligent user interfaces.*
- Rath, Andreas S. (2008). “Context-aware Knowledge Services, Personal Information Management” *PIM 2008, CHI 2008 Workshop, April 5-6, 2008, Florence, Italy.*
- Browning, Dave; Mundy, Joy. Microsoft Corporation (2001). “Data Warehouse Design Considerations” *Microsoft® SQL Server™ 2000 Resource Kit*
- Broda, Tal; Clugage, Kevin (2007) “Improving Business Operations With Real-Time Information: How to Successfully Implement a BAM Solution” *BPM.com* <http://www.bpm.com/improving-business-operations-with-realtime-information-how-to-successfully-implement-a-bam-solution.html> [Consulta: 20/02/2008]
- Jansen, Michel (2006) "Building data warehouses using open source technologies"*Draft version 197.*
- Sugandhi, Abhishek (2004) "Data Warehouse Design Considerations". *Course Seminar Report, Submitted in partial fulfillment of the requirements for the degree of Master of Technology - Department of Computer Science and Engineering, Indian Institute of Technology, Bombay, Mumbai.*
- Lechtenbörger, Jens (2001) "Data Warehouse Schema Design". *Dept. of Information Systems, University of Münster, Leonardo - Campus 3, D-48149 M`unster, Germany - Berlin.*
- Kumar, Akhil; van der Aalst, Wil & Weske, Eric (2002) Dynamic Work Distribution in Workflow Management Systems: How to Balance Quality and Performance. *Journal of Management Information Systems*, **18**, 157-193.
- Geppert, Andreas; Kradolfer, Markus & Tombros, Dimitrios (1998) Market-Based Workflow Management. *IJCIS*, **7**, 297-314.
- Ha, Byung-Hyun; Bae, Joonsoo; Tae Park, Yong & Kang, Suk-Ho (2006) Development of process execution rules for workload balancing on agents. *Data & Knowledge Engineering*, **56**, 64-84.
- Reijers, Hajo; Jansen-Vullers, Monique; Zur Muehlen, Michael & Appl, Winfried (2007) Workflow Management Systems+ Swarm Intelligence= Dynamic Task Assignment for Emergency Management Applications. *Lectures Notes in Computer Science*, **4714**, 125-140.
- Shen, Mixin; Tzeng, Gwo-Hshiung & Liu, Duen-Ren (2003) Multi-criteria task assignment in workflow management systems. *Proceedings of the 36th Annual Hawaii International Conference on System Sciences.*, 9.

Rodrigues Veloso, Rêne (2006) Distribuicao de Tarefas em Sistemas de Workflow com Base na Aptidao dos Recursos. In: Universidade Federal de Uberlandia.

Rodrigues Veloso, Rêne & Macêdo, Autran (2007) Balancing Quality and Performance of Task Distribution in Workflow Based on Resource Aptitude. *Lectures Notes in Computer Science*, **4489**, 281.

### ***Authorization and Disclaimer***

*Authors authorize LACCEI to publish the paper in the conference proceedings. Neither LACCEI nor the editors are responsible either for the content or for the implications of what is expressed in the paper.*