

IMPACTO GENERADO TRAS LA IMPLEMENTACIÓN DE CLOUD COMPUTING PARA OPTIMIZAR LOS RECURSOS
SUBUTILIZADOS EN STF GROUP S.A.

Jorge Mario Serrate

Nota de Aceptación

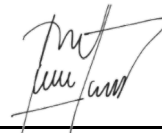
Certificamos que el presente Trabajo de Grado
Satisface, en alcances y calidad, todos los requisitos
Que demanda un Trabajo de Grado de Maestría.



Gerardo Sarria
Director

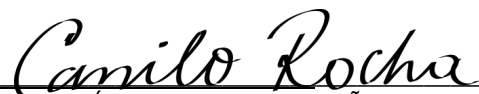


Frank Valencia
Jurado

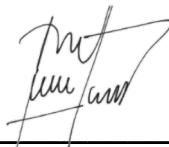


Juan Carlos Martínez
Jurado

Aprobado en cumplimiento de los requisitos exigidos por la
Pontificia Universidad Javeriana Cali, para optar el título de
Magister en Ingeniería énfasis Sistemas y Computación.



HERNÁN CAMILO ROCHA NIÑO Ph. D.
Decano Facultad de Ingeniería y Ciencias



JUAN CARLOS MARTÍNEZ ARIAS
Director Posgrados de Ingeniería y Ciencias

Santiago de Cali, 10 de Marzo del 2021

Acta de Correcciones al Documento de Trabajo de Grado

Santiago de Cali, Marzo 10 del 2021

Autor: Jorge Mario Serrate

Título del Trabajo de Grado: “IMPACTO GENERADO TRAS LA IMPLEMENTACIÓN DE CLOUD COMPUTING PARA OPTIMIZAR LOS RECURSOS SUBUTILIZADOS EN STF GROUP S.A.”

Director: Gerardo Sarria

Como indica el artículo 2.13 de las Directrices para Trabajo de Grado de Maestría, he verificado que el estudiante indicado arriba ha implementado todas las correcciones que los Jurados del Proyecto de Trabajo de Grado definieron que se efectuaran, como consta en el Acta de Evaluación correspondiente.



Firma del Director del Trabajo de Grado



FICHA RESUMEN

TRABAJO DE GRADO DE MAESTRÍA

TÍTULO: IMPACTO GENERADO TRAS LA IMPLEMENTACIÓN DE CLOUD COMPUTING PARA OPTIMIZAR LOS RECURSOS SUBUTILIZADOS EN STF GROUP S.A.

1. AREA DE TRABAJO: Programación concurrente y distribuida
2. TIPO DE PROYECTO: Aplicado, Investigación, medición y experimentación.
3. ESTUDIANTE: Jorge Mario Serrate Valencia.
4. CORREO ELECTRÓNICO: jotamarios@gmail.com
5. DIRECCIÓN Y TELÉFONO: Carrera 113 # 28-71 Apto 218E. Celular: 3013290277
6. DIRECTOR: Gerardo Sarria
7. VINCULACIÓN DEL DIRECTOR: Profesor de Planta
8. CORREO ELECTRÓNICO DEL PROFESOR: gsarria@javerianacali.edu.co
9. CO-DIRECTOR (Si aplica): No aplica.
10. GRUPO O EMPRESA QUE LO AVALA (Si aplica): STF Group S.A.
11. OTROS GRUPOS O EMPRESAS: No aplica.
12. PALABRAS CLAVE: Cloud Computing, Computación paralela y distribuida.
13. FECHA DE INICIO: Septiembre del 2019.
14. DURACIÓN ESTIMADA: 6 Meses.
15. RESUMEN:

Aunque el cloud Computing es un tema que ha sido abordado y explorado desde hace varios años, en Colombia actualmente no se le ha explotado como debería, a pesar de la disminución de costos que le pudiera significar a empresas, gobierno y emprendimientos. Esta tesis tiene como objetivo mostrar de manera práctica como una empresa puede aprovechar los usos de esta tecnología para mejorar condiciones de performance, y cómo estos impactan de manera directa o indirecta en los costos de implementación de soluciones de software, tiempos de entrega y fluidez de procesos cooperativos, que a la postre, redundará en beneficios corporativos para la competitividad, aumento de productividad y demás indicadores que generan mayor beneficio a las empresas.

Se debe resaltar la alta importancia de la adopción de las tecnologías en las empresas para apalancar la evolución y competitividad del país ya que solo de esta manera habremos de asegurar un resurgimiento económico que garanticen sostenibilidad y desarrollo, teniendo en consideración que la globalización y los tratados internacionales exigen competitividad en todos los ámbitos empresariales y sociales.

Resumen

El presente trabajo de grado aborda la problemática surgida tras el crecimiento acelerado de la empresa STF Group S.A. en términos de necesidad de cómputo y la consecuente insuficiencia de la infraestructura tecnológica. En su modelo de operación, las diferentes aplicaciones tienen demanda de recursos en diferentes frecuencias y rangos horarios, lo que le ha implicado a la empresa la adquisición de nueva infraestructura que supla estas demandas. Así entonces, se realiza un análisis de las diferentes aplicaciones con mayor impacto en términos de consumo de recursos y tiempo de respuesta al usuario final, para que, con una metodología de paralelización y distribución de las cargas de trabajo, se logre obtener mejores tiempos utilizando los recursos informáticos subutilizados, es decir, los PC de los diferentes usuarios de la empresa.

Este reto se aborda realizando una investigación sobre las diferentes plataformas de distribución de tareas que se han desarrollado, diseñando unos criterios de comparación, implementando el modelo que más se adecúe al ejercicio propuesto y, finalmente, determinando el impacto que genera la implementación en la empresa.

Los resultados obtenidos tras la aplicación de la distribución de las cargas de trabajo en paralelo usando distintos nodos en el sistema de nube de trabajo demuestra que para ciertos enfoques como el abordado aquí, resulta muy beneficioso en términos del esfuerzo y tiempo de respuesta con la consecuente satisfacción del usuario final y mejora en los flujos de procesos de la empresa ya que aplicaciones que hacen un uso constante y secuencial de los recursos, saturan su capacidad generando un tiempo de procesamiento alto y bloqueando otros procesos con necesidad del mismo recurso.

Se puede concluir entonces que el impacto generado en STF Group S.A., con esta investigación aplicada, deja un balance muy positivo para el proceso comercial y la aplicación en términos de tiempo de respuesta sin generar afectación negativa sobre los recursos subutilizados ampliando la capacidad de ver todas las partes como un todo que permitirá a futuro generar y procesar más bloques de trabajo de otras aplicaciones de la empresa.

Abstract

This master degree project addresses the problematic situation that arose after the accelerated growth of the STF Group S.A.'s company regarding the increased computing power need and the consequent technological infrastructure insufficiency . In its operating model, the demand for resources of each application (Software) is different in frequency and time ranges, such a situation has forced the company in acquiring new infrastructure to supply those demands. Consequently, an analysis of the different applications with the greatest impact in terms of resource consumption and response time to the end user is carried out, so that, with a methodology of parallelization and distribution of workloads, it is possible to obtain better times using underused computing resources, that is, the PCs of the different users of the company.

This challenge is addressed by researching the different task distribution platforms developed, designing comparison criteria, implementing the model that best suits the proposed exercise and, finally, determining the impact generated by the implementation in the company.

The results obtained after applying the distribution of workloads in parallel using different nodes in the working cloud system shows that for certain approaches such as the one discussed here, it is very beneficial for the effort and response time with the consequent end-user satisfaction and improvement in company's process flows since applications that make constant and sequential use of resources, saturate the capacity generating high processing time and blocking other processes wich need the same resource.

It can be concluded then that the impact generated in STF Group SA, with this applied research, leaves a very positive balance for the commercial process and the application's response time without generating a negative impact on the underutilized resources, expanding the capacity by overseeing all the parts as a whole, allowing to generate and process more work blocks from other company applications in the future.



Vigilada Mineducación



Res. 2333 del 2012

Impacto generado tras la implementación de Cloud Computing para optimizar los recursos subutilizados en STF Group S.A.

JORGE MARIO SERRATE VALENCIA
CÓDIGO 0068650

Proyecto de trabajo de grado para optar al título de Magister en
Ingeniería con énfasis en Ingeniería de Sistemas

Director

Gerardo Mauricio Sarria, PhD

FACULTAD DE INGENIERÍA Y CIENCIAS

MAESTRÍA EN INGENIERÍA - ÉNFASIS EN INGENIERÍA DE
SISTEMAS SANTIAGO DE CALI, ENERO DE 2021

Tabla de contenido

INTRODUCCIÓN.....	7
1. DEFINICIÓN DEL PROYECTO.....	8
1.1. DEFINICIÓN DEL PROBLEMA	8
1.1.1. PLANTEAMIENTO DEL PROBLEMA	8
1.1.2. FORMULACIÓN DEL PROBLEMA.....	9
1.2. Objetivos del proyecto	10
1.2.1. Objetivo General	10
1.2.2. Objetivos específicos.....	10
1.2.3. Resultados esperados	11
1.3. ALCANCE.....	12
1.4. JUSTIFICACIÓN	13
1.5. MARCO TEÓRICO DE LA REFERENCIA.....	14
1.5.1. BASES TEÓRICAS DE LA REFERENCIA.....	14
• Computación paralela y distribuida	14
• Cloud Computing.....	16
• Adopción de las tecnologías en Colombia y Latinoamérica.....	16
• Penetración de las tecnologías en las empresas Colombianas.....	18
• Herramientas para programación paralela y distribuida.	19
1.6. TRABAJOS RELACIONADOS.....	19
2. Desarrollo de la solución	21
2.1. Selección de aplicación a intervenir	21
2.1.1. Criterios de selección.....	21
2.2. Selección de plataforma de computación paralela y distribuida	27
2.2.1. Investigación	27
2.2.2. Criterios de selección de plataforma de paralelismo	29
2.2.3. Métricas de criterios.....	30
2.2.4. Análisis de la selección de criterios	31
2.2.5. Importancia de los criterios para el negocio.....	34
2.2.6. Criterios de evaluación de impacto de la implementación	35
2.3. Desarrollo.....	36
2.3.1. Análisis.....	36
2.3.5. Diseño.....	41
3. Pruebas y resultados.....	48
3.1. Análisis	49
3.1.1. Comparativo	52



3.2. Calificación de impacto	53
3.3. Conclusiones	54
4. REFERENCIAS BIBLIOGRÁFICAS	56
Anexo 1. Tabla de aplicaciones STF Group S.A.	58

LISTA DE FIGURAS

Figura 1: Uso de tecnología en empresas.....	17
Figura 2: Penetración de las tecnologías avanzadas por estadio de la cadena de valor ..	18
Figura 3: Diagrama de Despliegue.....	39
Figura 4: Modelo entidad relación	41
Figura 5: Diagrama de secuencia.....	42
Figura 6. Diagrama de despliegue solución final.....	43
Figura 7. Flujo grama de solución antes de mejoras	44
Figura 8. Flujograma con paralelismo de procesos.	45
Figura 9. Detalle de ejecución en nodos.	45
Figura 10. Método principal con orden de secuencia.	46
Figura 11. Método de liquidación que ejecuta cada nodo.....	47
Figura 12. Tiempos antes y después de distribución por horarios.....	49
Figura 13: Tiempos promedio por franja horaria.....	50
Figura 14: Tendencia de tiempos por número de nodos.	51
Figura 15. Linares Moreno, C. 2014. Tiempo de procesamiento de la migración sísmica [17].....	52
Figura 16. Jing Xiong, Jianliang Wang, Jianliang Xu. 2010. Velocidad relativa [16].	53

LISTA DE TABLAS

Tabla 1: Matriz de criterios de selección de aplicación a intervenir.	23
Tabla 2: Top 5 aplicaciones para selección.....	24
Tabla 3: Calificación de aplicaciones	26
Tabla 4: Criterios de evaluación de plataforma de programación paralela y distribuída. ..	30
Tabla 5: Calificación de plataformas	33
Tabla 6: Tiempos de respuesta por horarios y número de nodos.....	48

INTRODUCCIÓN

El Cloud Computing es una tecnología que ya ha sido abordada desde los años 60, y aunque actualmente el término está acuñado y es más utilizado en la tercerización de las plataformas de infraestructura; por definición, una nube privada empresarial también clasifica y tiene un gran potencial para la explotación de los recursos de los que dispone.

Las empresas usualmente disponen de un gran conjunto de dispositivos de cómputo en toda la organización con diferentes propósitos como lo son las computadoras personales, servidores de aplicaciones, bases de datos, dispositivos de red y hasta dispositivos móviles. Una importante participación en el presupuesto de las compañías está enfocado a adquirir, administrar, soportar y dar mantenimiento a la infraestructura que soporta las aplicaciones de la empresa, de tal manera que se garantice la estabilidad de las plataformas que soportan la operación de manera automatizada y conectada. En empresas de un tamaño mediano y grande, la operación automatizada prácticamente es obligatoria pues la gran cantidad de datos y procesos harían imposible considerar un panorama en el cual se opere manualmente de manera eficiente. Así las cosas, gran parte de los costos operativos se han de considerar para proyección de crecimiento, soporte a nuevos procesos y verticales de negocio.

Para dar solución a la creciente demanda de las compañías por el consumo de tecnología tanto en software como hardware, surgen dos alternativas que resultan ser las más comunes en Colombia, según la encuesta de transformación digital del 2017 desarrollada por Andi (Asociación Nacional de Empresarios de Colombia) [1]; implementar un centro de cómputo propio con una infraestructura que soporte la operación de las diferentes plataformas de software, teniendo en consideración los costos generados por el mantenimiento, el soporte, el desgaste del hardware que produce una obsolescencia y por lo tanto desvalorización. Por otro lado está la alternativa de adquirir plataforma bajo la modalidad de hosting por virtualización o plataforma compartida. No obstante que las compañías van aumentando su capacidad en la medida que ciertas aplicaciones ven disminuido su potencial por falta de recursos, otros equipos informáticos podrían no estar siendo utilizados a su totalidad. Por éste motivo, ésta tesis plantea una solución al desequilibrio que surge en las empresas, implementando sus aplicaciones en bloques de datos que en paralelo y de manera distribuida logren utilizar los recursos de la compañía de manera balanceada y logrando así una optimización de sus recursos.

1. DEFINICIÓN DEL PROYECTO

1.1. DEFINICIÓN DEL PROBLEMA

1.1.1. PLANTEAMIENTO DEL PROBLEMA

Actualmente las empresas invierten su capital de tecnología en adquisición de equipos, compra y construcción de software, pero pocas en Colombia hacen una adecuada participación en I&D que pudiera cerrar las brechas que existen con países del denominado primer mundo [1][2]. Los altos costos de implementar tecnología que se ajuste a la actual demanda de procesos en línea hacen inviable pensar en estas arquitecturas que permitirían dar una respuesta oportuna al cliente. De no participar activamente en la creación de soluciones de implementación abordando tecnologías de punta, el país no podrá lograr el avance en términos de competitividad y se verá relegado a utilizar propuestas de empresas extranjeras en las que los estudios avanzados (maestría y doctorado) marcan una diferencia importante en la participación poblacional [3].

Cloud Computing permite hacer uso de la computación paralela y distribuida utilizando los recursos subutilizados para mejorar notablemente los tiempos de respuesta de un sistema. Su uso permitirá mejorar procesos de compañía que al depender del análisis de grandes bloques de información, deben supeditar los pasos del proceso a horas o días y terminar un flujo en un tiempo poco conveniente para promover la competitividad.

En los planes de crecimiento académico tenemos la obligación moral de generar conocimiento que impacte de manera positiva en el crecimiento del país, que nos permita avanzar en la utilización de tecnologías, y mejor aún, proponer otras nuevas o no exploradas.

STF Group S.A es una empresa del sector retail fashion que se dedica al diseño, fabricación y comercialización de productos de moda en Latinoamérica, habiendo nacido y conservando su principal sede en Cali, Colombia. Considerando los beneficios que pudieran teóricamente generar a una empresa como STF Group, un gran dilema a resolver consiste en determinar cuál es la mejor manera de aplicar éstas tecnologías en el contexto específico de la compañía dada la arquitectura de las aplicaciones, y posterior a esto, resulta indispensable plantearse cuáles serán aquellas aplicaciones que serán objeto de estudio para determinar si se cumple la premisa de otorgar un beneficio a la compañía; así entonces habría que determinar unos criterios se utilizarán para la escogencia de los procesos a

abarcar y la selección de las métricas tal que permitan al final del ejercicio, medir qué tanto impacto genera la introducción del Cloud Computing.

Así las cosas, el problema que se plantea es la manera en que se determinará si la implementación de Cloud Computing en una nube privada para STF Group genera un impacto positivo para la organización utilizando unos criterios que den luces de lo que esto implica para las demás empresas que decidan hacer frente a la competencia apalancándose de tecnología de gran envergadura.

1.1.2. FORMULACIÓN DEL PROBLEMA

El problema se divide en varias preguntas importantes: ¿Qué impacto genera la implementación de Cloud Computing en STF Group con respecto a la manera que viene operando? Para lograr determinar éste punto, resulta necesario plantearse ¿Cómo medir el impacto tras la aplicación de Cloud Computing? Y adicionalmente será necesario determinar para tener un alcance acotado a éste proyecto ¿cuáles y cuantas aplicaciones deberían ser intervenidas para obtener un resultado que permita medir un impacto notable? Lograr determinar las aplicaciones a intervenir requerirá que podamos responder ¿Qué criterios de selección debería utilizar para determinar de manera que el ejercicio sea medible e imparcial?

1.2. Objetivos del proyecto

1.2.1. Objetivo General

Implementar computación concurrente y distribuida en STF Group S.A que genere un impacto positivo en el rendimiento de procesos de la compañía.

1.2.2. Objetivos específicos

- Determinar los procesos más críticos de la compañía donde su tiempo de ejecución puedan resultar cruciales para mejorar los procesos internos.
- Escoger unos criterios de selección que de manera asertiva permitan la correcta asignación de esfuerzos en términos de la implementación, para impactar positivamente los procesos de la compañía.
- Analizar las diferentes plataformas que habilitan el Cloud Computing para la selección más adecuada para el ejercicio de ésta tesis.
- Implementar algoritmos que habiliten el Cloud Computing sobre los sistemas escogidos para mejorar el tiempo de respuesta.
- Seleccionar métricas que permitan medir adecuadamente el impacto que genere en los procesos seleccionados.
- Implementar cambios en la aplicación seleccionada por su impacto en los tiempos de respuesta, tal que se utilice computación paralela y distribuida mediante la herramienta seleccionada para tal fin.
- Analizar los factores de éxito que la implementación aplica a la compañía.

1.2.3. Resultados esperados

Acorde a los objetivos específicos de esta tesis, se espera obtener como resultado de la implementación:

1. Una matriz de priorización con criterios medibles e imparciales para la selección de los procesos y aplicaciones que pudieran mejorar la productividad de la compañía.
2. Una lista priorizada de los procesos y aplicaciones implicadas que podrían ser intervenidas para mejorar su performance.
3. Un conjunto de métricas que permitan determinar el impacto causado a la organización tras implementar las mejoras dadas por la distribución de los procesos en recursos sub utilizados.
4. Un documento de análisis de selección de la plataforma más idónea para la implementación de Cloud Computing en STF GROUP.
5. Una aplicación modificada para trabajar con una arquitectura de procesamiento distribuida en un Cloud conformado por hardware sin completa utilización en la compañía.
6. Un documento de análisis que corrobore el impacto generado al implementar el procesamiento en Cloud utilizando las métricas seleccionadas para tal efecto.



1.3. ALCANCE

El presente proyecto consistirá en la implementación de la modificación de los códigos fuentes actuales de una aplicación que sea seleccionada mediante un conjunto de criterios tal que impliquen un máximo beneficio expresado por las métricas que le califican. Así pues, tras seleccionar éste aplicativo y analizar su posible impacto, se le aplicarán ajustes al código fuente para que permita que sus procesos más críticos puedan ser distribuidos en los computadores de la empresa que por la demanda en un determinado momento pudiera no estar haciendo uso de su total capacidad de cómputo.

Tanto para la distribución de los procesos, orquestamiento de los paquetes procesados como para el análisis de la capacidad disponible se utilizará una plataforma de las ya disponibles de código abierto para la implementación de computación distribuida, de tal manera que en este proyecto no se abordará la creación de estas características necesarias éste cometido, dado que no resulta necesario.

La selección de la herramienta se hará basado en un análisis con juicio de experto y considerando el entorno de la empresa objeto del estudio y la facilidad que la herramienta otorgue para la implementación misma.

1.4. JUSTIFICACIÓN

Este proyecto tiene varias aristas que justifican su implementación desde diferentes perspectivas; en primera instancia, la relevancia para el país en términos de la evolución de las empresas que deben adaptarse a la alta competencia que surge como resultado de los fenómenos de la universalidad, la falta de implementación de tecnologías en el país [2] dada la poca especialización y baja academia en los empleados [3] de las diferentes empresas; la pertinencia de dar a conocer el impacto de estas investigaciones e implementaciones, que podrían dar origen a la aplicación de temas abordados durante la maestría por su relevancia teórica y práctica.

Las empresas actualmente trabajan de una manera tradicional adquiriendo sus propios cuartos de cómputo y adecuándolo con la infraestructura necesaria para operar; nuevos modelos de solución para el alojamiento de las aplicaciones corporativas y la data como Cloud Computing en el cual se plantean negociaciones como PaaS (Platform As A Service o Plataforma como servicio), están dando origen a una migración a la nube para almacenar y procesar la información. Pero, esto se está haciendo tal y como si simplemente se desplegara en un nuevo servidor [4], sin balancear las cargas en diferentes bloques de información procesada de manera distribuida para permitir un paralelismo que de origen a un performance mucho más eficiente; adicionalmente, el utilizar aquellos recursos que están subutilizados permitirá un impacto positivo en la disminución de gastos de las compañías y ayudarán a tener un margen mayor para maniobrar en las estrategias de asignación de precios siendo más competitivos.

Desde el punto de vista académico, resulta congruente con el contenido programático de la maestría y pertinente la aplicación en las empresas Colombianas. Éste tema que a pesar de tener varios años siendo implementado en diferentes tipos de proyectos, resulta tener una relevancia tan importante que no se considera como un tema dentro de una materia sino una materia como tal. Su aplicación a podido ayudar a dar solución a grandes retos de la humanidad como las investigaciones astrológicas y de biotecnología; y resulta interesante que éstas implementaciones no se queden en proyectos de las más altas empresas e investigaciones, sino que por el contrario sean práctica constante de las empresas de nuestro país.

Se debe tener en consideración los esfuerzos del ministerio de educación [3] y el presupuesto nacional de invertir en educación y tecnología para impactar positivamente las empresas y por ende la evolución del país en términos de competitividad, y ya que no es un

tema que esté en pleno y común uso de las empresas, se puede considerar que se encuentra en estado del arte.

1.5. MARCO TEÓRICO DE LA REFERENCIA

1.5.1. BASES TEÓRICAS DE LA REFERENCIA

Como parte importante del desarrollo de ésta tesis, se ha realizado una investigación de las diferentes bases teóricas que envuelven los temas a tratar durante éste proyecto y que resultan relevantes para tener un adecuado entendimiento y una correcta determinación para resolver cada una de las preguntas planteadas en la problemática del proyecto.

- **Computación paralela y distribuida.**

La programación paralela y distribuida hace referencia a la metodología de programación que considera el cómputo de datos particionando el total de la información a procesar de tal manera que diferentes máquinas puedan de manera colaborativa distribuir dichas tareas de procesamiento y al final obtener mejores resultados en términos del tiempo de ejecución bajo el pensamiento de “Divide y vencerás” en el cual se concibe que un problema puede ser resuelto más fácilmente y eficientemente particionándolo en subproblemas. Así pues, estos subproblemas serán resueltos paralelamente por las máquinas colaboradoras y serán retornadas a un sistema central con el rol de orquestador que deberá consolidar los resultados para obtener un único y final resultado.

Un programa puede ser paralelo sin que necesariamente sea distribuido, esto es, trabajar con multihilo donde la memoria es compartida y se utilizan los diferentes cores de la máquina para procesar paralelamente bloques de

información. Por el contrario, en la programación distribuida, las tareas de programación se distribuyen entre máquinas distribuidas que bien podrían estar en una red LAN o podrían estar a grandes distancias en una red WAN.

Se deben considerar factores importantes para la determinación de la aplicación de la computación paralela como la dependencia que existe en el cómputo de los bloques de información, pues, si la computación exige una secuencia de dependencia entre un elemento y otro, la computación paralela no podría llevarse a cabo. Adicionalmente se debe tener en cuenta la clasificación del paralelismo como de grano fino, cuando las subtareas deben comunicarse muchas veces por segundo; de grano grueso cuando la comunicación entre las subtareas no se debe realizar muchas veces por segundo y una tercera clasificación denominada paralelismo vergonzoso cuando las subtareas no se comunican nunca o casi nunca, lo cual implicaría una mejor condición para implementar el paralelismo puesto que cada paquete se computaría independientemente y regresaría al orquestador solo cuando la computación haya terminado.

A pesar de que los paradigmas de programación paralela y distribuida no son nuevos, “La mayoría de los desarrolladores continúan diseñando soluciones secuenciales por lo que no aprovechan en toda su dimensión las capacidades del hardware de las configuraciones disponibles en la actualidad” [5].

- **Clustering**

Los clusters constituyen un conjunto de computadoras unidas por red que habilitan la programación distribuida. Un cluster por definición tiene como base de comunicación una red de alta velocidad, por lo cual, se considerarían aquellos equipos de cómputo que se encuentran en una red LAN. Sin embargo, hay que destacar que las actuales tecnologías de comunicación como la fibra óptica han llevado a tener redes WAN de muy altas velocidades, lo cual podría habilitar un cluster con comunicaciones entre grandes distancias.

Los beneficios del Clustering incluyen la mejora en rendimiento, la disminución de los costos, baja latencia, la distribución de cargas y la recuperación de las capacidades subutilizadas de los equipos de cómputo, como puede constatarse en [6].

- **Cloud Computing.**

El cloud computing o computación en la nube, refiere al paradigma de ofrecer servicios en servidores remotos principalmente en internet, que facultarían el procesamiento de información o la distribución de servicios en servidores que no necesariamente son propios. La tendencia del Cloud Computing ha venido en crecimiento en los últimos años, y ha implicado para muchas compañías la migración de servicios a internet con el propósito de disminuir costos y delegar las tareas de administración de la infraestructura a empresas especializadas que prestan sus servicios de Hosting de manera compartida o privada según se acuerde. Si bien éste concepto puede vincularse con el Clustering, dista considerablemente su significado; un Clustering bien puede concebirse dentro de servidores que estén a disposición en Cloud Computing de tal manera que se complementan, pero también es factible implementar un Cluster en equipos de cómputo propio [2].

Se deben recalcar algunos atributos interesantes del Cloud Computing como la capacidad de elasticidad y escalabilidad, lo que le da la habilidad de adaptar los recursos a la medida de la necesidad por la demanda. Los costos pueden ser más bajos que la implementación de infraestructura propia dado que estos servicios de cómputo están distribuidos en sus diferentes clientes y el aprovechamiento de la capacidad estará proporcionada en la medida que uno u otro tengan una mayor demanda. El mantenimiento de los equipos es otra gran ventaja para las empresas dado que corre por cuenta del proveedor del servicio, y éste mantenimiento puede a veces ser proveído también a las aplicaciones si se adquiere como SaaS (Software as a Service) en el cual se opta por pagar por el software como un servicio y no solo la infraestructura.

- **Adopción de las tecnologías en Colombia y Latinoamérica**

Recientes estudios del Ministerio de tecnología de Colombia [1] han revelado que en términos de la adopción de tecnologías reconocidas como avanzadas, el país viene teniendo un incremento en los últimos años; sin embargo, éste crecimiento resulta incipiente si se le compara con la adopción que ya logran

países pertenecientes a la OCDE (Organización para la cooperación y el desarrollo económico) [7].

“El informe muestra que las tecnologías avanzadas con más baja penetración a nivel nacional son realidad virtual (1%), robótica (1,5%) y también el Blockchain (1,6%), siendo las mipymes las de menor contribución.”¹¹

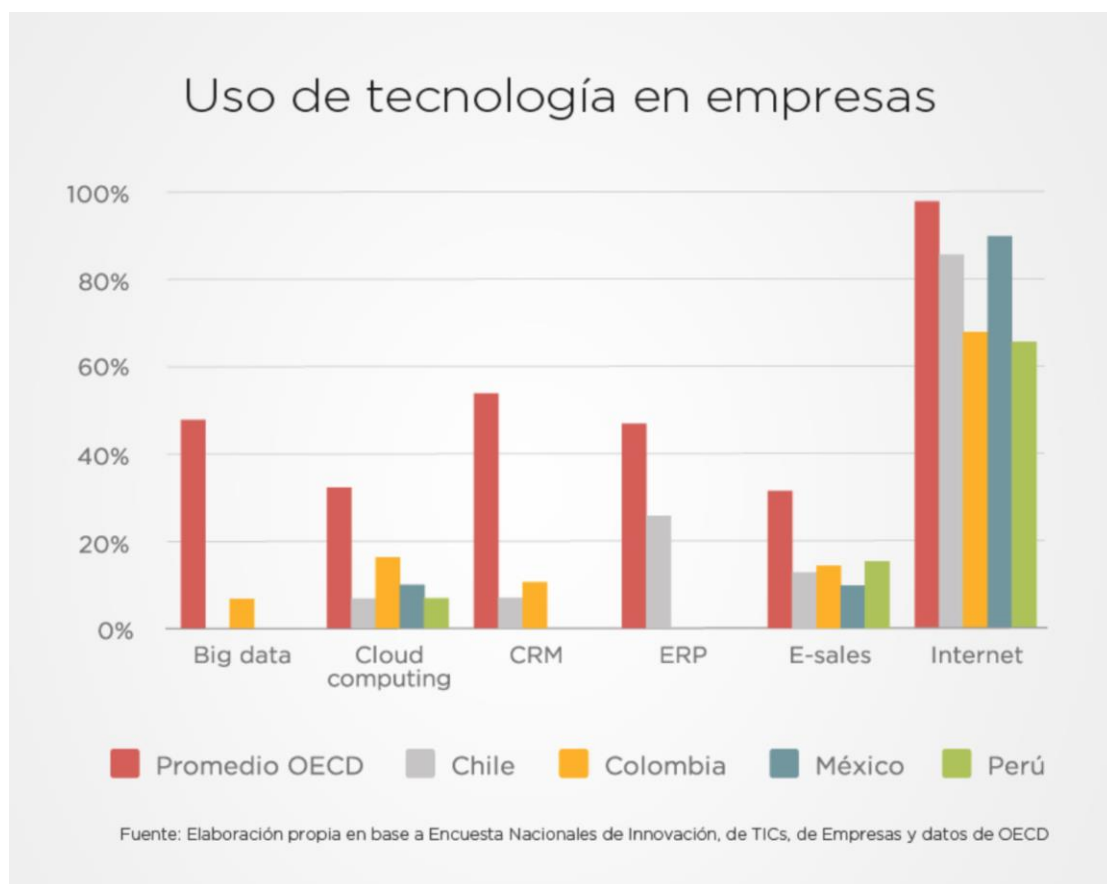


Figura 1: Uso de tecnología en empresas

COLOMBIA: PENETRACION DE TECNOLOGIAS
AVANZADAS POR ESTADIO DE LA CADENA DE
VALOR (2017)

Pilar	Tecnología	Nacional	Grandes	Pymes	Micros
Infraestructura	Ciberseguridad	32.2 %	67.2 %	37.1 %	25.4 %
	Computacion en la nube	19.1 %	48.8 %	22.9 %	13.6 %
Procesamiento	Internet de las cosas	9.0 %	14.8 %	9.3 %	8.2 %
	Robotica	1.5 %	11.1 %	1.2 %	0.6 %
	Impresión 3D	2.2 %	4.8 %	2.1 %	2.1 %
	Realidad virtual	1.0 %	1.7 %	0.9 %	1.0 %
Distribucion	Big data	3.2 %	16.8 %	4.0 %	1.3 %
	Inteligencia artificial	1.8 %	9.7 %	2.4 %	0.7 %
	Blockchain	1.6 %	5.9 %	1.6 %	1.1 %

Fuente: Encuesta MINTIC/CDC, 2017

Figura 2: Penetración de las tecnologías avanzadas por estadio de la cadena de valor

- Penetración de las tecnologías en las empresas Colombianas.

La penetración de las nuevas tecnologías en las compañías mantiene un sesgo según el tamaño de la empresa como habría de esperarse, pero aún somos un país que en términos de competitividad lleva un paso atrás si de automatización, virtualización y demás aspectos tecnológicos lo comparamos (ver Figura 2). Según MINTIC, solo el 19.1% de las empresas a nivel nacional han adoptado la computación en la nube para sus servicios tecnológicos, siendo ésta una tecnología que lleva ya varios años en el mercado. Por su parte, tecnologías más vanguardistas como la inteligencia artificial, tienen una participación del 1.8% a nivel nacional. El caso no es ajeno a STF Group S.A, quien a la fecha viene realizando un proceso de transformación digital, pero que en términos de adopción mundial, representa un atraso en tiempo que le significa una pérdida de oportunidad (ecommerce), y desventaja con la competencia (WMS, Cloud Computing, CRM).

- **Herramientas para programación paralela y distribuida.**

Múltiples herramientas para trabajar programación paralela y distribuida han sido implementadas al día de hoy. Boinc es una de las herramientas referencias a día de hoy, siendo implementada para el proyecto SETI por la universidad de California en Bakerley, y que resulta una herramienta de referencia por el impacto del proyecto por el cual surgió, pero, esto dio origen a otras herramientas como Rocks, creada en 2000 por la National Science Foundation y utilizada ampliamente por científicos con la necesidad de investigar problemas de gran envergadura con soluciones enfocadas en particionar en sub problemas que pudieran tener un tiempo menor de solución [6][8][9][10].

Charity Engine es otro de los grandes ejemplos, basados en Boinc, de herramientas que apuntan a minimizar el esfuerzo de quien investiga problemas que pueden ser resueltos de manera distribuida para lograr tiempos prudentes para las necesidades del mercado, y recalando fuertemente el concepto de “Computación voluntaria”, para lo cual, los computadores personales toman un papel importante aportando su computación “en reposo”, o “sobrante”, dada la sub utilización de él o los procesadores.

1.6. TRABAJOS RELACIONADOS

Consultando en los diferentes estudios formales con publicaciones de revistas, publicaciones académicas y otros, podemos encontrar varios ejemplos de uso, pero por lo general enfocados en pruebas académicas. Un ejemplo de éstos lo encontramos en [11] en el cual se ha planteado un uso de un Cluster de computadores para correr el algoritmo para encontrar las matrices de Hadamard. Los resultados son bastante dicientes, logrando obtener una mejora en performance por encima del 67% en términos del tiempo, lo cual constituye un ahorro importante en la ejecución, y por demás en costos puesto que al utilizar los tiempos de desuso del procesador de las diferentes máquinas participantes, se optimiza el uso de los recursos.

En estos ejemplos de uso podemos ver que existe un número importante de herramientas dispuestas para orquestar la distribución el procesamiento y consolidar los resultados, lo cual resulta bastante importante para no reinventar la rueda cada vez que se pretende realizar Clustering para cada aplicación.

Un impacto importante tiene también el método de asignación de las tareas y distribución de las mismas según [8]; trabajo en el cual se propone un método de planificación de la asignación de recursos y tareas basado en la determinación dinámica de las tareas susceptibles de ser asignadas y la pertinencia de los procesadores disponibles para ser asignados. Múltiples factores deben tenerse en consideración en éstas dos tareas que resultan determinantes en los resultados esperados; sin embargo, para éste proyecto, se dejarán éstos aspectos a la herramienta distribución y procesamiento paralelo, de tal manera que su cola de tareas se encargará de distribuirlo como lo haya determinado el equipo de investigación de dicha herramienta.

Trabajos como [8] y [12], muestran la importancia de la implementación de metodologías de segmentación de las tareas en las aplicaciones con el objetivo de poder distribuir la carga operativa de las transacciones en diferentes tareas que pudieran ser resueltas por nodos dentro del conjunto de clientes. Gran parte del proyecto estará enfocado a la manera en que deba separarse la actual forma de cálculo y/o procesamiento de información de tal manera que puedan considerarse entidades de cálculo separadas para darle una mejor perspectiva de segmentación del problema, y así entonces, lograr dividir y unificar resultados.

Otros trabajos como vemos en [13], utilizan la plataforma de Cloud Computing de Amazon para el despliegue de varios nodos que permiten trabajo paralelo y distribuido utilizando Rocks como herramienta de distribución y computación de trabajo; a ésta, le agregan un componente llamado Condor que permite hacer un rápido despliegue de nuevos nodos. Esto cobra importancia cuando se quiere realizar un cluster que tenga la flexibilidad de agregar nodos a demanda, y que tiene como implicación, toda la configuración de la infraestructura de manera manual si es que no se cuenta con éste tipo de herramientas de automatización.

2. Desarrollo de la solución

2.1. Selección de aplicación a intervenir

La selección de la aplicación que se adaptará a la computación en grid colaborativa es un componente muy importante para el presente proyecto, pues a partir de ésta, podrá determinarse el impacto que se genera a la aplicación misma y a la organización para la cual se desarrollan los procesos. STF Group tiene un conjunto de aplicaciones con las que se operan diferentes procesos de la compañía, muchos de alta criticidad y otros de soporte a las operaciones de algunos puntos específicos de los procesos.

Dada la naturaleza del presente proyecto en términos de tiempo y disponibilidad de equipo, se pretende tomar una sola aplicación tal que pueda demostrarse que el uso de estas metodologías de implementación le permitirán a las empresas maximizar el uso de sus componentes de hardware y software a un menor costo y con mejores resultados.

Los criterios que se han de considerar para la evaluación de la selección de mejor oportunidad entre las aplicaciones existentes, deberán entonces tener en cuenta el marco estratégico de la compañía, para así, apuntar los resultados a un conjunto de capacidades objetivo de la empresa y no solo a mejorar el performance de cualquier aplicación.

2.1.1. Criterios de selección

- Tiempo de respuesta
- Criticidad para el negocio
- Ejecuciones por mes
- Número de sub procesos

2.1.2. Objetividad y pertinencia

Tiempo de respuesta: Este criterio tiene una relevancia alta dada la premura de entrega de información de valor a la compañía y la importancia cuando un

proceso de cómputo tiene predecesores y precursores que generan una dependencia para culminar el total del proceso.

Criticidad para el negocio: La importancia que tenga la aplicación para el flujo de un proceso específico, siendo que este representa una dependencia para otros pasos del proceso y que puedan implicar una disminución en ventas, retraso en el ciclo productivo o dato relevante para la toma de decisiones.

Ejecuciones por mes: Es importante determinar la frecuencia con que se generan ejecuciones de alto impacto pues al aumentar en número de ejecuciones debería otorgarse una mayor relevancia la repercusión total sobre las aplicaciones de la compañía.

Número de subprocesos: La importancia de este criterio se basa en la posibilidad de determinar la criticidad y complejidad de la solución de software a evaluar; al tener un número mayor de subprocesos, éstos podrían a su vez generar un impacto mayor en diferentes sistemas.

2.1.3. Filtro de aplicaciones

STF Group cuenta con un inventario de 62 aplicaciones (ver anexo) entre propias, de terceros desplegadas on premise, terceros en la nube administradas por STF Group y terceros que prestan un servicio pero no tienen una administración de la compañía.

Se aplicó un primer filtro a éstas aplicaciones para la determinación de la aplicación ideal para el ejercicio de éste proyecto en el cual se identificó si es susceptible de ajustar ya sea porque es un desarrollo propio de la compañía o porque el código fuente ha sido entregado a la empresa y su forma de licenciamiento le permite modificar dicho código. Al final de la aplicación de este filtro se cuenta con un total de 42 aplicaciones que pudieran ser ajustadas.

Como parte del método de trabajo de este documento, se seleccionan las siguientes 5 aplicaciones a juicio de experto para la evaluación cualitativa mediante los criterios seleccionados en la sección anterior para minimizar el ejercicio sobre aplicaciones que por experiencia se conoce del impacto que genera en los tiempos de respuesta y la criticidad para el negocio.

1. Middleware ecommerce
2. Binaps Comisiones.
3. Corners
4. Conciliaciones
5. Hoja de costos

Las aplicaciones listadas hoy en día representan las aplicaciones con más queja de parte de las áreas usuarias por la lentitud en realizar las operaciones para las cuales fueron desarrolladas. Por este motivo, se evaluará con los criterios seleccionados, cual tiene un mayor impacto para realizar un proceso de paralelización y distribución de carga de sus operaciones para procurar una mejora en su rendimiento utilizando los actuales recursos de la compañía.

2.1.4. Matriz de criterios

Tabla 1: Matriz de criterios de selección de aplicación a intervenir.

Criterio	Valores	Calificación	Totales
Tiempo de respuesta	Dif expectativa hasta el 10%	1	30%
	Dif expectativa hasta el 50%	2	
	Dif expectativa hasta el 80%	3	
	Dif expectativa hasta el Superior al 80%	5	
Criticidad para el negocio	Alta	5	40%
	Medio	3	
	Baja	1	
Ejecuciones por mes	Entre 1 y 30	1	15%
	Entre 30 y 90	3	

	90 en adelante	5	
Número de subprocesos	De 0 a 2	1	15%
	De 3 a 7	3	
	De 7 en adelante	5	

2.1.5. Análisis top 5

Tabla 2: Top 5 aplicaciones para selección.

	Operaciones	Operación más crítica	Tiempo de respuesta	Criticidad para el negocio	Ejecuciones por mes	Número de subprocesos
Middleware e-commerce	1. Productos 2. Cambios de precio 3. Actualización de inventario 4. Sincronización de órdenes 5. Envío de facturas. 6. Envío de guía de despacho. 7. Notificaciones SMS	Sincronización de órdenes	45 segundos	Alta	7600	5 -> Creación cliente, crear recibo de caja, Crear pedido, Generar guía de despacho, Notificar por SMS.
Binaps Comisiones	1. Cargue de presupuesto. 2. Distribución de	Liquidación de comisiones	7 horas	Alta	30	9. Consulta de metas, Consulta de tiendas, Consulta de



	presupuesto 3. Asignación de presupuesto. 4. Asignación de recursos 5. Liquidación de comisiones.					empleados, Consulta de facturas, Consulta de devoluciones, Consulta de cargos, Consulta de porcentajes de cumplimiento, Cálculo de comisión, Almacenamiento de comisión.
Corners	1. Cargue de facturas. 2. Cargue de movimientos 3. Distribución de bodegas 4. Facturación. 5. Cargue de maestros	Facturación	12 minutos	Media	1	3. Consulta de Stock, Consulta de valores de cargue, creación de documento.
Conciliaciones	1. Cargue de archivos bancarios. 2. Proceso de conciliación. 3. Generación de movimiento contable	Proceso de conciliación	45 segundos	Baja	4	4. Consulta de archivo, consulta de recibos, comparación, Informe final.
Hoja de costos	1. Parámetros por marcas 2. Generación de hoja de costos.	Generación de hoja de costos	120 segundos	Media	500	5. Consulta de materia prima, Consulta de procesos de



	3. Almacenar temporal 4. Cierre de hoja de costos.						producción, Consulta de parámetros de gastos, informe de costo, asignación de precios por margen.
--	---	--	--	--	--	--	--

2.1.6. Calificación final

Tabla 3: Calificación de aplicaciones

	Tiempo de respuesta	30 %	Criticidad para el negocio	40 %	Ejecuciones por mes	15 %	Número de subprocesos	15 %	TOTAL
Middleware eCommerce	2	0,6	5	2	5	0,8	3	0,5	3,8
Binaps Comisiones	5	1,5	5	2	3	0,5	5	0,8	4,7
Corners	2	0,6	3	1,2	1	0,2	1	0,2	2,1
Conciliaciones	1	0,3	1	0,4	1	0,2	3	0,5	1,3
Hoja de costos	5	1,5	3	1,2	5	0,8	3	0,5	3,9

2.1.7. Conclusión de selección

Por la gran diferencia en expectativa de los tiempos de respuesta, la importancia del negocio y el número de subprocesos que le componen, la aplicación Binaps Comisiones, bajo la calificación otorgada a los criterios de selección, califica para ser el sistema más representativo y de mayor impacto para la organización, lo cual implica que es un ejemplar ideal para realizar las adecuaciones de paralelismo y distribución de procesos de tal manera que la priorización nos arrojará un resultado de mayor interés.

2.2. Selección de plataforma de computación paralela y distribuida

2.2.1. Investigación

- **BOINC**

BOINC (Berkeley Open Infrastructure for Network Computing) es una plataforma para la distribución de procesamiento en diferentes dispositivos de manera colaborativa, haciendo uso de un porcentaje parametrizado de la capacidad de procesamiento de la máquina cliente. Originalmente, BOINC fue creado por un grupo de investigación de la universidad de Berkely para atender el proyecto SETI@Home pero posteriormente fue liberada para que fuera adoptada por múltiples proyectos científicos que requirieran una gran cantidad de procesamiento. La filosofía de esta plataforma es principalmente colaborativa, en el cual personas naturales con sus recursos personales (Computadores y dispositivos móviles) donan el tiempo de máquina subutilizado. La herramienta presenta muchos beneficios como la posibilidad de instalar en clientes Linux, Mac OS, Windows y una extensión para Android; el hecho de ser una plataforma libre y su capacidad de trabajar en background sin interrumpir el trabajo de las personas que aportan con los ciclos de máquina inoficiosos.

- **JPPF**

JPPF es una plataforma para la programación distribuida escrita en java y para el lenguaje de programación Java, Android y .Net. Utiliza una figura de cliente servidor en el cual los nodos representan clientes esperando peticiones de trabajo para realizar cómputo acorde a las instrucciones entregadas desde un “cliente”. La arquitectura contiene entonces dos aplicaciones principales del JPPF en la cual un llamado Driver recibe las peticiones de una aplicación “cliente” y distribuye el trabajo a los Nodos para finalmente consolidar los resultados y entregarlos al cliente.

Entre los beneficios que representa la utilización de JPPF están la facilidad de uso, pudiendo desplegar muy rápidamente un código de bloques de trabajo mediante plantillas entregadas por JPPF; alta escalabilidad, pudiendo agregar nodos automáticamente; Seguridad de la comunicación mediante cifrado de la información con TLS/SSL y soporte a la computación voluntaria, con las características que le representan como la utilización reducida e inofensiva de los ciclos de cómputo del procesador [14].

- **OpenMosis**

Es una plataforma para distribución de tareas basado en Linux que permite agregar múltiples nodos para generar una súper computadora que procesa grandes cantidades de información. Las ventajas de este son la gran capacidad de añadir nodos de manera dinámica tal que al aparecer nuevos nodos serán agregados al clúster sin requerir de una configuración muy compleja. La comunicación del server con los nodos se realiza a través de SSH. Este software fue diseñado como un ISO que se porta en un CD. Otra de las grandes ventajas es que las aplicaciones que actualmente trabajan sin uso compartido de memoria, se les hace transparente el uso de la distribución de proceso y esto implica entonces que no se debe modificar el código fuente [15].

- **APACHE MESOS**

Plataforma de clúster escrita originalmente en C++ bajo el nombre de Nexus y que para el año 2016 se bautizó como Apache Mesos, que al igual que BOINC fue desarrollado en la universidad de Bakerley. Ésta plataforma permite ser desplegada en Dockers y aunque ha sido construido basado en el kernel de

Linux permite que su despliegue se realice en Windows o MacOS. Ésta aplicación tiene como foco la distribución de grandes volúmenes de datos, más que de procesos, por lo cual la hace un framework ideal para el Big Data. Cuenta con un componente central que habilita la tolerancia a fallos de manera magistral, siendo este atributo uno de sus más destacados beneficios.

La arquitectura de Apache Mesos ha sido creada pensando en un componente esencial correspondiente al framework que le utiliza; hoy en día muy conocidos frameworks como hadoop, spark y kafka hacen uso de esta plataforma para la distribución de data. Otro atributo importante en Apache Mesos es su alta disponibilidad pues permite tener varios nodos en modo Master en modo activo pasivo para tener contingencia en caso de que el master pierda conectividad.

La escalabilidad de Mesos aunque es alta, tiene un número límite que corresponde a 10.000 nodos para ser administrados por el recurso central.

- **Charity Engine**

CharityEngine es una plataforma de procesamiento paralelo y distribuido con una filosofía de prestar un servicio de procesamiento, tal que las personas que deciden instalar la aplicación para “prestar” su capacidad de procesamiento en un PC personal, recibirán una compensación que no necesariamente es económica. Ésta plataforma está basada en BOINC pero incluye módulos de compensación para medir la cantidad de procesamiento entregado de tal manera que pudiera compensarse de manera medible.

Hereda todos los beneficios técnicos de la plataforma BOINC pero agrega funciones de compensación para atraer participantes y aumentar la red de nodos disponibles para el procesamiento.

2.2.2. Criterios de selección de plataforma de paralelismo

Los siguientes criterios han surgido de un análisis de la necesidad de la compañía para la cual se presenta el trabajo en curso y teniendo en consideración los beneficios y facilidades que implicaría para el desarrollo de este ejercicio.

1. Lenguaje de programación
2. Tipo de licencia
3. Sistema operativo
4. Facilidad de implementación
5. Uso en dispositivos móviles
6. Posibilidad de parametrizar uso de máquina
7. Escalabilidad

2.2.3. Métricas de criterios

Las métricas de los criterios seleccionados han sido otorgadas teniendo en consideración la previa selección de la plataforma a adecuar puesto que criterios como el lenguaje de programación sólo podrían ser calificados tras haber escogido esta aplicación.

Tabla 4: Criterios de evaluación de plataforma de programación paralela y distribuida.

Criterio	Calificación	Valores	Totales
Lenguaje de programación	Java	5	40%
	Java y otros	5	
	Tiene API para Java	3	
	C++ y otros	1	
Licencia	Libre		20%
	Paga		
Sistema operativo	Windows	3	15%
	Windows, Linux, MacOS	4	

	Windows, Linux, MacOS, Android	5	
	Linux	1	
Facilidad de implementación	Muy fácil	5	5%
	Fácil	3	
	Difícil	2	
Uso en dispositivos móviles	Sí	5	5%
	No	3	
Posibilidad de parametrizar uso de máquina	Si	5	10%
	No	2	
Escalabilidad	ilimitada	5	5%
	Limitada Alta	4	
	Limitada Baja	2	

2.2.4. Análisis de la selección de criterios

Los criterios seleccionados tienen relevancia para la empresa a la cual se le realiza la implementación pero adicionalmente al trabajo de la tesis pues tendrá un impacto en el tiempo de culminación de la misma. El siguiente es un análisis de cada uno para la claridad de la selección de los mismos:

Lenguaje de programación: Éste punto tiene principal relevancia pues la aplicación seleccionada para ser intervenida está escrita en JAVA. Bien podría implementarse mediante API's pero esto agrega un componente extra generando degeneración al tiempo de respuesta y mayor riesgo a fallos por el componente adicional.

Licencia: Éste criterio es muy importante pues al ser un proyecto de investigación académico, aún no cuenta con recursos económicos de inversión de la compañía para decidir por una plataforma que tuviera costo. La licencia por lo tanto debería ser de libre uso y explotación pues en un plan piloto la empresa hará uso de esta plataforma hasta que defina qué se puede y debe adoptar para otras aplicaciones que tuvieran la necesidad de disminuir su tiempo de respuesta.

Sistema operativo: La empresa actualmente cuenta con licencias de Microsoft Windows y todos sus equipos de cómputo para los empleados tienen instalado alguna distribución de Windows, sin embargo, hay dispositivos móviles tanto iOS como android, y en éste orden de ideas, será importante para determinar el número de nodos que se pudieran agregar a la red dependiendo de qué tanto pueda distribuirse.

Facilidad de implementación: Al ser un proyecto piloto con tiempo y recursos limitados, resulta importante tener una plataforma que de una facilidad alta al momento de realizar la implementación; y esto es, el conjunto de configuraciones, capas intermedias de software, componentes a desplegar, etc.

Uso en dispositivos móviles: Es un plus para el esquema de trabajo el hecho de que la plataforma cuente con aplicación de nodos o clientes para dispositivos móviles, pues habrá entonces un mayor número de posibles participantes en la red de cómputo.

Posibilidad de parametrizar el uso de máquina: Éste criterio hace referencia a la posibilidad de indicarle a la plataforma la cantidad de ciclos de procesamiento o porcentaje de uso sobre el procesador para no malograr el trabajo que desempeñan los funcionarios en la empresa.

Escalabilidad: Se dará una calificación con un peso de relevancia baja puesto que para el ejercicio no se abordará un número muy elevado de nodos participantes, pero para el estudio en general resulta de interés proponer una plataforma que le permita a la empresa crecer tanto como se requiera y como sea posible con los dispositivos con los que cuente la empresa.

Tabla 5: Calificación de plataformas

	Lengua je de progra mación	N o t a %	50 %	Sistema operativ o	N o t a %	Facilida d de implem taci ón	N o t a %	Paramet rización de % de uso de procesa dor	N o t a %	10 %	Escala bilida d	N o t a %	10 %	TOTAL		
BOINC	Phyton	1	0,5	Windows / MacOs / Linux / Android	5	1	Difícil	2	2	Sí	5	0,5	Ilimita da	5	0,5	2,7
JPPF	Java y .Net	5	2,5	Windows / MacOs / Linux / Android	5	1	Muy fácil	5	5	Sí	5	0,5	Ilimita da	5	0,5	5
openMo six	C	1	0,5	Linux	1	2	Fácil	3	3	Sí	5	0,5	Ilimita da	5	0,5	2
Apache Mesos	Java	5	2,5	Windows / MacOs / Linux / Android	5	1	Fácil	3	3	No	2	0,2	Limita da alta	3	0,3	4,3
Charity Engine	Phyton	1	0,5	Windows / MacOs / Linux / Android	5	1	Difícil	2	2	Sí	5	0,5	Ilimita da	5	0,5	2,7

2.2.5. Importancia de los criterios para el negocio

La utilización de la programación paralela y distribuida tiene muchos beneficios que aportan a los atributos de calidad de software como el tiempo de respuesta, la escalabilidad, la alta disponibilidad; adicionalmente, existen beneficios que aportan a la operación del negocio mejorando su desempeño bajo las premisas tácticas y estratégicas. La aplicación seleccionada, tiene como objetivo la liquidación de las comisiones de las (os) vendedores o asesores comerciales en punto de venta. Para STF Group, los puntos de venta físicos representan a la fecha, el canal que más aporta en términos de facturación y contribución total al negocio. La comisión para los asesores es un importante aliciente que impulsa fuertemente la venta; por esto, resulta fundamental tener un adecuado plan de comisiones que motive constantemente al vendedor a esforzarse por tener tasas de conversión más altas.

Con el actual tiempo de respuesta de la liquidación de las comisiones, y la repercusión que genera al liquidar para el ERP y por ende a todas las operaciones de la compañía; este es un proceso que se realiza de manera nocturna y que no tiene actualizaciones durante el día. El área comercial ve un potencial muy grande al poder contar con tiempos de respuesta adecuados sobre la plataforma pues esto les habilitaría la posibilidad de liquidar varias veces en el día y generar nuevas estrategias para impulsar la venta creando nuevos incentivos de comisión con estrategia de competencia entre tiendas y zonas para promover que se haga más participación en la venta, mayor ticket promedio por cliente, mayor venta por estado promocional del producto, etc.

Para el presente proyecto, los criterios habrán de ser aquellos medibles durante el tiempo propuesto para la entrega; sin embargo, otros criterios de negocio determinarán al interior de la compañía, el éxito de la implementación; y se ha de considerar que hay una directa relación entre estos criterios.

2.2.6. Criterios de evaluación de impacto de la implementación

Los criterios que se plantean para evaluar el impacto de la implementación son:

1. Tiempo de respuesta.

Éste criterio es el más básico y corresponde a la medición del tiempo que tardará la aplicación en realizar la ejecución de la liquidación de las comisiones de los asesores comerciales. Se deberá otorgar una calificación que dependerá de los resultados obtenidos y esta calificación aunque numérica, será avalada por el negocio para contrastar con la expectativa del cambio.

Considerando el actual tiempo de 7 horas para la culminación de la liquidación de las comisiones, los tiempos de respuesta posible esperada y sus calificaciones serían:

Entre 1 hora y 2 horas -> 3

Entre 30 minutos y 59 minutos -> 4

Entre 1 minuto y 30 minutos -> 5

2. Consumo de recursos

El criterio propuesto, determinará el impacto en términos del consumo de las máquinas tanto de nodos cliente como de servidores. Éste criterio es de gran importancia para determinar el impacto o huella que deja el proyecto a la empresa.

Entre el 10% y el 20% de consumo de CPU -> 3

Entre el 5% y el 10% de consumo de CPU -> 4

Entre el 1% y el 5% de consumo de CPU -> 5

3. Percepción de usuarios clientes



Este criterio está muy relacionado con el “consumo de recursos”, pero será entregado por los usuarios que participarán del experimento, ya que se pretende medir el impacto en el trabajo que desarrollan las personas que utilizan los recursos que se aprovecharán en la ejecución paralela y distribuida.

Puedo percatarme que todo está más lento -> 1

En algunos momentos tengo lentitud -> 3

Imperceptible -> 5

4. Número de ejecuciones posibles

Se analizará dependiendo de los picos de uso de los recursos informáticos en la compañía, cuánto es el número máximo de ejecuciones que se podrán realizar para mantener la liquidación de las comisiones lo más cercano al tiempo real, para que a partir de ésta, el negocio pueda realizar planteamientos de estrategias de venta.

Entre 2 y 3 veces al día -> 3

Entre 4 y 6 veces al día -> 4

Mayor a 6 veces al día -> 5

Para realizar esta calificación se realizará un análisis del tiempo de consumo en base de datos del ERP, bloqueo de tablas, tiempos críticos de la base de datos y alta demanda en los equipos de cómputo de las personas que tendrán nodos instalados.

2.3. Desarrollo

2.3.1. Análisis

2.3.2. Ingeniería inversa

La aplicación seleccionada para realizar el ejercicio de paralelizar sus subprocesos se llama Binaps Comisiones y se trata de una aplicación web que permite almacenar las metas de venta de las tiendas para posteriormente realizar

el cálculo de la comisión al personal que le aplique. Las reglas y premisas para dicho cálculo son las siguientes:

1. De manera centralizada, el área comercial analiza el movimiento de ventas año a año y establece metas mensuales por tienda.
2. Los jefes de tienda distribuyen la meta mensual en metas diarias basadas en el histórico de ventas y los días de la semana que aplican.
3. Las jefes de tienda distribuyen entre las asesoras de venta el presupuesto de venta diario acorde al rango horario del turno que tengan asignado.
4. El personal de tienda diferente al asesor comercial obtendrá una comisión si la tienda total logra superar la meta.
5. Las asesoras comerciales no dependen del cumplimiento de meta de la tienda sino de su propio cumplimiento mensual.
6. El esquema de cálculo no es binario indicando que cumple o no cumple; se cuenta con tablas de comisión en donde se indica el rango de cumplimiento y el valor porcentual sobre la venta lograda a recibir por cada asesor.
7. El personal diferente al cargo de asesor de ventas recibe un valor fijo si la tienda cumple la meta de venta.
8. Las devoluciones en venta o 'Nota crédito' constituyen una disminución en la acumulación de venta tanto para la tienda como para el asesor.
9. Existen denominaciones de tipo de tienda para asignar diferentes estrategias de comisión basada en su venta y exposición al público.
10. Existen tablas de comisión por 'tipo de tienda'.

2.3.3. Análisis de arquitectura

La aplicación está construida en J2EE montada sobre Glassfish 3.1 con hibernate para el mapeo relacional. Ésta aplicación ha sido creada con una distribución de paquetes de una concepción dada por la utilidad. Existe un conjunto de paquetes básicos que contienen los objetos transversales a toda la aplicación y un paquete muy importante que contiene una interfaz web para llamados Web Service SOAP con el cual se automatizó el proceso de ejecución del cálculo de la liquidación de la comisión pues al delegar la función al usuario la opción de la liquidación producía caídas por el alto tráfico de solicitudes a final de mes en el cual las asesoras querían consultar constantemente su estado de cumplimiento.

- Interfaces:

La aplicación web realiza dos conexiones principales; conexión con el sistema gestor de recursos humanos (Kactus) y el ERP de la compañía (Siesa Enterprise UNOEE). El ERP dispone de un módulo POS que centraliza las operaciones de las terminales de pago POS y recibe la información de las ventas asociadas a los centros de operación o tiendas, vendedores, documentos, etc. La conexión que se establece con el ERP es la que tiene alta relevancia para este proyecto, pues representa el reto más importante a ser dividido en múltiples tareas. La interfaz de la aplicación de comisión con el ERP se establece a través de ODBC utilizando vistas de base de datos (SQL Server 2008 R2) que se encargan de extraer las ventas y vendedores asociados.

La aplicación cuenta con una base de datos propia para el almacenamiento de los datos maestros de la misma, los presupuestos y la liquidación generada por el proceso automatizado.

El siguiente gráfico representa la arquitectura de la aplicación en un diagrama de despliegue:

Como se puede observar en la figura 3, la solución tiene la capa de aplicación y base de datos separadas y las bases de datos del aplicativo es independiente de

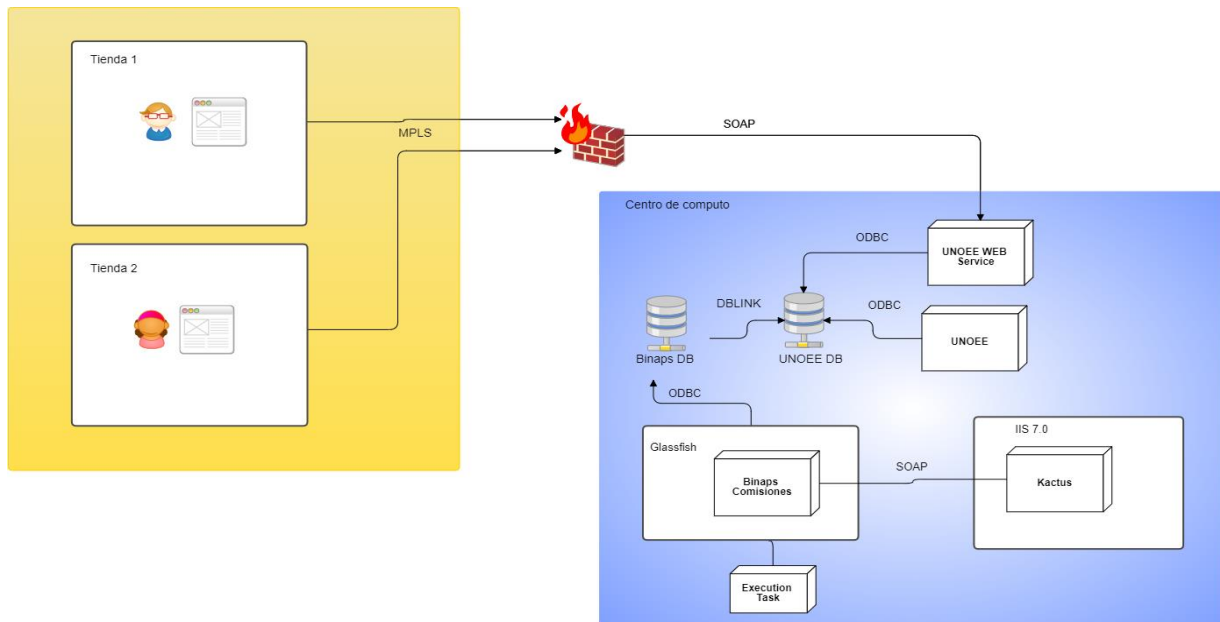


Figura 3: Diagrama de Despliegue

la base de datos del

ERP (UNOEE). Un componente de gran importancia en el proceso de liquidación corresponde a una tarea programada como servicio Windows que ejecuta el servicio de liquidación de la comisión sobre Binaps, ya que ésta aplicación corresponde a un aplicativo web, y los eventos provienen de una programación con Spring Quartz. El DBLink creado entre las dos bases de datos (Binaps y UNOEE) Permite ejecutar consultas a ambas bases de datos desde un mismo Script para hacer JOIN entre los resultados obtenidos en esquemas diferentes.

2.3.4. Descripción de algoritmo.

El proceso inicia con el componente de servicio windows que dispara el evento de ejecutar la liquidación de nómina llamando al web service dispuesto en Binaps para tal efecto; El flujo se da de la siguiente manera:

1. Se llama al método `com.stfgroup.integraciones.liquidarcomision.ws`.
2. El método del paso 1 llama al método `calcular` de la clase `com.stfgroup.integraciones.liquidarcomision.calculo`
3. El método `'calcular'` consulta las tiendas que tienen un presupuesto asignado con el método `GetTiendas` de la clase `com.stfgroup.integraciones.liquidarcomision.calculo`.
4. Recorre cada resultado de tiendas y con el método `'GetEmpleados'` extrae los empleados que tienen un presupuesto para la tienda en que se está iterando.
5. Recorre el resultado de `GetEmpleados` y con el método `'GetVtasUnoee'` consulta las ventas que realizó dicho empleado.
6. En cada iteración calcula la comisión del empleado y llama al método `'GetValComision'` que verifica basado en el porcentaje de cumplimiento, cuánto dinero comisiona el empleado.
7. Almacena la comisión del empleado.

Éste algoritmo tiene 3 ciclos anidados y el más anidado de todos consulta las ventas cada vez sobre las mismas tablas del ERP (Cabecera y detalle) que pueden llegar a tener hasta 250 millones de registros (el detalle de las ventas), de tal manera que éste punto específico genera el mayor tiempo para la respuesta final. Adicionalmente, crea interbloqueos con los demás procesos de la compañía, por lo que solo puede ser ejecutado en horarios no hábiles.

- Diagrama de secuencia

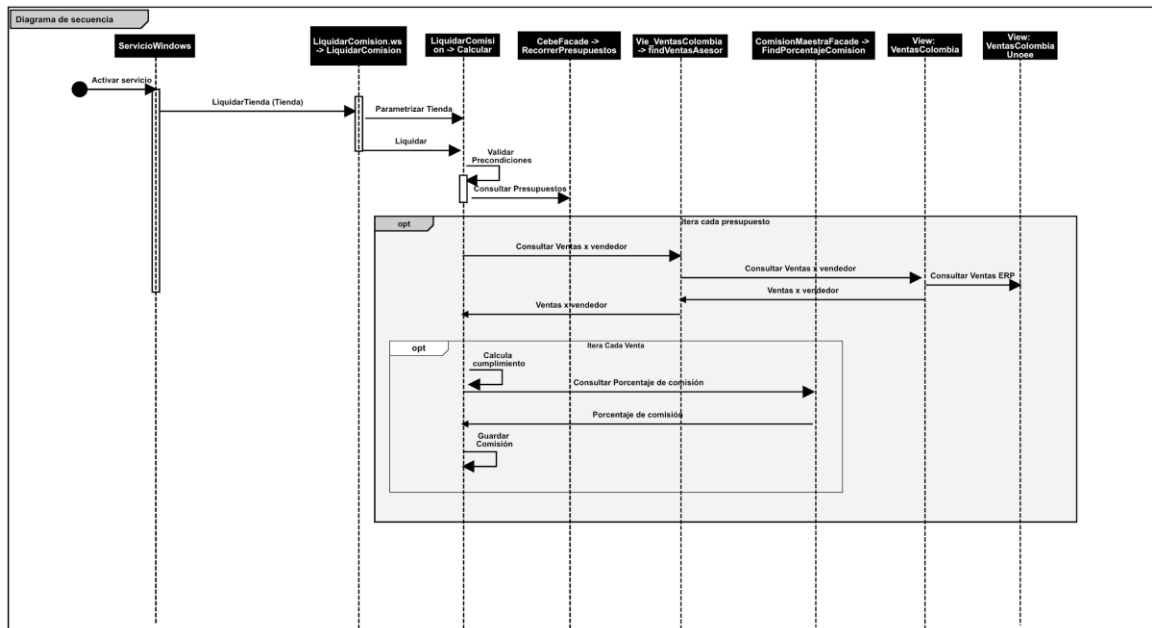


Figura 5: Diagrama de secuencia.

Este diagrama de secuencia nos resulta muy relevante para entender que el proceso de liquidación depende de una secuenciación de actividades con dependencia una de la otra de tal manera que no hay nada paralelo en la ejecución. Así entonces, se tiene además una constante consulta a la base de datos con los mismos campos resultantes y con cruces entre tablas que resulta en un esfuerzo del ERP para dar respuesta a múltiples llamados, haciendo que la sumatoria total de tiempos de respuesta no sea la ideal.

- Flujo de trabajo antes y después

En las figuras 7 y 8 podemos ver la variación entre la solución previa en la cual las consultas y cálculos se hacen secuencialmente con dos ciclos anidados que iteran las tiendas y empleados yendo cada vez a la base de datos a extraer todas las ventas y filtrar por tienda-empleado. En la figura 7, estos ciclos son llevados a cada nodo de cómputo en el cual se itera consultando sobre un conjunto de datos en JSON cargados en un arreglo y usando el método stream de Java (iteración implícita). Ésta solución resulta en una mejora notable dado que los datos a iterar son un subconjunto del total de base de datos que está distribuida en múltiples nodos sin tener un recurso compartido que genere bloqueos por múltiples consultas paralelas.

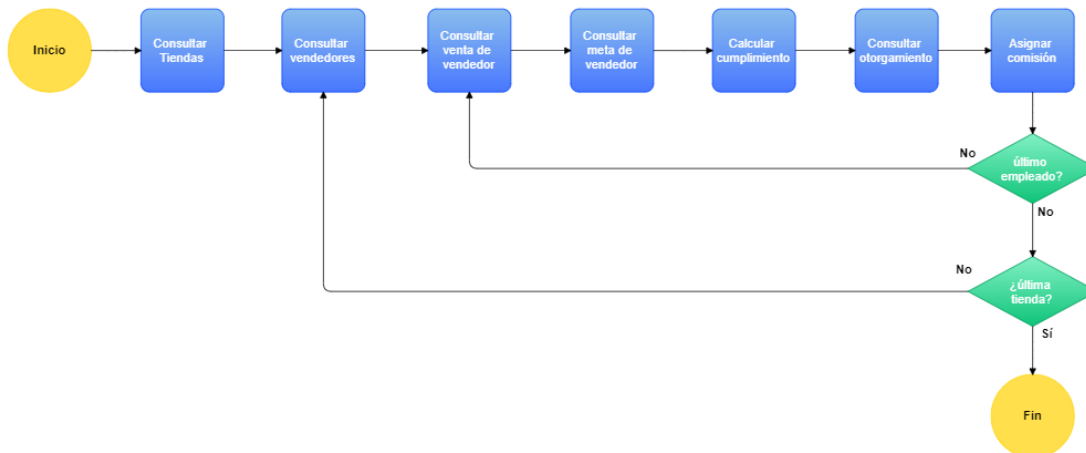


Figura 7. Flujo grama de solución antes de mejoras.

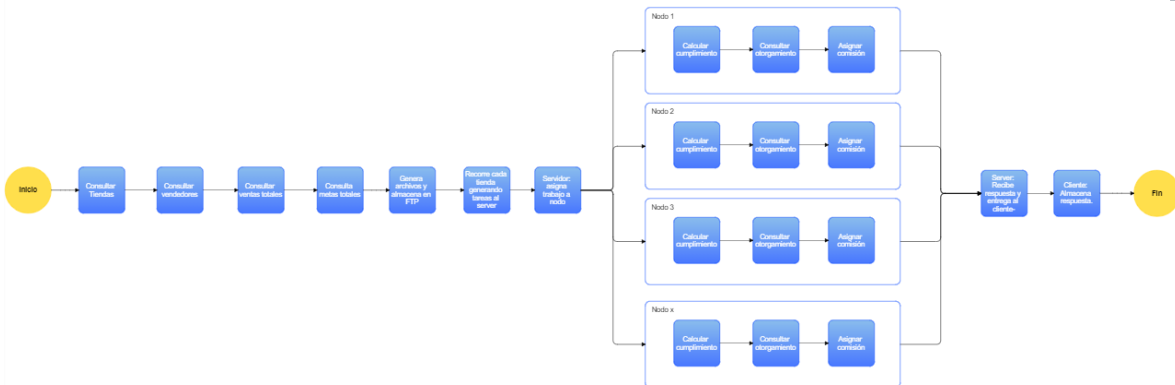


Figura 8. Flujograma con paralelismo de procesos.

En la figura 9 hacemos un foco en el punto de ejecución de cada nodo y en él mostramos como se dan 3 ciclos anidados (uno explícito y dos implícitos) para llegar al valor necesario de cálculo de cada empleado en la tienda asignada al nodo, para finalmente almacenarlo en el FTP y dar respuesta al server.

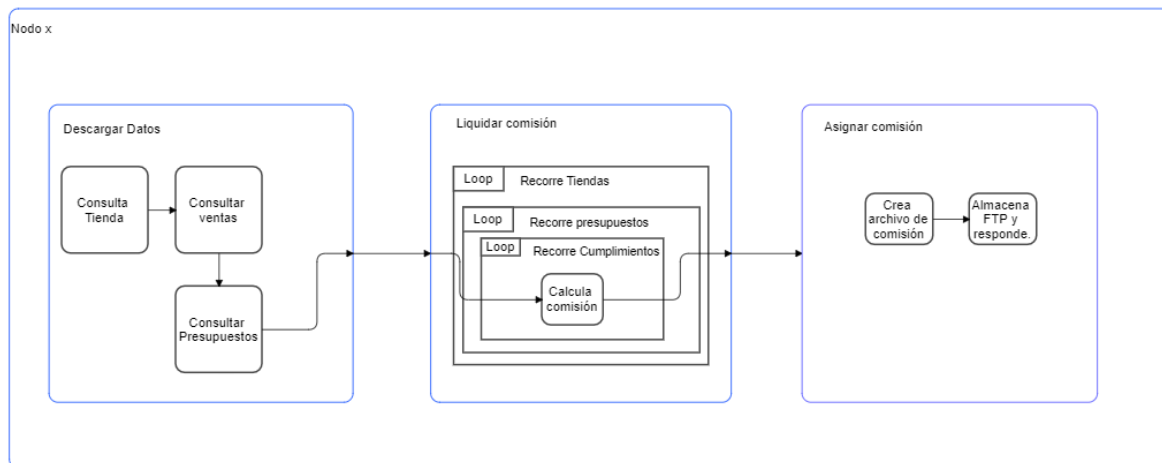


Figura 9. Detalle de ejecución en nodos.

En la figura 10 tenemos el código correspondiente a la consulta del repositorio de archivos para cargar en memoria los datos de ventas, cumplimientos, presupuestos y la tienda asignada. Adicionalmente, se llama al proceso de liquidación y el resultado es enviado al FTP.

```
22 import java.text.DecimalFormat;
23 import java.util.stream.Collectors;
24 import org.apache.commons.net.ftp.FTPClient;
25
26 /**
27  *
28  * @author jserrate
29  */
30 public class ClienteComision {
31
32     /**
33      * @param args the command line arguments
34      */
35     static List<RegistroCumplimiento> tablaCumplimiento;
36
37     public static void main(String[] args) {
38
39         Tienda laTienda = leerJsonLaTienda();
40         List<Tienda> tiendaVentas = getTiendaVentas(laTienda.getNombre());
41         tablaCumplimiento = leerJsonRegistrosCumplimiento();
42         List<Presupuesto> presupuestos = getPresupuestosTienda(laTienda.getNombre());
43         List<LiquidacionC> liquidacion = liquidarTienda2(tiendaVentas.get(0), presupuestos);
44
45         GuardarLiquidacion(liquidacion);
46     }
}
```

Figura 10. Método principal con orden de secuencia.

El código de la figura 11 corresponde al método de liquidación de la comisión que tiene la complejidad más alta en términos del tiempo de ejecución, pues este realiza iteraciones entre los diferentes arreglos buscando el dato específico de cada empleado, filtrando las ventas que le corresponden y que a partir de la venta lograda, el cumplimiento que genera un porcentaje de comisión sobre la venta total lograda.



```
public static List<LiquidacionC> liquidarTienda2(Tienda TiendaVentas, List<Presupuesto> presupuestos) {  
  
    List<LiquidacionC> liquidados = new ArrayList<LiquidacionC>();  
    DecimalFormat df = new DecimalFormat("###.##");  
  
    try {  
        for (int i = 0; i < TiendaVentas.getVentas().size(); i++) {  
            final String vendedor = TiendaVentas.getVentas().get(i).getVendedor();  
            List<Presupuesto> ListpresupuestoVendedor = presupuestos.stream().filter(item -> item.vendedor.equals(vendedor)).collect(Collectors.toList());  
  
            float cumplimiento;  
  
            if (ListpresupuestoVendedor != null && ListpresupuestoVendedor.size() > 0) {  
                Presupuesto presupuestoVendedor = ListpresupuestoVendedor.get(0);  
                int ventaVendedor = TiendaVentas.getVentas().get(i).getTotal();  
                cumplimiento = ventaVendedor / presupuestoVendedor.getTotal();  
                final float valCumplimiento = cumplimiento * 100;  
  
                List<RegistroCumplimiento> ListRegCumplimiento = tablaCumplimiento.stream().filter(x -> x.cargo.equals(presupuestoVendedor.cargo)  
                    && x.v_inicial <= valCumplimiento && x.v_final >= valCumplimiento).collect(Collectors.toList());  
                if (ListRegCumplimiento != null && ListRegCumplimiento.size() > 0) {  
                    int comision = (int) ((ListRegCumplimiento.get(0).v_cumple / 100) * ventaVendedor);  
  
                    liquidados.add(new LiquidacionC(TiendaVentas.getNombre(), TiendaVentas.getVentas().get(i).getVendedor(), cumplimiento, comision));  
                }  
            }  
        }  
        return liquidados;  
    } catch (Exception e) {  
        return null;  
    }  
}
```

Figura 11. Método de liquidación que ejecuta cada nodo.

3. Pruebas y resultados

Para corroborar el funcionamiento de los ajustes realizados a la aplicación de liquidación de comisiones, se realizó un conjunto de ejecuciones en diferentes franjas horarias y en combinación con diferente número de nodos Cliente. El objetivo de hacer esta combinatoria de variables para la ejecución corresponde a validar el impacto que tiene sobre los resultados el hecho de ejecutarlo en momentos de mayor o menor uso de la base de datos tanto de binaps como del ERP Siesa. Adicionalmente, al contrastar estos resultados con la ejecución con diferentes número de nodos, podemos corroborar el efecto generado al fraccionar las tareas de cálculo de comisión por tienda en diferentes nodos, haciendo una distribución mayor, que por lógica pudiera apuntar a tener mejores tiempos de respuesta, pero que deberían corroborarse en laboratorio.

El siguiente cuadro muestra los números obtenidos en términos del tiempo de ejecución:

Tabla 6: Tiempos de respuesta por horarios y número de nodos.

NODOS	HORARIO					
	6am-8am	8am-12am	12m-4pm	4pm-8pm	8pm - 12pm	PROMED IO
1	7:04:30	12:13:54	9:34:12	14:44:21	8:16:35	10:22:42
2	0:02:02	0:03:07	0:02:22	0:04:08	0:02:02	0:02:44
4	0:01:52	0:01:12	0:02:25	0:04:00	0:03:54	0:02:41
6	0:01:44	0:02:25	0:01:14	0:03:45	0:03:43	0:02:34
8	0:01:40	0:02:40	0:01:43	0:04:01	0:03:30	0:02:43
10	0:01:35	0:01:50	0:01:12	0:03:52	0:03:23	0:02:22
PROMED IO	0:01:47	0:02:15	0:01:47	0:03:57	0:03:18	0:02:37

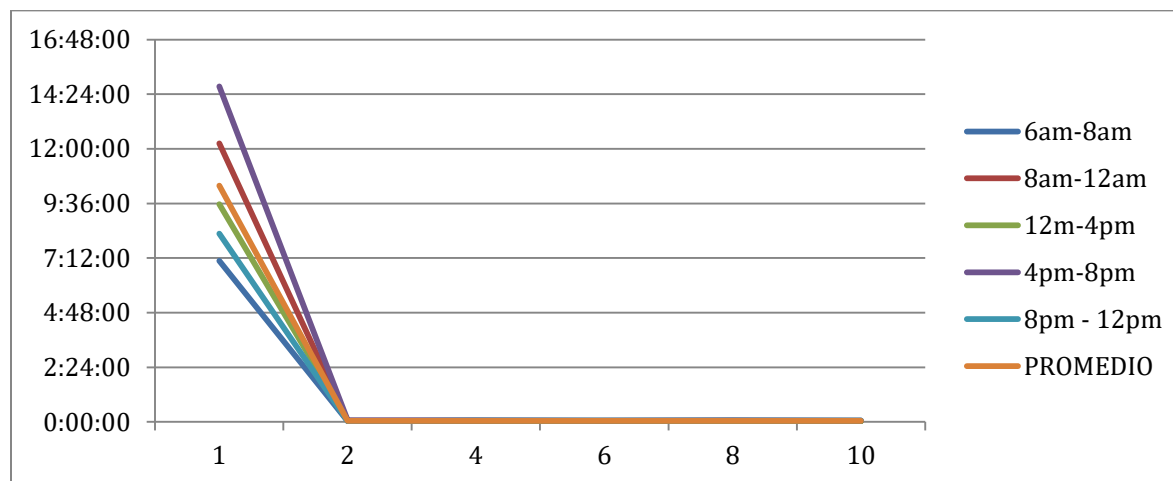


Figura 12. Tiempos antes y después de distribución por horarios.

3.1. Análisis

Al ejecutar las pruebas documentadas en la Tabla 6, ejecutando el proceso en los diferentes escenarios con nodos y horarios, se puede observar como hay ciertas tendencias a la disminución de tiempo de respuesta en la medida que se aumenta el número de nodos. Sin embargo, hay algunos valores que se salen de dicho patrón, y esto se puede deducir porque al modificar el comportamiento de la aplicación, los tiempos más impactantes están al momento de extraer la información del ERP correspondiente a las ventas de los asesores comerciales. Estas tablas, son altamente utilizadas por todos los sistemas que están involucrados en las operaciones de la compañía como los POS (aproximadamente 700 cajas POS activas ingresando y consultando información de manera simultánea), el sistema de BI que extrae información de ventas cada 2 horas para generar reportes de avance en la venta, el sistema de fidelización, que valida que un cliente cumpla con las políticas de mínima compra para fidelizar un cliente, entre muchas otras. Así las cosas, quien más genera un impacto en la respuesta final es el ERP y su base de datos quien tiene un nivel de consulta muy elevado. En la medida que el número de registros a calcular por tienda aumente, se vería una mayor diferencia cuando se aumentan o disminuyen nodos al sistema de cálculo.

Esta es la forma en que se afecta el tiempo de respuesta con respecto a la franja horaria de la ejecución con los tiempos promedio:

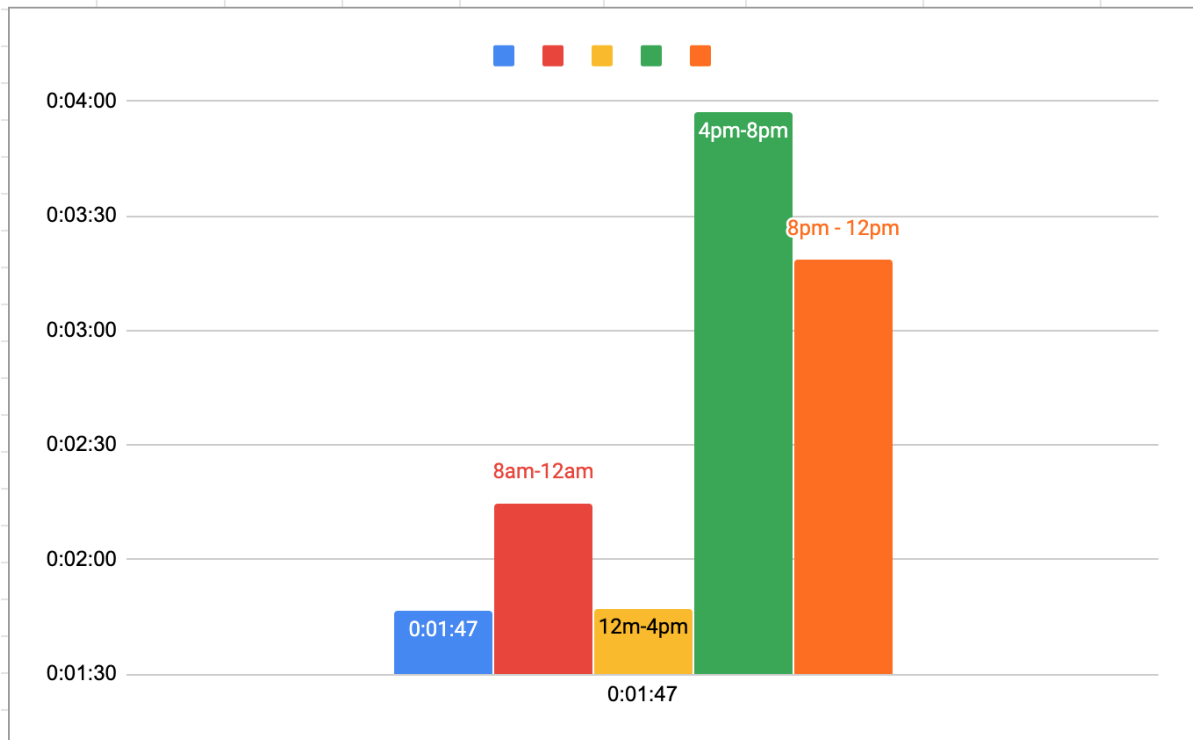


Figura 13: Tiempos promedio por franja horaria.

Los horarios comprendidos entre las 4 de la tarde y las 8 o 9 de la noche tienen tiempos más prolongados de respuesta del motor de datos puesto que son los horarios picos de ingreso de clientes a las tiendas, por lo cual, hay una constante sincronización de ventas hacia el ERP generando bloqueos o encolamiento en la base de datos.

El siguiente gráfico representa los tiempos promedios de ejecución de la liquidación de comisiones utilizando diferentes números de nodos clientes que procesan cada tienda:

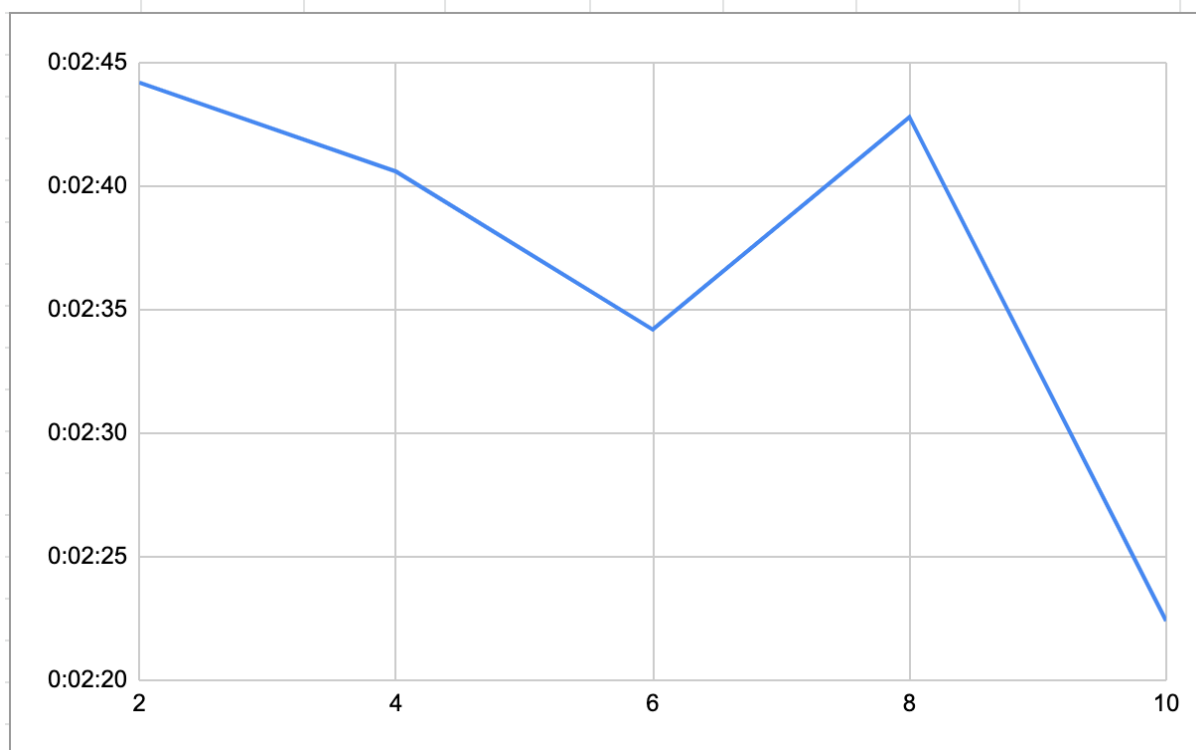
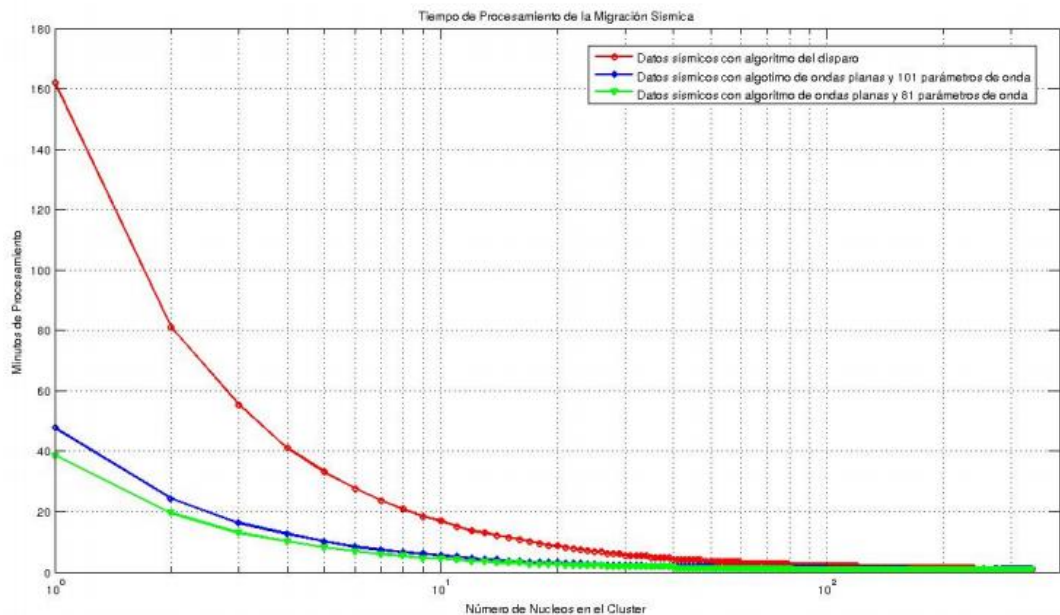


Figura 14: Tendencia de tiempos por número de nodos.

Aquí podemos ver que se presenta tendencia de disminución de los tiempos de ejecución en la medida que se le adhieren nodos al sistema; sin embargo, no es un comportamiento muy marcado y no representa disminuciones tan grandes comparados con cada bloque de nodos superior.

3.1.1. Comparativo

Realizando una comparativa con otros trabajos enfocados en la distribución de procesos, podemos encontrar tesis como [16][17] en las cuales hay una correspondencia de mejora de tiempos con el aumento de número de nodos introducidos en la solución. En las imágenes siguientes se observa como en la medida que se introducen nodos al sistema, los tiempos de ejecución disminuyen, y al igual que en éste trabajo, es representado por una curva en función exponencial que tiende a regularse a partir de un número de nodos. Así las cosas, se puede observar que a partir de un número de nodos, la curva tiende a aplanarse y no agrega un valor mayor en la reducción de los tiempos, en éste ejercicio específico está dado por la reducción mayor que corresponde a la consulta del ERP con extracción de ventas, presupuestos y tiendas.



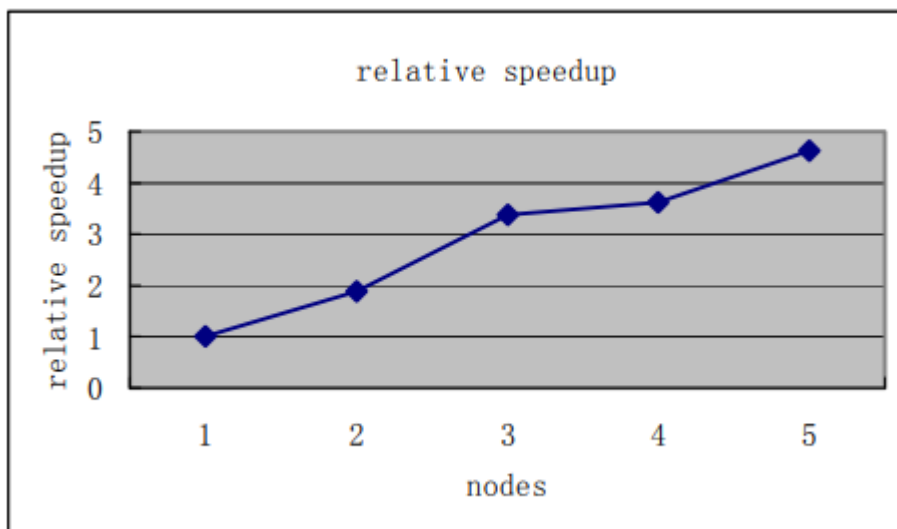


Figura 16. Jing Xiong, Jianliang Wang, Jianliang Xu. 2010. Velocidad relativa [16].

3.2. Calificación de impacto

Para calificar los resultados obtenidos en las ejecuciones de prueba, analizamos cada punto acorde a los criterios seleccionados en la sección 2.2.6, de la siguiente manera:

1. Tiempo de respuesta: Promedio 2 minutos 37 segundos, con lo que la calificación para este criterio sería de 5.
2. El consumo de recursos de las máquinas clientes estuvo alrededor del 1% del procesador haciendo una revisión visual del task manager de windows, por lo que su calificación sería de 5.
3. Percepción de los usuarios clientes: Para el particular, se realizó entrevista a las 4 personas participantes, a lo cual concluyeron que no era perceptible el momento en que los clientes iniciaban una ejecución de liquidación. Así las cosas, la calificación de este criterio es 5 siendo consecuente con las métricas definidas para éste criterio.
4. Número de ejecuciones posibles: Éste criterio se determina a partir de los 3 puntos anteriores. Dado que su ejecución no toma más de 3

minutos, teóricamente podría ejecutarse 240 veces en el día sin que se solapen las solicitudes; por lo cual su calificación también es de un 5.

3.3. Conclusiones

Hemos realizado un ejercicio basado en la teoría de que distribuir las tareas y paralelizarlas nos darán un mejor resultado en términos de performance, haciendo que los tiempos de ejecución disminuyan dramáticamente. La teoría y la lógica nos indicaban que en la medida que se añadieran nuevos nodos al sistema, éste nos arrojaría mejores resultados, obteniendo un resultado final más óptimo para la empresa objeto de estudio; utilizando una plataforma de gran capacidad que permite al desarrollador abstraerse de todas las tareas que implican la búsqueda en red de los nodos, distribución de tareas acorde a la capacidad del sistema de nodos, recopilar resultados y mantener el control de las tareas no ejecutadas; se logró implementar un sistema distribuido de liquidación de comisiones, que como principal característica tiene el objeto de re utilizar el compute inutilizado en los equipos de cómputo de la empresa dado que naturalmente, los equipos utilizados en software de poco esfuerzo como ofimática, comunicaciones y demás, no utilizan todo el potencial de que vienen dotados los equipos.

El resultado nos ha demostrado que evidentemente el ejercicio de dividir y paralelizar las tareas, para el contexto específico, disminuye de manera bastante notoria, con valores por debajo del 0,48 % del tiempo original. Los tiempos de ejecución se vieron fuertemente impactados al disminuir de 7 horas en promedio a un promedio de 2:27 minutos; esto posibilita al negocio a generar nuevas estrategias de consecución de ventas impulsando la sana competencia entre asesores, tiendas, zonas, canales de venta, etc.

Aunque la mayor parte de la mejora se debe al hecho de no realizar múltiples consultas a las tablas de ventas del ERP, la estrategia de división de la ejecución, tiene inherente la consulta unificada de ventas para distribuir entre nodos, que conlleva a una ejecución más rápida.

Un trabajo futuro podría aportar fuertemente a mejorar la arquitectura empresarial de STF Group S.A y cualquier otra empresa que se enfrente a éste tipo de dificultades de performance en aplicaciones de gran volumen de datos y procesamiento masivo, generando una metodología estandarizada de desarrollo cuando aplique, utilizando de manera centralizada el sistema de nodos completo de



la empresa, que no solo habría de aplicarse a equipos de cómputo personales sino también a servidores para que la distribución del trabajo mejoren la utilización de los últimos ya que usualmente las aplicaciones desplegadas suelen tener picos de uso de máquina dejando un uso de mesurado a mínimo para otros momentos.

A nivel académico, podría darse una continuidad a éste trabajo para comparar los resultados con plataformas tan altamente conocidas como BOINC y tener una carta clara de la conveniencia de utilizar el uno o el otro dependiendo de las necesidades puntuales de una empresa.

4. REFERENCIAS BIBLIOGRÁFICAS

- [1] Revista Dinero. (2017). Blockchain, robótica y realidad virtual: las tecnologías avanzadas menos adoptadas. Recuperado el 9 de septiembre del 2019. <https://www.dinero.com/economia/articulo/primeros-resultados-del-observatorio-de-economia-digital-mintic/253120>.
- [2] MinTic. (2017). MinTIC revela los primeros resultados del Observatorio de Economía Digital. Recuperado el 10 de septiembre del 2019. <https://www.mintic.gov.co/portal/inicio/Sala-de-Prensa/Noticias/61929:MinTIC-revela-los-primeros-resultados-del-Observatorio-de-Economia-Digital>.
- [3] Ministerio de educación (MEN), Observatorio Laboral para la Educación (OLE). https://ole.mineducacion.gov.co/1769/articles-380398_recurso_1.pdf. Recuperado el 15 de septiembre del 2019.
- [4] Asociación Nacional de Empresarios de Colombia – Andi (2017). ENCUESTA DE TRANSFORMACIÓN DIGITAL 2017. Recuperado el 10 de Octubre del 2019. <http://www.andi.com.co/Uploads/Encuesta%20Transformaci%C3%B3n%20Digital%20ANDI.pdf>
- [5] Hernandez Duany, O., Fulgueira Camilo, M., Muñoz Rojas, E., Henry Fuenteseca, V., Reyes Burunate, W., & Insua Suarez, E. (2015). Infraestructura y administración de cómputo paralelo y desarrollo de aplicaciones / Infrastructure and administration of parallel computing and application development. *Revista Cubana de Ciencias Informáticas*, 35–45.
- [6] Christian Benjamin Ries, Christian Schröder, Vic Grout, "Approach of a UML profile for Berkeley Open Infrastructure for network computing (BOINC)", Computer Applications and Industrial Electronics (ICCAIE) 2011 IEEE International Conference on, pp. 483-488, 2011.
- [7] Benavente, J. M. (2018, 3 diciembre). Políticas 4.0 para la cuarta revolución industrial [Publicación en un blog]. Recuperado 10 Septiembre, 2019, de <https://blogs.iadb.org/innovacion/es/politicas-de-transformacion-digital>.
- [8] C. Chang, T. Wei and I. Wu, "Job-level computing with BOINC support," 2016 Conference on Technologies and Applications of Artificial Intelligence (TAAI), Hsinchu, 2016, pp. 200-206.
- [9] M. Black and G. Bard, "SAT Over BOINC: An Application-Independent Volunteer Grid Project," 2011 IEEE/ACM 12th International Conference on Grid Computing, Lyon, 2011, pp. 226-227.
- [10] Y. Wu, J. Cao and M. Li, Private Cloud System Based on BOINC with Support for Parallel and Distributed Simulation. (2011). 2011 IEEE Ninth International Conference on



Dependable, Autonomic and Secure Computing, Dependable, Autonomic and Secure Computing (DASC), 2011 IEEE Ninth International Conference On, 1172.

[11] Alvarez, D., Barrios, J., Kuk Ho Chung, León, R., López, A., Suriano, A., Tomás Prieto, J. (2017). Implementación de un cluster de computadoras. *Revista de La Universidad Del Valle de Guatemala*, (35), 110.

[14] Velarde Martínez, A., Luna Ramírez, E., Soria Cruz, J., & Ponce de León Sentí, E. (2016). Evaluación de objetivos al planificar y asignar trabajos paralelos en un sistema de multicomputadoras / Objective evaluation to plan and allocate parallel tasks in multicomputer system. *Ingeniare. Revista Chilena de Ingeniería*, 24(2), 290–304.

[13] Papadopoulos, P. M. (2011). Extending clusters to Amazon EC2 using the Rocks toolkit. *International Journal of High Performance Computing Applications*, 25(3), 317–327.

[14] JPPF.org. (2005). About JPPF. JPPF. <https://www.jppf.org/about.php>

[15] Santo Oceró, D. (2018, 20 noviembre). OpenMosix user area tools. Internet Archive. <https://web.archive.org/web/20081120145150/http://www.orcero.org/irbis/openmosix/userland.html>

[16] Jing Xiong, Jianliang Wang, Jianliang Xu. Research of Distributed Parallel Information Retrieval Based on JPPF. (2010). *International Conference of Information Science and Management Engineering*.

[17] Camilo Antonio Linares Moreno (2014). Paralelización de algoritmos en plataformas distribuídas: Caso de estudio en procesamiento sísmico en la industria petrolera.



Anexo 1. Tabla de aplicaciones STF Group S.A.

NOMBRE	DESCRIPCIÓN	AREAS USUARIAS
F-PROD	Sistema multipropósito para suplir las necesidades de automatización que no tiene el UnoEE	Contabilidad, Calidad, Modisteria, patronaje, oficina técnica, despachos, CEDI, Programación.
Middleware eCommerce	Herramienta de integración desarrollada en .NET C# como un conjunto de servicios que envían información a la plataforma de comercio electrónico desde el ERP y viceversa.	E-Commerce
F-PED	Sistema para generar pedidos y consolidación para enviar a ERP para generación de las remisiones	Ventas al por mayor
		Franquicias
BINAPS	Sistema de liquidación de comisiones de equipo de ventas en tiendas.	Comercial



ORION	Sistema para extracción de reportes	Comercial
		Alta Gerencia
		Demanda
		Product Manager
REPORTING SERVICES	Sistema para extracción de reportes	Compañía (excepto tiendas)
DATAWAREHOUSE 1	Bodega de datos de la compañía (Antigua)	Informática
DATAWAREHOUSE 2	Bodega de datos de la compañía	Informática
CORNERS	Es un conjunto de varias aplicaciones para soportar el modelo de negocio de consignación de mercancía.	Comercial



Distribución Referencias	Sistema para generar automáticamente los RQI a partir de un archivo Excel	Demanda
ARRENDAMIENTOS	Llevar el control de los arrendamientos de las tiendas	Contabilidad
CUPOS EMPLEADOS	Aplicación para la centralización de cupos de descuento otorgado como beneficios para los empleados de la compañía.	
LEGALIZACIÓN ANTICIPOS	Aplicación utilizada para la autogestión de la legalización de los anticipos entregados a empleados por diferentes conceptos como viajes, compras y demás.	
PEDIDOS PRODUCCIÓN	Sistema para la captura de la información de los pedidos de producción, seguimiento y control de la producción	Programación
REPORTES PROGRAMACIÓN	Consultas y reportes para la programación	Programación



SERVICIOS PÚBLICOS	Aplicación para la gestión de la facturación de servicios públicos de las tiendas y edificios de la compañía.	Contabilidad
CALIDAD TELAS	Sistema de gestión de la calificación de la calidad de las telas adquiridas por la compañía. Aplicación web que permite registrar la revisión visual de las telas y otorgar una calificación de acuerdo al número de hallazgos.	Calidad
COMPRAS INTERNACIONALES	Aplicación cliente servidor que permite gestionar de manera móvil los pedidos de compra a terceros, ya sea de materia prima o producto terminado.	Compras de telas, Compras de producto terminado.
AUTOREGISTRO	Aplicativo web para que los clientes puedan autogestionar su registro al programa de fidelización de cada marca y actualización de datos.	Fidelización



TABLERO DE CONTROL	Reporte central del time to market de la compañía que evalúa los tiempos entre cada punto de interacción de las áreas para la producción y paso a tienda de las colecciones de la compañía.	Programación
Sincronizador Maestros	Sistema para sincronizar maestros a demanda desde PLM hacia el UNOEE	Programación, Costos, CEDI
Actualizador de clientes	Aplicación de escritorio que se utiliza para cargar archivos excel con información de clientes para que sean actualizados en el ERP.	Fidelización
Reembolsos	Aplicación web para contabilizar reembolsos y seguir un flujo de aprobación de los mismos.	Todas



<p>Referencias semiterminadas</p>	<p>Aplicación que crea una referencia copia en México y Colombia con el nombre de un patrón de tela para enviar la tela cortada y no una referencia terminada.</p>	<p>Producción, planeación</p>
<p>Habeas Data</p>	<p>Aplicación para la gestión de registros de aceptación o denegación de tratamiento de datos personales. Se conforma de una estructura de datos para almacenamiento. Una capa de web services para la gestión de llamados a la creación de registros y la generación de tokens</p>	<p>Servicio al cliente, fidelización, inteligencia de negocios, Riesgos, Comercial, mercadeo.</p>
<p>Importación de facturas</p>	<p>Aplicación de escritorio utilizado para importar información de facturas de una empresa en otra base de datos como una entrada y requisiciones para despacho de la mercancía.</p>	<p>Compras países</p>



<p>Garantías</p>	<p>Aplicación web</p> <p>destinada a la gestión del flujo de proceso correspondiente a la solicitud de</p> <p>una garantía de prendas. Involucra las diferentes etapas en las que participan</p> <p>los roles de la compañía que intervienen dentro del flujo de proceso con</p> <p>entrega de información al cliente (sms o mail) para mantenerlo al tanto de</p> <p>estado del mismo.</p>	<p>Comercial,</p> <p>Servicio al cliente, Cedi.</p>
<p>Log centralizado</p>	<p>Aplicación que</p> <p>gestiona de manera centralizada los logs de las demás aplicaciones, de tal</p> <p>manera que se utilice la lógica y estructura de datos, además de permitir una</p> <p>gestión centralizada.</p>	<p>NA</p>



SingleSignOn	<p>Aplicación web</p> <p>que permite la gestión centralizada de la seguridad en torno a la</p> <p>autenticación y autorización. Basada en el Estándar Auth2.0 que permite la</p> <p>autenticación con única clave y re utilización de token de autenticación</p> <p>entre las aplicaciones que lo utilizan.</p>	NA
Hoja de costos	<p>Aplicación web</p> <p>que permite gestionar los costos de las referencias para el cálculo del</p> <p>precio óptimo en la compañía.</p>	Costos
Macro de precios	<p>Macro en excel</p> <p>que permite formular los precios de las diferentes listas de la compañía y</p> <p>generar conector hacia el ERP para subir el precio.</p>	Contraloría



<p>Paquete completo</p>	<p>Software que permite automatizar la generación de documentos para el modelo de negocio</p> <p>'Paquete completo' en el cual la empresa terceriza la fabricación de prendas</p> <p>100% pero para garantizar el cumplimiento de diseños le vende al taller</p> <p>contratado todos los materiales necesarios para poder producir las unidades</p> <p>requeridas.</p>	<p>Producción, Planeación.</p>
<p>Conciliaciones</p>	<p>Aplicación</p> <p>desarrollada para conciliar los documentos de recibo de caja con el reporte</p> <p>de pago entregado por las franquicias y pasarelas de pago que recaudan el</p> <p>dinero de la compañía.</p>	<p>Contabilidad</p>



<p>Integrador</p> <p>Sorter</p>	<p>Aplicación de</p> <p>integración que entrega la interfaz de salida del WMS al software que</p> <p>administra el Sorter para que realice la distribución e igualmente, lee el</p> <p>archivo resultante de la transacción para informar al WMS las unidades a</p> <p>empacar.</p>	<p>CEDI</p>
<p>ASN</p>	<p>Aplicación web</p> <p>que permite a talleres y tiendas enviar avisos de despacho con el objetivo de</p> <p>que el WMS pueda realizar planeación de bahías de recepción y personal de</p> <p>recepción.</p>	<p>CEDI, Comercial, Producción.</p>



<p>Tracking ecommerce</p>	<p>Aplicación web que permite a Cedi y servicio al cliente hacer seguimiento a un pedido en particular para ante una llamada de un cliente tener información suficiente para dar respuesta. Integra de punta a punta el resultado de todos los sistemas que intervienen en el pedido ecommerce.</p>	<p>eCommerce, Servicio al cliente, CEDI.</p>
<p>Sigiti</p>	<p>Aplicación de gestión de los activos fijos tecnológicos de la compañía.</p>	<p>TI</p>
<p>Comité de iniciativas</p>	<p>Aplicación web que permite gestionar la priorización de iniciativas de las diferentes áreas en torno a nuevos desarrollos o solicitudes de intervención en automatización de procesos.</p>	<p>PMC, PMO, TI.</p>



Shooting	Aplicación para la gestión de fotografías de diferentes medios con gestión de entrega de referencias y control de los tipos de fotos necesarias y tomadas.	Mercadeo, CEDI, CALIDAD, Oficina técnica, Diseño.
Maestros web	Aplicación que permite de manera dinámica generar maestros para diferentes ámbitos que puede ser creado y utilizado en otros aplicativos mediante llamado a servicios.	TI
Photo Infinity	Módulo de gestión de fotografías para ecommerce que automáticamente asigna fotografías a las referencias creadas y las asigna en la plataforma ecommerce.	eCommerce.



Photo Finder	Aplicación de escritorio que permite encontrar fotografías por diferentes criterios en los repositorios de las fotografías del área de Shooting.	Mercadeo, eCommerce.
WMS	Sistema de gestión de bodegas de almacenamiento.	CEDI
WA	Sistema de gestión de la distribución de producto a los diferentes canales de venta y MRP (planificación de materiales) para la producción de la compañía.	Demanda y programación de la producción
Siesa Enterprise	ERP central de la compañía que contiene los módulos: Financiero, Comercial, POS, Producción.	Todas
Siesa POS	Sistema de punto de venta distribuido y sincronizado con POS Central de Siesa	Comercial



PLM	Product LifeCicle Managment -> Sistema gestor del ciclo de vida del producto. Aplicación para la creación de colecciones y productos con toda su información necesaria para la producción de la misma. Módulo de compras y distribución de carga entre jerarquía de producto y marca.	Colecciones, Oficina técnica, Patronaje.
Kactus	Sistema para la administración de los recursos humanos y talento humano	Recursos humanos
DocManager	Sistema de gestión de contratos de la compañía	Jurídico - Todas las áreas usuarias.
MFP	Sistema para la planificación financiera	Planeación financiera
CRM - Dynamics	Sistema de gestión de la relación con los clientes.	Fidelización, Mercadeo.
Zendesk	Software no propietario alojado en la nube para la gestión de tickets de clientes.	Servicio al cliente.
Credi10	Sistema para la gestión de la vertical de negocio de crédito de consumo.	Créditos y cartera



Binaps - Gestión documental	Aplicación de gestión de los documentos de la compañía.	PMC
IBM App Connect	Sistema de integración de aplicaciones.	NA
Oracle AP	Sistema de aprovisionamiento para la distribución de mercancía	Demanda
Importación de facturas	Aplicación de escritorio utilizado para importar información de facturas de una empresa en otra base de datos como una entrada y requisiciones para despacho de la mercancía.	Contabilidad
Gestor de pedidos	Aplicación web para la gestión centralizada de pedidos de venta y pedidos de distribución de mercancía.	CEDI, Comercial, Demanda.
Alianzas comerciales	Aplicación para el agrupamiento de clientes que en una alianza comercial permite la generación de campañas comerciales.	Comercial, Fidelización.