

Tipo de artículo: Artículo original

Temática: Inteligencia Artificial

Recibido: 31/10/2016 | Aceptado: 10/11/2016 | Publicado: 21/11/16

## **Instanciación de la técnica Replay P/C en el sistema XAVIA HIS. Elementos a tener en cuenta**

### ***Instantiation of the Replay P/C technique on the XAVIA HIS system. Elements to be considered***

**Arturo Orellana García<sup>1\*</sup>, Carlos Dávila Lima<sup>2</sup>, Ilsen León Herrera<sup>2</sup>, Ernesto Cabrera Viciano<sup>2</sup>**

<sup>1</sup>Universidad de las Ciencias Informáticas, Carretera a San Antonio de los Baños, km 2 ½, Boyeros, La Habana, Cuba.

<sup>2</sup>Ing. Ciencias Informáticas, Universidad de las Ciencias Informáticas, Carretera a San Antonio de los Baños, km 2 ½, Boyeros, La Habana, Cuba.

\* Autor para correspondencia: [aorellana@uci.cu](mailto:aorellana@uci.cu)

---

## **Resumen**

Los servicios sanitarios presentan disímiles retos, uno de los problemas que ocurren diariamente en los hospitales son los cuellos de botella. Estos provocan anomalías o eventualidades en el flujo de los procesos, afectando directamente la calidad de la atención a los pacientes. Las organizaciones de salud procuran aplicar un enfoque de gestión por procesos y el uso de diferentes tecnologías con el fin de mejorar su rendimiento. La minería de procesos es una tecnología para el análisis de procesos, la cual posibilita identificar características y problemas asociados a la ejecución de los procesos desde sistemas de información. El sistema XAVIA HIS de la UCI tiene las características necesarias para contribuir a la identificación de cuellos de botella aplicando minería de procesos. La presente investigación tuvo como objetivo desarrollar e incorporar una vista de análisis para identificar cuellos de botella desde el sistema XAVIA HIS. Para ello se seleccionó y personalizó la técnica Replay a log on Petri Net for Performance/Conformance sobre dicho sistema. La vista de análisis desarrollada e incorporada al sistema, permitirá realizar análisis de la ejecución de los procesos de las instituciones sanitarias. Se espera que a partir de la propuesta de solución se beneficien los profesionales y los pacientes de la institución hospitalaria, y así incrementar la calidad de los servicios que se prestan.

**Palabras clave:** cuellos de botella, minería de procesos, modelo de proceso, salud, sistema XAVIA HIS

## Abstract

*Sanitary services present various challenges, one of the problems that occur daily at hospitals are bottlenecks. This provokes anomalies and eventualities in the process flow, directly affecting the patient's attention quality. The health organizations procure to apply a process management focusing and the usage of different technologies to improve their performance. The process mining is a technology for the process analysis, which possibilities to identify characteristics and issues associated with the execution of the process from information systems. The XAVIA HIS system of the UCI has the necessary characteristics to contribute to the identification of bottlenecks applying process mining. The current investigation had as objective to develop and incorporate an analysis view to identify bottlenecks from the XAVIA HIS system. For this was selected and personalized the Replay a log on Petri Net for Performance/Conformance technique on the system. The analysis view developed and incorporated to the system, will allow making an analysis of the sanitary institutions execution processes. It's expected that the solution proposal benefits the professionals and patients of the hospital institutions, increasing that way the quality of the provided services.*

**Keywords:** bottlenecks, process mining, process model, healthcare, XAVIA HIS system.

---

## Introducción

La minería de procesos es una disciplina de investigación que permite descubrir, monitorear y mejorar procesos reales a partir de la extracción de conocimientos de los registros de eventos en los actuales sistemas de información. Estos registros de eventos son el resultado del almacenamiento de la información de las actividades que componen los procesos en un período determinado de tiempo (van der Aalst, 2011).

Para realizar análisis de minería de procesos es necesario hacer uso de herramientas orientadas a esta finalidad (ProM y Disco) (Claes & Poels, 2013; van der Aalst et al., 2009) las cuales generan modelos de la ejecución real de los procesos en sistemas automatizados. Estas herramientas presentan inconvenientes para la usabilidad y la comprensión de los modelos generados y se plantean estos problemas como desafíos en el Manifiesto de la minería de procesos (van der Aalst, et al., 2011), por lo que usuarios no expertos en el área se ven limitados a su uso.

El Centro de Informática Médica (CESIM) desarrolla un Sistema de Información Hospitalaria (*Hospital Information System*, HIS por sus siglas en inglés), el cual tiene como objetivo la informatización de los procesos hospitalarios en el nivel de atención secundario, permitiendo la gestión clínica y administrativa de los procesos médicos para elevar la

calidad de atención a los pacientes (Orellana, Pérez & Larrea, 2015).

En el CESIM se realizan investigaciones sobre los beneficios que brinda la aplicación de la minería de procesos en el sector de la salud (Orellana, Pérez & Larrea, 2015a; Orellana, Castañeda & Valladares, 2015; Orellana, Pereiras & Sosa, 2015; Orellana, Larrea & Pérez-Alfonso, 2015). Ejemplos de ello lo constituyen:

- Contribuir a los análisis para dedicar esfuerzos y recursos a priori a partir de los modelos obtenidos según registros históricos.
- Creación de equipos profesionales eficaces a partir de los registros de atención a pacientes.
- Análisis de las tendencias, comportamientos y eventos demográficos con el objetivo de hacer actualizaciones en programas sociales de salud.

Hasta el momento las investigaciones se han dirigido a la etapa de descubrimiento. El sistema XAVIA HIS tiene implementado una bitácora, como vía para recopilar información de la actividad de los usuarios y los procesos que ocurren en el sistema. En el sistema fue desarrollado un componente para la extracción y transformación de las trazas de ejecución en registros de eventos. El análisis de los resultados Orellana, Pérez & Larrea, 2015b; Orellana et al., 2015) de aplicar dicho componente evidenció que el sistema cuenta con las características necesarias para realizar análisis de la ejecución de los procesos. Además, se desarrollaron un conjunto de personalizaciones de complementos del marco de trabajo ProM para detectar variabilidad en la ejecución de los procesos del sistema XAVIA HIS. Su desarrollo contribuye a apoyar la toma de decisiones de importantes funciones gerenciales como la planificación, la gestión y el control de los recursos.

Luego de la aplicación de los componentes desarrollados en el sistema se detectaron las siguientes insuficiencias:

- El sistema tiene las características y los datos necesarios para realizar estudios de cuellos de botella, sin embargo no tiene la forma de mostrar explícitamente su existencia.
- A partir del sistema no es posible identificar la existencia de desviaciones con respecto al tiempo en la ejecución de los procesos.

Las técnicas de minería de procesos utilizadas en el marco de trabajo ProM son multiobjetivos, pueden ser empleadas para análisis de procesos en general. Para la identificación y análisis de cuellos de botellas es especialmente utilizada la técnica *Replay a log on Petri Net for Performance/Conformance* “Replay P/C”. La presente investigación tiene

como objetivo instanciar la técnica Replay P/C para el sistema XAVIA HIS, que permita la identificación de cuellos de botella en la ejecución de sus procesos.

## Materiales y métodos o Metodología computacional

“*Replay a log on Petri Net for Performance/Conformance*” es una técnica de chequeo de rendimiento y conformidad que posibilita identificar las desviaciones en el tiempo de cumplimiento de los diferentes procesos (Adriansyah, 2012). Está directamente relacionada con el análisis temporal. Además, mediante una escala de colores intuitiva, clasifica las actividades por colores según su desviación con respecto al tiempo medio de estancia de las transiciones.

Entre las características que definen esta técnica se encuentran:

- Detectar problemas.
- Extraer información de tiempo.
- Detectar cuellos de botella.
- Diagnosticar, predecir y recomendar.
- Generar modelo integrado mostrando los tiempos, frecuencias, etc.

El fin de la propuesta de solución es mejorar el entendimiento y usabilidad de los modelos para los usuarios no expertos en el área de minería de procesos. Para ello es necesario que el *plugin* desarrollado muestre de forma intuitiva e inteligible la formación de cuellos de botella en la ejecución de los procesos que están definidos en el sistema XAVIA HIS.

Este *plugin* requiere como entradas una *Petri Net/Inhibitor/Reset/ResetInhibitor Net* y un registro de eventos para crear alineaciones avanzadas (Adriansyah, et al., 2012) entre cada traza en el registro y la red. Un análisis documental realizado sobre la técnica arrojó que no posee dependencias con otro *plugin*. El resultado de la técnica muestra todas las alineaciones entre el registro de eventos y la red a partir de la visualización “*Projects Alignments to Log*”.

Con el uso del atributo de evento “*time:timestamp*” que indica fecha y hora, la visualización por defecto “*Performance projection to model*” proyecta información de rendimiento en el modelo original. Utiliza un esquema de colores sin fisura desde verde (valor bajo) hasta rojo (valor alto). Por defecto, el color de transición muestra el tiempo medio de permanencia de las transiciones, el lugar del color muestra el tiempo promedio de espera en los lugares, y el espesor del arco muestra la frecuencia de los *tokens* que pasan. Sin embargo, este esquema de colores también puede ser modificado. Permite separar además casos finalizados rápidamente de casos que requieren mucho

tiempo para completarse.

Al aplicar la técnica “*Replay a log on Petri Net for Performance/Conformance*” se obtienen otras visualizaciones para realizar análisis de chequeo de rendimiento y conformidad desde varias perspectivas. A continuación se realiza una breve descripción de cada una de ellas.

## Tecnologías y herramientas

El HIS está desarrollado con tecnologías libres, por tanto, se propone el uso de las tecnologías y herramientas siguientes para la implementación, además cumplen con las políticas de independencia tecnológica definidas en Cuba para la informatización de la sociedad cubana. Como lenguaje de programación se utilizó Java, el mismo permitirá la creación e implementación de las clases controladoras. Como Entorno Integrado de Desarrollo Eclipse 3.4.2, para el desarrollo del componente se le integró el marco de trabajo JBoss Seam 2.1.1. Como marco de trabajo se utilizó Hibernate 3.3 pues es un marco de trabajo de persistencia para Java de libre distribución que facilita el mapeo de atributos entre una base de datos relacional y el modelo de objetos de una aplicación.

## Resultados y discusión

Se deben agregar al servidor JBoss 4.2.2 un conjunto de librerías que posibiliten el funcionamiento de la vista de análisis desarrollada.

Librerías incluidas en el paquete PNetReplayer que deben ser añadidas al servidor:

- axis.jar
- bsh-2.0b4.jar
- collections-generic-4.01.jar
- colt.jar
- commons-compress-1.0.jar
- FilterableSortableTablePanel.jar
- flanagan.jar
- jargs.jar
- jcommon-1.0.16.jar
- jlfg-1\_0.jar
- jung-algorithms-2.0.jar
- jung-api-2.0.jarjung-graph-impl-2.0.jar
- jung-io-2.0.jar
- jung-visualization-2.0.jar
- slickerbox1.0rc1.jar
- Spex.jar
- TableLayout-20050920.jar
- weka.jar
- xpp3-1.1.4c.jar
- xstream-1.3.1.jar
- Log.jar
- LogDialog.jar
- Widgets.jar
- guava-16.0.1.jar
- jas-plotter-2.2.jar
- junit-4.8.1.jar
- openide-lookup-1.9-patched-1.0.jar
- jgraph.jar
- simmetrics.jar
- trove-3.0.3.jar
- ProM-Contexts.jar
- ProM-Framework.jar
- ProM-Models.jar
- ProM-Plugins.jar
- Uitopia.jar
- UITopiaResources.jar
- javailp-1.2a.jar
- lpsolve55j.jar
- BasicUtils.jar
- PetriNets.jar
- EfficientStorage.jar
- jbp-0.2.429.jar
- OpenXES-XStream.jar
- OpenXES.jar

Librerías necesarias de añadir al servidor para ejecutar la técnica mediante la vista de análisis desde el sistema XAVIA HIS:

- freehep-graphics2d-2.1.1.jar
- freehep-graphicsio-2.1.1.jar
- freehep-graphicsio-svg-2.1.1.jar
- freehep-io-2.0.2.jar
- freehep-util-2.0.2.jar
- freehep-xml-2.1.1.jar
- InductiveMiner.jar: en esta librería se encuentra el algoritmo *Inductive Miner-infrequent*.
- ProcessTree.jar: el algoritmo *Inductive Miner-infrequent* mina el registro de eventos y genera un Process Tree, el cual es convertido a una *Petri Net* posteriormente.
- PNetAlignmentAnalysis.jar: este paquete contiene todas las clases necesarias para dar colores a la *Petri Net* y añadirle la información de los tiempos asociados a las actividades representadas.
- Replayer.jar: es una adaptación del paquete de clases PNetReplayer.jar de la técnica “Replay P/C” debido a que este presenta un conjunto de clases, métodos e interfaces de usuario que no son necesarios para realizar la identificación de cuellos de botella desde el sistema XAVIA HIS.
- Properties.jar: establece las propiedades de la imagen con formato *svg* que se genera.
- jgrapht-jdk1.6.jar: generara el grafo que se construye a partir de la *Petri Net*.

Para agregar la vista de análisis se implementaron las clases *ReplayControl*, *ManifestPerf* y *Actividad*. La clase *Actividad* contiene el atributo *id* el cual guarda el nombre de la actividad y el atributo *tiempos* se encarga de almacenar los valores mínimo, máximo, promedio y desviación típica asociados a los tiempos de ejecución, espera y estadía de la actividad en cuestión.

La clase *ManifestPerf* es una adaptación de la clase *ManifestPerfPanel* que se encuentra en el paquete *PNetAlignmentAnalysis*, se le realizaron modificaciones debido a que incluía atributos y métodos innecesarios para su aplicación en el entorno del sistema XAVIA HIS, por la necesidad de las modificaciones se agregó como una clase nueva y es utilizada para dar color a la *Petri Net* obtenida luego de ejecutar el algoritmo *Inductive Miner-infrequent*.

La clase *ReplayControl* contiene el atributo *manifestPerf* que es una instancia de la clase *ManifestPerf*. Esta clase almacena el manifiesto con todos los datos necesarios para su visualización. Recibe por parámetros el registro de eventos generado por el Componente para la extracción y transformación de trazas del sistema XAVIA HIS con el fin de realizar todas las operaciones necesarias para obtener el manifiesto. Además posee las instancias de las clases *IMMiningDialog* y *ProcessTree* de la técnica *Inductive Miner-infrequent*, así como las instancias de las clases *PNManifestReplayer*, *PNManifestReplayerParameter*, *PNManifestReplayerILPAlgorithm*, *Manifest* y *PerfCounter* de

la técnica “*Replay P/C*”

Para la integración con el sistema XAVIA HIS:

1. Se añadieron al sistema los paquetes de clases *PNetAlignmentAnalysis.jar* y *Replay.jar*, este último es una modificación de *PNetReplayer* debido a que el original contiene clases que no son necesarias para realizar el análisis y la detección de cuellos de botella. *Replay.jar* contiene las clases de la técnica “*Replay P/C*” necesarias para ser utilizadas posteriormente en la construcción del manifiesto, mientras que *PNetAlignmentAnalysis* contiene las clases para dar color a la *Petri Net* y visualizarla en el entorno del sistema XAVIA HIS.
2. Se creó un paquete (gehos.almacen.session.procesos.replay) con las clases: *Actividad*, *ManifestPerf* y *ReplayControl*.
3. A partir del Componente para la extracción y transformación de trazas que posee el sistema XAVIA HIS, se obtiene un registro de eventos con todos los datos necesarios para aplicar la técnica “*Replay P/C*”.
4. En la vista de la página Analisis\_Proceso.xhtml es posible seleccionar el rango de fechas y del proceso a analizar, estos son recibidos por la clase controladora Analisis\_Proceso.java la cual a partir de estos datos genera el registro de eventos haciendo uso del Componente para la extracción y transformación de trazas. Esta vista de análisis se encarga de acceder al modelo y utilizando las tecnologías *Hibernate*, *Enterprise Java Bean* y *Java Persistence API* extraer de la base de datos del sistema los datos necesarios para generar el registro de eventos.
5. Luego en la clase Analisis\_proceso.java se instancia la clase *ReplayControl.java* para a partir de ella generar mediante el método *init()* la *Petri Net* que representa las actividades con una escala de colores intuitiva para el usuario y la lista de actividades que contiene los datos de los tiempos asociados a estas.
6. Al hacer uso de las funcionalidades *Attribute()*, *getClassFor()* y *createCaseFilter()* implementadas en la clase *ReplayControl* se pueden generar las métricas y atributos necesarios para obtener el manifiesto, y la funcionalidad *listaActividades()* que devuelve la lista de las actividades representadas en la *Petri Net* con sus respectivos datos de tiempos asociados.
7. Se visualiza en la página *Replay.xhtml* la *Petri Net* que puede cambiar en consecuencia de la perspectiva temporal que se seleccione a partir de los radio buttons mostrados. La página permite visualizar además una tabla con los datos de los tiempos asociados a la actividad que se escoja en un Combo Box que se encuentra debajo de la imagen de la *Petri Net*, todo esto gracias al uso de los componentes de richfaces *rich:dataTable* y

rich:comboBox respectivamente.

### **Métricas configuradas para la instanciación**

Al aplicar la técnica de minería *Mine a Petri Net with Inductive Miner* y la selección de la variante *Inductive Miner-infrequent* es posible modificar el valor correspondiente al umbral de ruido para a partir del registro de eventos, ya previamente generado en *Xlog*, producir un árbol de proceso. Se mantuvo por defecto el valor del umbral de ruido de un 20 por ciento (por recomendación del autor de la técnica), con el objetivo de hacer más fácil el procedimiento del modelado del proceso.

Cuando se aplica la técnica “*Replay P/C*” se selecciona que tipo de clasificador de eventos se va a utilizar para obtener la lista de clases de evento y los patrones comprometidos. Entre *Event name*, *Resource*, y *MXML Legacy Classifier*, este último es el seleccionado por defecto. Como característica, este clasificador permite tener en cuenta los valores de *start-complete* para el dato *lifecycle:transition* perteneciente a los registros de eventos.

La asignación de las transiciones a los patrones puede ser realizada a partir de dos clasificadores de transiciones: *ID-based Classifier* y *Label-based Classifier*. Se selecciona por defecto *Label-based Classifier* como clasificador. El algoritmo utilizado para construir las alineaciones es el ofrecido únicamente por la técnica, *A\* ILP-based manifest replay* permite reproducir la construcción de la manifestación de secuencias de patrones de clases de eventos mediante una función heurística. Otras de las métricas configurables son la cantidad de estados a explorar, con valor por defecto de 2000 y los costos asociados al tránsito por el modelo dentro de cada una de las transiciones y eventos, con valor 1; excepto valor 0 para las transiciones que representan ruido generadas por *Inductive Miner-infrequent* en el descubrimiento de la *Petri Net*.

### **Punto de Entrada: generación del registro de eventos**

En el sistema XAVIA HIS los eventos se registran automáticamente y de manera sistemática y confiable a partir del motor de flujo jBPM. Se da soporte de manera explícita a nociones tales como instancia de proceso (caso) y actividad. Se define el registro de eventos en el nivel 4 de madurez, por lo que al aplicarle técnicas de minería de procesos se pueden obtener resultados.

A partir del Componente para la extracción y transformación de las trazas de ejecución, se obtiene un fichero de extensión .xes, el cual contiene la información del registro de eventos que permitirá aplicar la técnica “*Replay P/C*” en el sistema XAVIA HIS y se almacena temporalmente a partir de la sentencia `this.template_dir_tempLogs = dir_real_path + "/modAlmacen/Logs/tempLogs/";`

### **Obtención de una Petri Net**

Una vez preparado el punto de entrada de la herramienta, se procede a la realización del modelado de la *Petri Net* correspondiente. Esto comienza con la realización de una instancia de la librería `IMMiningDialog` que recibe como parámetro el *XLog* generado. Posteriormente la clase `ProcessTree` se encarga de almacenar el árbol de procesos que se obtiene de aplicar el método `IMProcessTree.mineProcessTree`. La variable *pn* almacena la *Petri Net* con marcas que se obtiene de convertir el árbol de procesos (*tree*) mediante la utilización de la funcionalidad `convert` de la clase `ProcessTree2Petrinet`. A continuación se expone el código en el lenguaje Java utilizado para obtener la *Petri Net*.

```
IMMiningDialog dialog= new IMMiningDialog(xLog);
ProcessTree tree = IMProcessTree.mineProcessTree(xLog,
dialog.getMiningParameters());
ProcessTree2Petrinet.PetrinetWithMarkings pn = null;
pn = ProcessTree2Petrinet.convert(tree);
```

### Obtención del manifiesto

En este caso se utiliza la palabra manifiesto para nombrar la salida del paquete *PNetReplayer*, que se conforma con varios parámetros de entrada además del *XLog* y la *Petri Net* correspondiente. Al obtener la *Petri Net* del método *Inductive Miner-infrequent* se generan las marcas de inicio y de fin, que son utilizadas como parámetros. Se selecciona el algoritmo *A\* ILP-based manifest replay* por defecto utilizado para realizar el chequeo de rendimiento. Posteriormente al instanciar un objeto de tipo `PNManifestReplayerILPAlgorithm` es posible aplicar el método `replayLog` para almacenar el resultado en una variable de tipo `Manifest`, entrada para poder aplicar `PNNetAlignmentAnalysis` funcionalidad encargada de obtener la visualización “*Performance Projection to Model*”, para la identificación a través de una vista de análisis que muestre una *Petri Net* con los posibles cuellos de botella.

```
PNManifestReplayer replayer= new PNManifestReplayer();
PNManifestReplayerParameter parameter= new PNManifestReplayerParameter();
Marking [] finalMarkings= new Marking [1];
finalMarkings [0]= pn.finalMarking;
Object [] algAndParam=replayer.chooseAlgorithmAndParam(null, pn.petrinet,
xLog, pn.initialMarking, finalMarkings);
parameter=(PNManifestReplayerParameter) algAndParam [1];
PNManifestReplayerILPAlgorithm alg= (PNManifestReplayerILPAlgorithm)
algAndParam [0];
Manifest manifest = alg.replayLog(null, pn.petrinet, xLog, parameter);
```

### Visualización de los resultados

En la visualización de los resultados se muestra una *Petri Net* y una tabla con los tiempos asociados a cada actividad de proceso. La *Petri Net* descubierta puede ser obtenida en tres perspectivas diferentes: tiempo de estadía, tiempo de espera y tiempo de procesamiento. Para identificar los posibles cuellos de botella, al igual que la técnica de *ProM*, esta vista de análisis utiliza la escala de colores definida en la leyenda mostrada y el grosor de los conectores para

describir la frecuencia de trazas. A cada actividad de proceso le corresponde una transición (rectángulo) y dentro de cada transición se puede observar el tiempo correspondiente a la perspectiva seleccionada. El sistema permite expandir, contraer y arrastrar el modelo. Por su parte, la tabla que se visualiza en la vista de análisis es la equivalente a la tabla que se muestra en el panel Inspector de ProM, obtenido por la técnica “Replay P/C”. Para visualizar los indicadores, primeramente se selecciona la actividad a analizar y a partir de esta se genera la tabla correspondiente. La estructura de la tabla está diseñada para que a cada fila le corresponda un indicador; cada columna es asociada a los valores máximos, mínimos, promedio y desviación típica. La Figura 1 muestra la interfaz de usuario de la vista de análisis en el sistema XAVIA HIS.

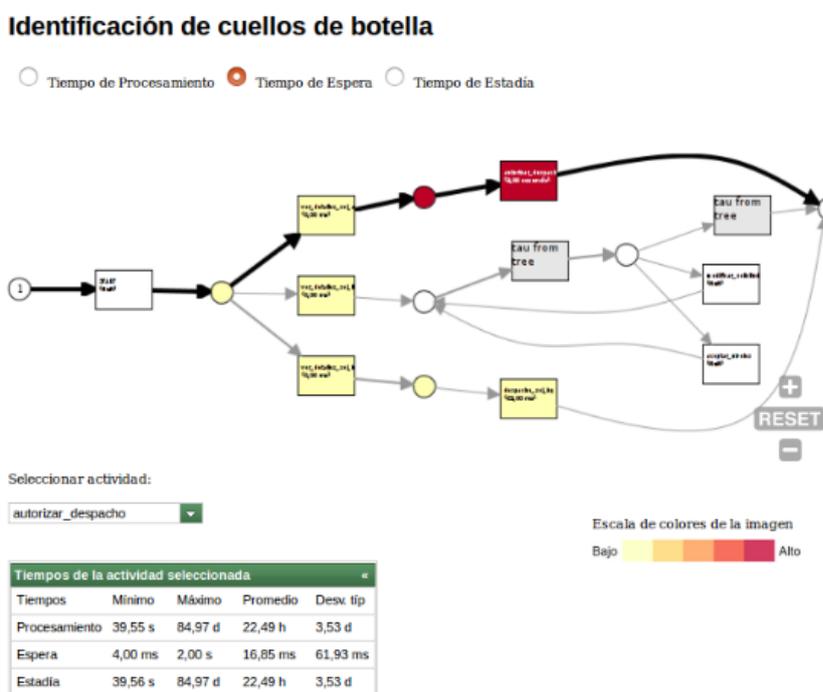


Figura 1. Vista de análisis incorporada al sistema XAVIA HIS. Fuente: elaboración propia

## Conclusiones

La minería de procesos es una tecnología novedosa, ampliamente utilizada para realizar análisis de procesos de negocio y se evidencia en la literatura un auge hacia el entorno hospitalario. Los cuellos de botella traen consigo un conjunto de dificultades e insuficiencias que provocan variabilidad en la ejecución de un proceso e irregularidades en el flujo de atención a pacientes. Entre las técnicas que permiten la identificación de cuellos de botella, “Replay P/C” es la más intuitiva, además de mostrar resultados sobre el rendimiento de los procesos que resultan de interés para la investigación. Las métricas de la técnica propuesta permitieron usar sus valores por defecto en aras de obtener resultados confiables y proporcionar menor complejidad de uso a los usuarios. Fue posible disminuir la complejidad del proceso de minería para obtener resultados con la técnica “Replay P/C”, extendiendo y facilitando su uso a

especialistas del dominio hospitalario.

## Referencias

- ADRIANSYAH, A., (2012). Replay a Log on Petri Net for Performance/ Conformance Plug-in. Technische Universiteit Eindhoven.
- ADRIANSYAH, ARYA, VAN DER AALST, WILL Y VAN DONGEN, B. (2012). Replaying history on process models for conformance checking and performance analysis. s.l.: WIREs Data Mining and Knowledge Discovery, 2012.
- CLAES, J., & POELS, G. (2013). Process mining and the ProM framework: an exploratory survey. In Business Process Management Workshops. Springer Berlin Heidelberg, p. 187-198.
- ORELLANA, A., PÉREZ, Y.E. & LARREA, O.U., (2015). Process Mining in Healthcare: Analysis and Modeling of Processes in the Emergency Area. Latin American Transactions, IEEE, Vol. 13 no. 5, p.1612-1618.
- ORELLANA, A., PÉREZ, Y.E., & LARREA, O.U, (2015a). Diseño de un componente para la identificación de eventualidades aplicando la técnica minero inductivo a los del Sistema de Información Hospitalaria del CESIM. 10ma Peña Tecnológica Nacional. ISBN: 978-959-286-030-8.
- ORELLANA, A., PÉREZ, D. & LARREA, O.U. (2015b). Analysis of Hospital Processes with Process Mining Techniques. In MEDINFO 2015: EHealth-enabled Health: Proceedings of the 15th World Congress on Health and Biomedical Informatics (Vol. 216, p. 310).
- ORELLANA, A., LARREA O. U., & PÉREZ-ALFONSO, D. (2015). "Generador de Registros de Eventos para el análisis de procesos en el Sistema de Información Hospitalaria XAVIA HIS." Convención Salud 2015.
- ORELLANA, A., PEREIRAS, K. & SOSA, A., (2015). Diseño de un componente para la detección de ruido en los procesos del Sistema de Información Hospitalaria del CESIM personalizando la técnica minero difusa. 10ma Peña Tecnológica Nacional. ISBN: 978-959-286-030-8.
- ORELLANA A., CASTAÑEDA, L. & VALLADARES, A., (2015). Diseño de un componente desde la perspectiva temporal para el análisis de los procesos del Sistema de Información Hospitalaria del CESIM. 10ma Peña Tecnológica Nacional. ISBN: 978-959-286-030-8.
- ORELLANA, A., LARREA, O.U, PÉREZ, Y.E., & PÉREZ, D. (2016). Inductive Visual Miner Plugin Customization for the Detection of Eventualities in the Processes of a Hospital Information System. Latin America Transactions, IEEE, Vol. 14 no. 4, pp.1930-1936.
- VAN DER AALST, W. M., VAN DONGEN, B. F., GÜNTHER, C. W., ROZINAT, A., VERBEEK, E., &

WEIJTERS, T. (2009). ProM: The Process Mining Toolkit. BPM (Demos), Vol. 489, p. 31.

- VAN DER AALST, W M. P. (2011). Process mining: discovery, conformance and enhancement of business processes, Springer Science & Business Media.

- VAN DER AALST, W, ADRIANSYAH, A., MEDEIROS, A.K.A., ET AL. (2011). Manifiesto sobre Minería de Procesos. s.l.: IEEE.