

**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

**FACULTAD DE INGENIERÍA**



# **City of Dallas Big Data Platform**

## **INFORME DE ACTIVIDADES PROFESIONALES**

Que para obtener el título de

**Ingeniero en Computación**

**P R E S E N T A**

Ernesto Ramírez Orihuela

**ASESOR DE INFORME**

Ing. Guadalupe Lizeth Parrales Romay



Ciudad Universitaria, Cd. Mx., 2021



## Contenido

Introducción	5
Objetivo	7
Capítulo 1 – Marco Teórico	8
1.1 Ingeniería de Software	8
1.1.1 Modelos Tradicionales	8
1.1.2 Métodos Modernos	12
1.1.3 Modelos emergentes y Desarrollo Ágil.	16
1.2 Administración de Proyectos	18
1.2.1 Metodología de Proyectos	22
1.3 Big Data (Grandes cantidades de datos)	24
Capítulo 2 - Organigrama	34
2.1 GBS AppOps	34
Capítulo 3 – Antecedentes y Descripción de Proyectos	39
3.1 Subject Matter Expert	39
3.2 Competency Lead	40
3.2.1 Proyectos como Competency Lead	44
3.2.1.1 Actividades con Herramientas de pruebas	44
3.2.1.2 Herramientas de Builds	46
3.2.1.3 Herramientas de CLM	46
3.3 Solution Designer	49
3.3.1 Proyectos con Herramientas de Pruebas	54
3.3.2 Proyectos con Herramientas de CLM	56
Capítulo 4 - Contexto de la Participación Profesional	58
4.1 Delivery Manager	58
Capítulo 5 - Análisis y metodología empleada	60
5.1 Despliegue	60
5.2 Soporte	63
Capítulo 6 - City of Dallas Big Data Platform - Participación profesional	66
6.1 Inicios como Delivery Manager e incorporación al proyecto	66
6.2 Herramientas	67
6.3 Inicio de Actividades	68

6.4 Instalación del Ambiente de Desarrollo	70
6.5 Instalación del Ambiente de Producción	76
6.6 Transición a Soporte	84
6.7 Migración a HDP	90
6.8 Actualización a HDP 3.x	97
Capítulo 7 - Resultados.	100
Conclusiones	103
Glosario	105
Referencias y Bibliografía	108
Anexos	112
<b>Implicaciones del Big Data</b>	<b>112</b>

## Índice de Figuras

Figura 1 Modelo en Espiral.....	12
Figura 2 Rational Unified Process.....	15
Figura 3 Triángulo de Administración de Proyectos.....	22
Figura 4 Información en Big Data.....	27
Figura 5 Map Reduce.....	31
Figura 6 Divisiones de la organización.....	34
Figura 7 Organigrama de AppOps.....	36
Figura 8 Servicios.....	38
Figura 9 Actividades de Competency Lead.....	41
Figura 10 Arquitectura típica de un ambiente de pruebas con herramientas.....	45
Figura 11 Arquitectura típica de un ambiente CLM distribuido.....	48
Figura 12 Flujo de soluciones.....	53
Figura 13 Equipos a cargo del soporte para los ambientes de City of Dallas.....	71
Figura 14 Ambiente interno para pruebas.....	75
Figura 15 Arquitectura del ambiente de producción.....	83
Figura 16 Solicitudes.....	88
Figura 17 Arquitectura general del ambiente de desarrollo a nivel de componentes.....	94
Figura 18 Arquitectura general del ambiente de producción a nivel de componentes.....	96
Figura 19 Información generada en el mundo.....	113

## Introducción

El siguiente reporte expone la experiencia referente a mis actividades como especialista de tecnologías de la información en una de las empresas más grandes de esa industria nivel global. Esta empresa internacional proveedora de servicios de tecnología de la información (en adelante referenciada como la organización) es una multinacional de origen estadounidense, ha existido desde hace más de un siglo y ha sido varias veces a través de la historia una de las pioneras en desarrollos tecnológicos.

He trabajado en proyectos de tecnologías de la información desde hace más de 14 años. Durante los cuales he desempeñado varios roles relacionados a la industria. Durante estos años he trabajado en otra empresa internacional de origen estadounidense, en dos empresas nacionales y desde hace algunos años en la empresa donde administré el proyecto mencionado en este informe.

Mi participación en la industria comenzó haciendo aseguramiento de calidad a portales web, principalmente para sitios bancarios y de aseguradoras. Posteriormente tomé algunos roles de liderazgo y me comencé a involucrar en automatización, procesos y herramientas especializadas de pruebas.

Debido a la experiencia que adquirí fui elegido para participar en proyectos de consultoría en México y Sudamérica, principalmente orientados a diseñar y a ejecutar pruebas de desempeño en sitios web, pero también brindando entrenamiento a diversas instituciones.

Además de conocer las herramientas especializadas tuve capacitación en metodologías como CMMi, ITIL y RUP. De esta manera, adicional a conocer la funcionalidad del software también adquirí experiencia en buenas prácticas y procesos personalizados para clientes.

En la compañía donde administré el proyecto relacionado a este informe he trabajado por más de ocho años, ingresando en el mes de abril del 2013. Desde entonces he participado en varios proyectos a nivel global. Mi paso hacia esta organización fue a partir del conocimiento y las certificaciones sobre herramientas especializadas Rational® de la misma empresa.

En mis actividades durante mi trayectoria laboral he aplicado los conocimientos que adquirí al estudiar en la Facultad de Ingeniería de la UNAM, como son fuerte ciencia básica y los conocimientos técnicos propios de ingeniería en computación. Han sido más que suficientes para enfrentar los retos técnicos que me he encontrado en los proyectos laborales. Por otro lado, el fondo humanista también me dio la flexibilidad suficiente para estar abierto a nuevos enfoques, paradigmas y metodologías.

El haber tenido una formación analítica, científica y técnica me ha dado una base sólida para asumir diversos retos técnicos y tener una apertura total a adquirir nuevos conocimientos y desarrollar habilidades y competencias. Así mismo asumir nuevos retos en el desempeño de mis labores diarias de forma constante.

La industria de tecnologías de la información evoluciona constantemente y nos motiva a adquirir de manera continua educación para enfrentar nuevos desafíos y encontrar soluciones día a día. La empresa donde trabajo actualmente y las organizaciones anteriores donde he laborado han sido de un valor realmente notorio durante toda mi vida laboral, pues me han brindado diversas oportunidades para crecer como especialista en TI y como persona.

En cada organización donde he trabajado he adquirido nuevos conocimientos y he desarrollado nuevas habilidades, tanto técnicas como de liderazgo y relaciones interpersonales.

Las llamadas habilidades suaves de comunicación y de liderazgo son importantes en un mundo tan globalizado y comunicado como el de esta época. Sin embargo, los cimientos en esta industria son profundamente técnicos y analíticos.

Tengo confianza en que la formación adquirida en la Universidad me ayudará en los proyectos futuros como ha sido hasta ahora.

Uno de los últimos proyectos en los que he participado y ha significado algunos retos nuevos, pero también experiencias enriquecedoras y muy gratas a diversos niveles; se relaciona con uno de los paradigmas recientes en las tecnologías de la información, el Big Data.

En este documento presentaré dicho proyecto manteniendo la siguiente estructura:

En el capítulo 1 se presenta el marco teórico que sustenta la administración de proyectos con relación a la ingeniería de software y despliegue de aplicaciones. Además de incluir un contexto general del Big Data.

En el capítulo 2 se describe la participación de mi equipo dentro de la organización. Así mismo se presenta el tipo de proyectos, actividades y servicios en los que se enfoca AppOps. Aunque AppOps se considera un grupo “mediano” dentro de la organización a nivel global, podría ser por sí mismo una empresa.

Durante el capítulo se describe cuál ha sido mi participación en los diferentes proyectos en los que he trabajado dentro de la compañía, así como los roles que he desempeñado durante los más de 8 años que he trabajado en esta organización.

Luego en el capítulo 4 se explica el proyecto de City of Dallas, tema principal en el cual se centra este informe. Se presentan detalles desde mi incorporación al proyecto, las actividades y roles que desempeñé, así como los diferentes retos que tuve que afrontar.

Finalmente, durante este mismo capítulo se mencionan los resultados del proyecto de City of Dallas. Tanto para el cliente como para mi equipo de trabajo y para mí persona.

De esta descripción, se concluye que este fue un proyecto lleno de retos, pero también enriquecedor en cuanto a aprendizaje y experiencia.

## Objetivo

En este trabajo presentaré mi experiencia laboral. Como se ha mencionado, he participado en varios proyectos. Sin embargo, me enfocaré a detalle en uno en particular, el de City of Dallas. Esto por tratarse del más complejo, retador y extenso hasta el momento. Este proyecto en particular demandó atención especial de mi parte porque fui el responsable principal, tanto para el cliente como para la organización. Desde la parte de liderazgo hasta la de finanzas, pasando por todos los procedimientos internos, de seguridad y comerciales.

Describiré desde mi participación inicial, los retos al enfrentar diversos cambios, la administración y el cierre del proyecto. Finalmente presentaré los resultados y conclusiones sobre las actividades que desempeñé en el proyecto de Big Data para City of Dallas.

# Capítulo 1 – Marco Teórico

## 1.1 Ingeniería de Software

Desde que la historia moderna de la computación comenzó se han creado diversos paradigmas para el desarrollo de software. Los diferentes autores suelen clasificar y dividir a estos paradigmas entre modelos tradicionales y modernos.

A continuación, se mencionan algunos de los modelos más conocidos de ambas categorías.

### 1.1.1 Modelos Tradicionales

Desde hace algunas décadas se propusieron y documentaron diversos modelos para el desarrollo de software. Han estado tanto tiempo en la industria de TI que son bastante conocidos por todos los profesionales que trabajan en la misma. A continuación, se describen algunos de estos modelos.

#### **Modelo en Cascada**

El modelo en cascada es probablemente el más antiguo de todos, también es llamado como el modelo ciclo de vida de software clásico. Muchas organizaciones lo utilizaron en el pasado, pero hoy en día es poco común. El nombre deriva de que las tareas ocurren de forma secuencial, una tras otra. Siendo las salidas de una fase las entradas de la siguiente.

Entre los aspectos positivos de este modelo se pueden mencionar:

- Los requerimientos deben especificarse en los primeros pasos.
- Cuatro actividades deben ser completadas antes de que el software pueda ser empaquetado y liberado (requerimientos, diseño, implementación y pruebas).
- Las salidas de cada fase son las entradas de la siguiente en forma secuencial.
- El proyecto debe ser rastreado conforme se mueve entre las distintas fases de forma secuencial.



Por la gran cantidad de documentos generados con los requerimientos, diseño y pruebas, el modelo en cascada también es llamado por algunos autores como método conducido por documentos.

La más importante contribución de este modelo a la ingeniería de software fue brindar a los especialistas una manera de organizar y documentar las actividades de desarrollo. Por otro lado, ha caído en desuso desde hace mucho tiempo. Esto por ser complejo hacer cambios sobre la marcha en este modelo, el cambio es algo que sucede mucho cuando se da interacción cercana con los clientes. Así que, identificar los cambios de manera tardía causa sobrecostos y toma bastante tiempo rehacer lo que debe actualizarse [1][3].

### **Modelo Incremental**

Este modelo podría ser visto como una modificación al modelo en cascada. Conforme los proyectos de software crecieron en cuanto a tamaño y complejidad, surgieron nuevas necesidades. Se identificó que era más práctico el desarrollo de software si se dividía en componentes más sencillos, trabajando así de manera incremental e iterativa. Al principio cada componente se desarrolla tomando como base el modelo en cascada y al final se hace una integración y pruebas de todo el sistema. Este modelo es más flexible y tiene mitigación de riesgos. Si un componente tiene retrasos o problemas los demás pueden continuar en desarrollo, a menos que los problemas afecten a todo el sistema.

Además, en ciertos desarrollos la perspectiva es desarrollar primero los componentes principales que contienen la mayor parte de la funcionalidad requerida. Así el primer incremento de hecho podría entregarse a los usuarios y al cliente como la primera liberación. La funcionalidad adicional y características complementarias podrían irse completando y liberando en entregas subsecuentes hasta terminar. Hay dos opciones en el modelo incremental, una para producir una liberación general y otra para hacer múltiples liberaciones. El número de liberaciones dependerá de la naturaleza y metas de cada proyecto. Ambos modelos incrementales se basan en la estrategia de divide y vencerás, donde un problema grande y complejo se descompone en varias partes. La dificultad de este modelo radica en que esas partes puedan estar estrechamente conectadas, haciendo la descomposición en partes algo que puede ser complicado [1].

### **Modelo en espiral.**

Este modelo es un método evolutivo para el desarrollo de software, fue propuesto por Barry Boehm (1988) en un momento donde había preocupaciones sobre el modelo en cascada que era conducido por mucha documentación. Un aspecto importante de este modelo es el énfasis que se hace en la reducción temprana de riesgos. De esta forma este modelo puede considerarse como un método conducido por riesgos.

En este modelo se pueden considerar dos características distintivas:

- Un enfoque cíclico para el crecimiento incremental del sistema y de su grado de implementación. A la par va disminuyendo el riesgo.
- Un conjunto de puntos de referencia para asegurar el compromiso de los participantes en el desarrollo del software.

En este paradigma el software se desarrolla con entregas de tipo evolutivo. De esta manera durante las primeras liberaciones normalmente lo que se puede entregar son un modelo o prototipo. En las iteraciones subsecuentes se producen versiones cada vez más completas del sistema que se está desarrollando.

Al inicio del proceso evolutivo el equipo realiza actividades propias de comunicación, definición de objetivos y planeación. Se parte del centro y se va avanzando en sentido horario hasta completar una revolución durante la cual se considera el riesgo.

Usualmente durante el primer circuito se genera una especificación del producto a ser construido. Así, conforme se van completando vueltas se construye un prototipo, se agregan características y se obtienen versiones cada vez más sofisticadas del software. En cada paso por la sección de planeación se van realizando los ajustes correspondientes en el plan del proyecto. El costo y la planeación se ajustan de acuerdo a la retroalimentación del cliente después de cada entrega.

A diferencia de algunos otros modelos que finalizan con la entrega, este modelo puede adaptarse para ser aplicado durante toda la vida del software. Es decir, se puede utilizar en proyectos subsecuentes.

Algunos autores mencionan 4 o 5 cuadrantes, como se muestra en la figura 1. Por esos cuadrantes pasa la espiral en cada circuito, de forma general trabajando con 4 cuadrantes podemos mencionar:

- Identificar objetivos, alternativas o restricciones para cada ciclo de la espiral.
- Evaluar las alternativas relativas a los objetivos y restricciones. Durante este paso muchos de los riesgos son identificados.
- Dependiendo de la cantidad y tipo de riesgos, se desarrolla un prototipo, una evaluación más detallada, un desarrollo evolutivo o algunas otras acciones para reducir el riesgo de conseguir los objetivos identificados. Por otro lado, si los riesgos se redujeron de manera significativa, las siguientes tareas podrían relacionarse a requerimientos, diseño o codificación.
- Validar si los objetivos se alcanzaron y planear el siguiente ciclo.

Una parte integral de cada ciclo es la revisión de las actividades y productos completados por parte de los principales interesados en el proyecto. El principal objetivo de la revisión es asegurarse que todas las partes están fuertemente comprometidas en el proyecto y acordar la metodología para la siguiente fase del proyecto.

Sin embargo, al igual que otros modelos, el espiral tiene sus particularidades y áreas de oportunidad. No es fácil convencer a los clientes de que el modelo evolutivo es controlable. Irónicamente uno de sus riesgos es la dependencia de la experiencia en la evaluación de riesgos y se basa en ella para lograr el éxito. No todos los ingenieros de software están entrenados o tienen la suficiente experiencia para identificar y analizar los riesgos [1][2].

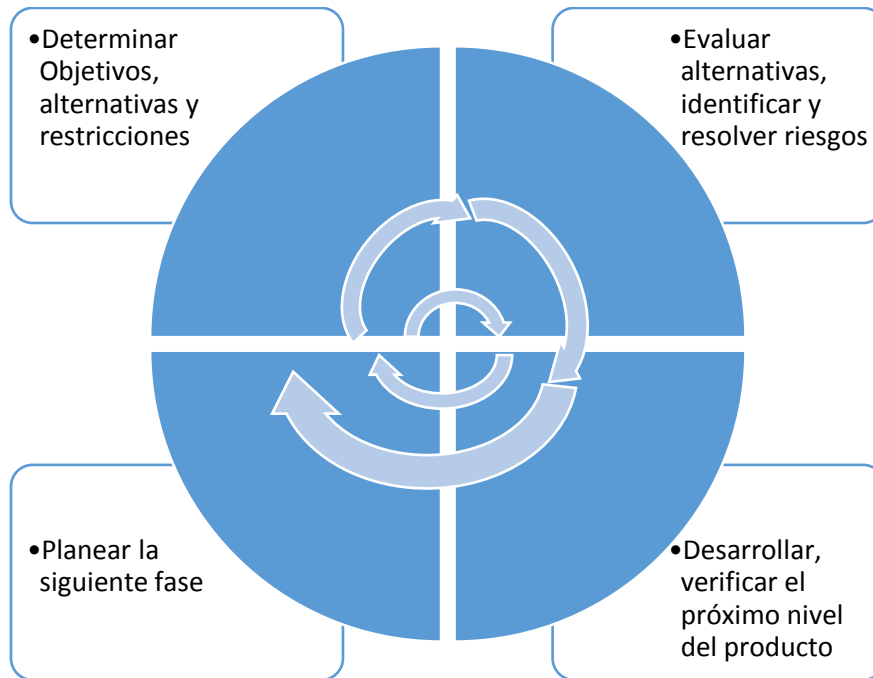


Figura 1 Modelo en Espiral

## 1.1.2 Métodos Modernos

A continuación, se describe uno de los métodos modernos que se ha utilizado en la industria y que retoma buenas prácticas de los métodos tradicionales en conjunto con herramientas específicas para ejecutar las actividades.

### **Rational Unified Process.**

El Rational Unified Process también conocido como RUP, es un marco de referencia, no solo un proceso. Fue desarrollado por Rational Software Corporation, después adquirido por IBM. Sus orígenes se remontan a 1987 en el Objectory Process y el Rational Objectory Process de 1997, así como el Lenguaje de Modelado Unificado (UML). En muchas maneras RUP incorpora elementos de los modelos de proceso genéricos como el incremental e iterativo y el espiral. Además, que se basa en buenas prácticas de desarrollo de software.

Debido a que toma bases de métodos tradicionales y buenas prácticas de la industria (menos formales), algunos autores lo consideran como un modelo híbrido.

Este marco de referencia es conducido por 3 grandes conceptos:

1. Requerimientos conducidos por casos de uso
2. Centrado en la arquitectura
3. Iterativo e incremental

El concepto de caso de uso se remonta a Ivar Jacobson y su trabajo temprano en grandes sistemas de comunicaciones en Ericsson, fue su primera contribución al UML (Lenguaje de Modelado Unificado).

En RUP el caso de uso es la manera en que se registra un requerimiento funcional de la aplicación para dar un resultado de valor al usuario.

Las características de un caso de uso son las siguientes:

1. Están relacionados por comportamiento, esto significa que la interacción entre ellos como grupo constituye una unidad auto contenida.
2. El caso de uso debe ser iniciado por un actor.
3. El resultado de valor debe ser medible, es decir, el caso de uso debe alcanzar una meta de negocio en particular.
4. El caso de uso debe dejar al sistema en un estado estable, no puede dejar actividades a “medias”.

Los casos de uso han sido utilizados principalmente para el levantamiento de requerimientos, pero pueden ser usados para describir cualquier interacción entre el sistema de software y cualquier ente externo; ya sea un usuario u otro sistema.

La arquitectura juega un rol muy importante dentro de RUP, de hecho, las primeras iteraciones y casos de uso en desarrollarse están orientados a construir una arquitectura sólida y estable. Está describe tanto los aspectos estáticos como dinámicos del sistema, resaltando los más importantes y dejando un poco de lado - temporalmente - los de menor impacto.

RUP es también iterativo e incremental, puesto que también motiva a trabajar los grandes desarrollos de software a través de pequeñas partes o incrementos. La primera iteración se

suele enfocar en la arquitectura y en trabajar los casos de uso de mayor impacto. La segunda iteración administrará los riesgos más importantes en el incremento elegido. Las iteraciones sucesivas se enfocan en ir construyendo el sistema con base en los resultados de las iteraciones previas.

RUP se desarrolla a través de 4 fases:

- Inicio. Durante esta fase se definen los casos de negocio para el sistema, se identifican las entidades externas y se definen las interacciones.
- Elaboración. Se desarrolla la comprensión del sistema y se trabaja para definir la arquitectura. Desarrollar plan de proyecto e identificar riesgos. Al completar esta fase se deben tener identificados casi todos los requerimientos y contar con el modelo general de casos de uso del sistema, además del plan de desarrollo de software.
- Construcción. Principalmente se realizan actividades de diseño, codificación y pruebas, también de integración. Al término de esta fase se debe de tener software que pueda ser entregado a los usuarios junto con la documentación asociada.
- Transición. Principalmente se trabaja en mover el software desde el ambiente de desarrollo hacia el entorno real del usuario y realizar el despliegue, también dar entrenamiento en el nuevo software en caso de ser necesario.

Al ser de tipo incremental, el desarrollo del software puede pasar varias veces a través de cada fase. Sin embargo, el grado de actividades tales como requerimientos, codificación, pruebas o despliegue varía en cada una de ellas.

Este marco de referencia se trató de comercializar de tal forma que Rational desarrolló varios entrenamientos y herramientas de software específicamente para adaptarse a RUP. Durante algunos años fue toda una *suite* bastante robusta.

En la figura 2 se representan de manera gráfica las fases y las actividades, se puede observar el grado de participación de cada actividad en las diferentes fases.

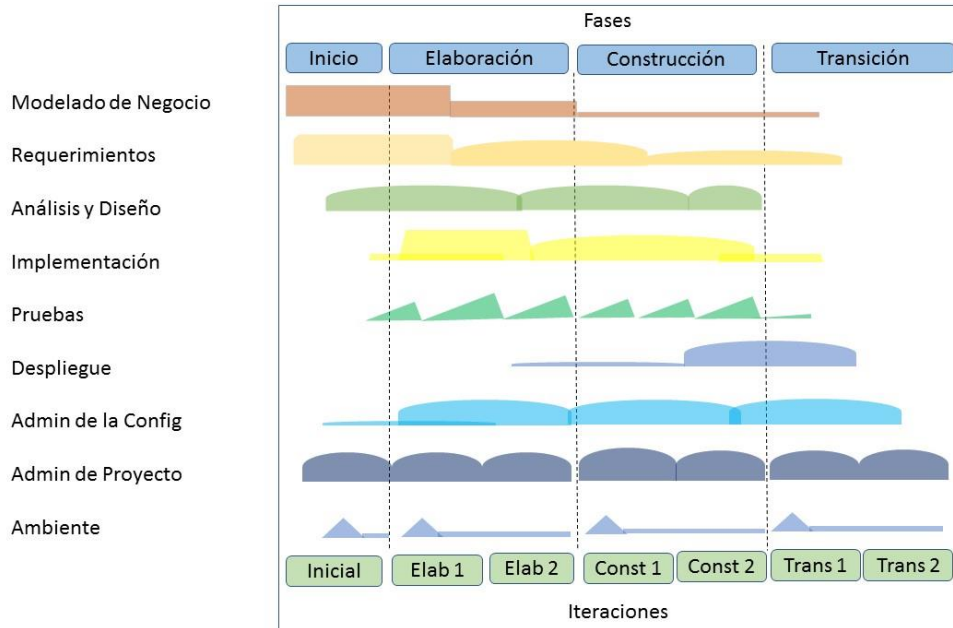


Figura 2 Rational Unified Process

Adicionalmente han surgido otros marcos de referencia que ayudan al desarrollo de software incorporando las mejores prácticas de la industria. Así podemos mencionar al CMMi enfocado en identificar e incrementar la madurez de las organizaciones en cuanto al desarrollo de software, los estándares ISO que se utilizan para seguridad y calidad, ITIL que es la gran referencia en cuanto a servicios, entre otros. Cada uno de estos marcos de referencia es muy extenso y son motivo de gran estudio por los especialistas de TI.

### 1.1.3 Modelos emergentes y Desarrollo Ágil.

En 2001 un grupo de 16 afamados desarrolladores de software, consultores y escritores firmó lo que se conoció como el manifiesto ágil (<https://agilemanifesto.org>). En este acuerdo se plasmaban muchas ideas basadas en la experiencia, de forma general se puede mencionar que las ideas establecen lo siguiente:

- Anteponer los individuos y sus interacciones sobre los procesos y las herramientas.
- El software que puede usarse por el usuario es prioritario sobre la documentación detallada.
- Colaboración con cliente y usuarios más que lo establecido en un rígido contrato.
- Adaptarse al cambio, no apegarse tanto a un largo plan establecido en un inicio.

Aunque no hay una definición absoluta y estricta de lo que significa trabajar con un método ágil, existen muchas características comunes compartidas por los métodos ágiles:

- Liberaciones cortas e iteraciones. Se trata de dividir el trabajo en pequeñas piezas y liberar el software al cliente tan frecuentemente como sea posible.
- Diseño incremental. No intentar completar el diseño si no se tiene la información suficiente, mejorar el diseño existente conforme se obtiene más conocimiento de las necesidades del usuario.
- Involucramiento del usuario. Más que intentar producir estándares completos, rígidos y formales, pedir retroalimentación constante de los usuarios involucrados.
- Mínima documentación. Elaborar solo la cantidad necesaria de documentación, siendo el código fuente una parte importante de la misma.
- Comunicación menos formal. Mantener comunicación constante pero no necesariamente a través de documentos formales. La idea es que la gente se comunica mejor de manera informal. Esto funciona principalmente durante el entendimiento de las necesidades de los usuarios.
- Cambio. Asumir que los requerimientos y el ambiente van a cambiar, tratar de encontrar maneras adecuadas de adaptarse a ese hecho.



Entre algunas de las metodologías ágiles podemos citar:

- Extreme Programming (XP). Una de las primeras metodologías ágiles, utilizado por primera vez en una de las mayores empresas automotrices de Estados Unidos. Se utilizan equipos pequeños trabajando en el mismo lugar para motivar la comunicación. En este modelo se recomienda generar solo la documentación estrictamente necesaria, con el código en sí mismo y las pruebas unitarias como documentación [3].
- El proceso unificado para ágil (AUP) de Scott Ambler. Aunque este proceso no especifica técnicas particulares para todas las fases, usualmente es considerado como más pesado por el hecho de que hace más énfasis en obtener los requerimientos al principio, durante la fase de inicio. Requiere además definir la arquitectura y el diseño en sus fases tempranas, al igual como una de sus instancias más populares, el RUP. De hecho, es como una adaptación de RUP para ágil, pero restringiendo los productos de trabajo y facilitando las iteraciones [3].
- Scrum. Esta es otra metodología ágil que ha mostrado buenos resultados. Es más ligera que la programación extrema (XP) aunque retoma algunas de sus buenas prácticas. Fue introducida con base en su exitosa aplicación en la industria de la manufactura y toma varios de sus términos del juego del rugby. Consiste de roles, artefactos, eventos y reglas. Es un modelo iterativo basado en iteraciones cortas llamadas “sprints”, las cuales son las unidades básicas de desarrollo en cuanto a tiempo. El objetivo es que al final de cada sprint se tenga una versión potencialmente entregable del software. Los roles son bastante conocidos en la industria, el dueño del producto (product owner), el equipo de desarrollo y el scrum master. Sus prácticas se encuentran entre las más usadas en la actualidad [3].
- El método Kanban. Si bien no es una metodología propia de desarrollo de software, se ha utilizado en los últimos años como marco de trabajo. Sus orígenes se remontan a Japón y derivó del sistema de manufactura de una gran empresa automotriz de ese país. El concepto central en este tipo de manufactura es minimizar el trabajo que se encuentra en progreso haciéndolo muy visible a todo el equipo, se logra al “arrastrar” los demás componentes involucrados en el proceso. Esto se hace a través de tarjetas, de esta manera se especifica qué partes son necesarias para completar la tarea. De la

aplicación de manufactura fue llevado al desarrollo de software para mejorar la ejecución de actividades haciendo pequeñas modificaciones [3].

Cada una de las metodologías mencionadas ha sido útil para varios proyectos, con sus particularidades y prácticas propias, pero siempre tratando de apearse a los principios del manifiesto ágil.

## 1.2 Administración de Proyectos

Un proyecto es un esfuerzo temporal llevado a cabo para crear un producto o servicio. Es temporal porque tiene un comienzo y fin determinados, por esta razón lo normal es tener un alcance y recursos definidos. Los proyectos suelen ser únicos en cuanto a que la gente trabajando en ellos lo hace por tiempo determinado y ejecutan labores específicas.

Algunos ejemplos de proyectos pueden ser la construcción de un edificio, la expansión de ventas en una zona geográfica, el desarrollo de software para una mejora o el despliegue de infraestructura y aplicaciones para lograr objetivos de negocio de una empresa u organización. Todos estos esfuerzos deben ser administrados para ser completados en tiempo y de acuerdo al presupuesto definido.

La administración de proyectos es la aplicación de habilidades, herramientas, conocimientos y técnicas para lograr cumplir con los requerimientos de un proyecto. Estas actividades han sido con frecuencia realizadas de forma empírica en el pasado. Sin embargo, comenzaron a considerarse como una profesión, rol y área de conocimiento a partir de la segunda mitad del siglo 20.

En cuanto a computación e informática, la administración de proyectos de software y de todas las actividades para implementar soluciones de tecnologías de la información es un área esencial. Independientemente del paradigma o modelo de desarrollo con el que se estén realizando las actividades. Está incluida en los trabajos y libros de la mayoría de los autores de la ingeniería de software como algo indispensable. Administrar bien las actividades no

siempre funciona satisfactoriamente, pero no hacerlo simplemente conduciría a fallar en casi todos los casos. La mala gestión puede conducir a fallar en fechas acordadas, exceso de costos, malas entregas o incluso el fracaso completo de un proyecto. Desde el punto de vista de las organizaciones esto puede llevar al fracaso de los programas y metas establecidas.

Los administradores de proyectos trabajan en la planificación y en el seguimiento de las actividades de desarrollo del software o de la implementación y despliegue de soluciones de software, infraestructura u otras actividades relacionadas a tecnologías de la información.

Administrar proyectos en la industria de tecnologías de la información es algo que siempre está presente, esto debido a que las actividades de ingeniería de software y de soluciones están estrechamente relacionadas a restricciones de tiempo y presupuesto. La labor del administrador de proyectos es asegurarse de completar las actividades en tiempo y sin sobrecostos, además de apegarse a las políticas y cumplir con los objetivos definidos.

La administración de proyectos de software es un conjunto de actividades que suelen cubrir todas las áreas y fases en la ingeniería de software, de una u otra manera en todos los paradigmas de desarrollo de software y despliegue de aplicaciones. Normalmente empieza incluso antes de cualquier labor técnica y pasa por el modelado, construcción y despliegue del software. También cubre otras actividades como la implementación de infraestructura y despliegue de software de “caja”. Entre las actividades se incluyen cuestiones de métricas, estimaciones, planeación, análisis de riesgos, seguimiento y control; así como manejo de personal especializado.

Los administradores de tecnologías de la información suelen hacer actividades muy similares a otros administradores; aunque la ingeniería de software y otras actividades de TI involucran ciertas diferencias y algunas particularidades.

Algunos ejemplos que se pueden citar son:

- *El producto normalmente no es algo tangible.* A diferencia de un edificio, una máquina o un barco, el avance del software no se puede ver a simple vista por todos.
- *Los procesos de software no están completamente estandarizados.* Es diferente a ciertas ramas de procesos industriales donde hay normas muy estrictas y conocidas. La ingeniería de software es más flexible y hay muchos paradigmas y distintas buenas prácticas de distintas empresas.
- *Con frecuencia los proyectos grandes no se pueden repetir.* Los requerimientos de software para cada cliente son muchas veces muy específicos para cada caso de negocio y la reutilización de componentes suele darse solo dentro del mismo cliente.
- *Es común que lo que se está construyendo o implementando sea algo nuevo sobre lo que poca gente tiene experiencia previa.* De esta manera son frecuentes los retos, cambios y el aprendizaje en paralelo conforme se van completando las actividades.

De lo anterior se puede notar que hay muchas dependencias y factores que pueden causar impacto en un proyecto de administración de software y tecnologías de la información. Por todo lo mencionado, no debería causar asombro que los proyectos tengan retrasos, que necesiten cambios de presupuesto o que tengan cambios sobre la marcha. Es parte de los retos de trabajar frecuentemente con nuevas tecnologías y prácticas. Sin embargo, es satisfactorio hacer notar que, a pesar de las dificultades, muchos proyectos de software y TI son entregados en tiempo y de acuerdo al presupuesto establecido.

De manera general, los procesos incluidos en casi cualquier administración de proyectos de software o TI se pueden incluir en 5 grupos:

- Iniciar
- Planear
- Ejecutar
- Monitoreo y Control
- Cierre

Por otro lado, la administración de proyectos de software y TI deriva en las siguientes áreas:

- *Integración.* Hacer que varios procesos interactúen en conjunto.
- *Alcance.* Identificar claramente las actividades que se incluyen en el proyecto.
- *Tiempo.* Calendarización y agenda de actividades
- *Costos.* La administración del presupuesto asignado a las actividades que se deben completar.
- *Delegar.* Ser capaz de asignar actividades a otros miembros del equipo para ser más eficientes.
- *Calidad.* Asegurar que los productos y/o servicios cumplen con los estándares de calidad establecidos.
- *Recursos Humanos.* Identificar los roles y actividades para los distintos miembros del equipo de trabajo.
- *Comunicación.* Identificar como se llevará a cabo la comunicación con el equipo de trabajo y los interesados, se incluyen la frecuencia y los canales de comunicación.
- *Riesgos.* Definir como se identificarán y categorizarán los riesgos; así como la manera en que se responderá ante los riesgos.
- *Relaciones con los interesados (stakeholders).* Listar a los interesados y priorizar sus intereses, indicar como impactarían al proyecto.

Los anteriores son grupos y áreas de administración con los que trabaja el gestor de proyectos de software y TI, en casi cada proyecto de mediana o alta complejidad.

Por otro lado, el administrador de proyectos hace un balance de manera constante de estos tres factores para asegurar tener bajo control el proyecto:

- Alcance
- Tiempo
- Recursos

Esto se representa gráficamente en la figura 3.

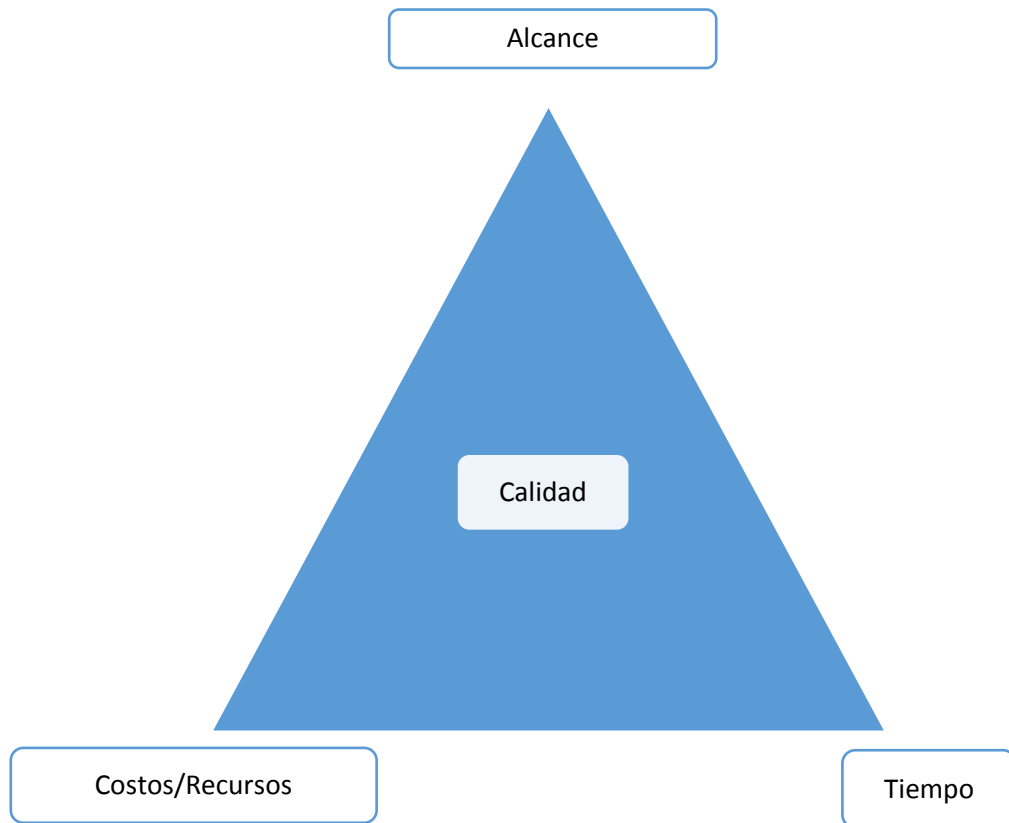


Figura 3 Triángulo de Administración de Proyectos

Es así que el administrador de proyectos de software y TI está constantemente balanceando esos 3 factores; dando seguimiento a los procesos y cubriendo las áreas de administración. Todo esto para obtener productos y soluciones de calidad.

La necesidad de incluir actividades de administración de proyectos para desarrollo de software y soluciones de TI podría no ser tan clara en un inicio. Parecería que se puede simplemente seguir procesos, metodologías y prácticas de la industria. Esta idea podría sobre todo surgir para proyectos pequeños. Sin embargo, en todos los casos es necesario determinar el número de personas que se necesitan, si es que todos conocen un proceso o prácticas en específico o qué nivel de experiencia y conocimiento tienen. Más aún, es necesario saber si el progreso del proyecto va de acuerdo con las metas del cliente, para esto hay que monitorear las actividades.

De esta manera, aun el desarrollo de pequeñas y relativamente simples aplicaciones necesita prácticas de administración de proyectos. Una de las cosas que usualmente difiere entre los distintos proyectos de desarrollo o implementación es el grado de administración que se requiere en cada caso. Algunos proyectos requerirán solo unas prácticas básicas; otros necesitarán incluir todos los procesos, áreas e incluso herramientas para conseguir los objetivos establecidos.

La administración de desarrollos de software y de implementación de aplicaciones debe adaptarse a los diferentes modelos o paradigmas de la industria, los cuales fueron mencionados con anterioridad.

Es así que en la administración de proyectos se trabaja usualmente de dos maneras.

- Modelo clásico. Basado originalmente en cascada, pero adaptado a los modelos iterativos en incrementales. Aquí se definen tareas con fechas y tiempos específicos. Además, se asignan responsables y se hace el seguimiento y monitoreo de acuerdo al plan de trabajo establecido desde un inicio. Sin embargo, se va actualizando respecto a los cambios que se van presentando durante el desarrollo del proyecto.
- Ágil. Se toman también elementos del modelo clásico pero adaptados a metodologías ágiles, principalmente en scrum, pero también SaFE, XP o incluso Kanban. De hecho, muchos administradores de proyectos toman actividades a nivel general, las plasman en un plan de proyecto tradicional y hacen un monitoreo diario a través de una llamada o reunión del tipo scrum. La realidad es que en la práctica se trata de llevar la metodología de administración formal, pero incorporando las prácticas de los modelos ágiles, esto se aplica sobre todo en proyectos de infraestructura e implementación.

Sin embargo, en proyectos propiamente de desarrollo de software si se llevan a cabo los principios ágiles de una forma mucho más apegada a las recomendaciones del manifiesto ágil (<https://agilemanifesto.org>). En estos casos los clientes también están cada vez más familiarizados a medir el esfuerzo de otras maneras. Por ejemplo, considerando el número de historias de usuario por semana o mes y basados en eso se negocian los contratos y otros acuerdos.

## 1.3 Big Data (Grandes cantidades de datos)

En los últimos años los datos son constantemente creados y tienen una tasa de crecimiento mayor. Los dispositivos móviles, las redes sociales y tecnologías de imágenes para diagnósticos médicos generan una cantidad muy grande de información ya que se crean datos de manera constante.

También, se deben considerar dispositivos y sensores que generan de forma automática información de diagnóstico que necesita almacenarse y procesarse en tiempo real. Estar al paso de este enorme flujo de datos es difícil y es aún más grande el reto de analizarlos por las grandes cantidades que se generan. En especial porque no se apegan a las estructuras tradicionales de datos, se deben identificar patrones y extraer información que pueda ser útil. Estos retos representan por otro lado oportunidades para transformar negocios, gobiernos, ciencia y la vida de las personas en general.

El Big Data es un concepto relativamente reciente en el área de las tecnologías de la información y en el mundo. De acuerdo a Erl, Khattak y Buhler, el Big Data es un campo de estudio dedicado al análisis, procesamiento y almacenamiento de grandes cantidades de datos que frecuentemente se originan a partir de fuentes diferentes. Las soluciones de Big Data usualmente son necesarias cuando el análisis tradicional de datos no es suficiente. En específico, las prácticas de Big Data son utilizadas cuando se combinan múltiples fuentes o grupos de datos que no están relacionados entre sí. En esos casos se debe procesar grandes cantidades de datos no estructurados y recolectar información que no está totalmente a la vista.

Aunque hoy aparece como una nueva disciplina, el Big Data ha estado desarrollándose por años. La administración de grandes cantidades de información ha sido un problema de mucho tiempo. Esta área ha evolucionado desde métodos de intenso trabajo hasta basarse en técnicas de actuaría utilizadas en cálculos de seguros.



Varias industrias han liderado el desarrollo de habilidades para recolectar, ordenar y explotar los datos. Se pueden mencionar:

- Compañías de tarjetas de crédito
- Empresas de telefonía celular
- Corporaciones de redes sociales

Es posible utilizar ciertos atributos para definir las características del Big Data:

- Enormes volúmenes de datos. Más que miles o millones de registros, en Big Data se pueden manejar miles de millones de renglones y millones de columnas.
- Complejidad en tipos de datos y en estructuras. En el Big Data se refleja la variedad de nuevas fuentes de datos, formatos y estructuras. Incluyendo información de la web y otros repositorios digitales.
- Velocidad en creación de nuevos datos y en crecimiento. El Big Data puede mostrar un crecimiento de datos a alta velocidad con rápida ingestión de datos y análisis casi en tiempo real.

Aunque el volumen manejado en el Big Data usualmente atrae la mayor parte de la atención, en realidad la gran variedad y la velocidad con la que crecen los datos da una mejor definición de Big Data. Algunas veces es descrito con las 3 V's de volumen, variedad y velocidad.

Debido al gran tamaño y sobre todo a la estructura, el Big Data no puede ser analizado de manera eficiente utilizando métodos tradicionales para bases de datos. Los problemas de Big Data necesitan herramientas nuevas y tecnologías capaces de almacenar, administrar e identificar los beneficios de negocio que pueden ser obtenidos de tales cantidades de información. Estas nuevas herramientas y tecnologías permiten la creación y manipulación; así como administrar los grandes grupos de datos.

El Big Data se presenta de diversas formas, incluye datos estructurados y no estructurados como la información financiera, archivos de texto y mapas genéticos. A diferencia del análisis tradicional de datos que manejaban muchas organizaciones, la mayor parte de la

información en Big Data es no estructurada o semi estructurada en su naturaleza. Se requiere de diferentes técnicas y herramientas para procesar y analizar esa información. Los ambientes de cómputo distribuido y arquitecturas de procesamiento masivo en paralelo son las opciones preferidas para procesar esos complejos grupos de datos.

Estos son unos ejemplos de tipo de información:

- *Datos estructurados*. Información que contiene tipos de datos definidos, con formato y estructura. Tal como datos de tipo transaccional, OLAP, cubos de datos, los RDBMS tradicionales, archivos CVS e incluso hojas de cálculo.
- *Datos semi estructurados*. Datos de texto con algunos patrones discernibles, lo cual hace posible ejecutar un análisis sobre ellos. Podemos mencionar por ejemplo los archivos de tipo xml.
- *Datos no estructurados*. Información que no tiene una estructura inherente, se pueden incluir documentos de texto, PDFs, imágenes, videos.

A futuro se espera que entre 80 o 90 % de los datos generados sean no estructurados. Aunque los datos son de distintos tipos, con frecuencia hay una mezcla de datos estructurados y no estructurados en la información generada hoy en día.

Como puede notarse de la figura 4, los datos no estructurados tienen cada vez una proporción más grande entre el tipo de información que se produce cada año en el mundo.



Figura 4 Información en Big Data

Los datos obtenidos de fuentes externas frecuentemente no tienen un formato o estructura que puedan ser procesados de manera directa. Para superar estos problemas de compatibilidad y preparar la información para su procesamiento ciertas adecuaciones son necesarias. Es necesario filtrar, limpiar y preparar los datos para su análisis. Desde una perspectiva de almacenamiento, una copia de los datos es guardada en su forma original y después de prepararlos deben ser guardados otra vez.

Típicamente se necesita almacenamiento en estos casos:

- Cuando se obtienen grupos de datos externos.
- Los datos son manipulados para preparar su análisis
- Los datos son procesados por medio de un ETL o son el resultado de una operación analítica.

Debido a que frecuentemente es necesario guardar múltiples copias de grupos de datos, en Big Data se han creado innovadoras estrategias de almacenamiento para reducir costos y tener sistemas altamente escalables.

### **Clústeres**

En computación un clúster es una agrupación estrecha de servidores o nodos. Estos servidores normalmente tienen el mismo tipo de hardware y están comunicados por medio de una red, de tal forma que puedan trabajar como una sola unidad. Cada nodo tiene sus propios recursos dedicados, tales como memoria, procesadores y discos duros. El clúster es capaz de ejecutar una tarea al dividirla en pequeñas piezas y distribuir la ejecución entre las diferentes computadoras que son parte del clúster [7].

### **Sistemas de archivos y sistemas de archivos distribuidos.**

Un sistema de archivos o *file system* es un método por el cual se almacenan y organizan los datos en un dispositivo de almacenamiento. El archivo es la unidad atómica usada por el sistema de archivos para guardar datos. Un sistema de archivos brinda una vista lógica de los datos almacenados en el dispositivo de almacenamiento y usualmente presenta esta vista en forma de un árbol a manera de estructura de directorios y archivos. Los sistemas operativos utilizan los sistemas de archivos para almacenar y recuperar los datos para las aplicaciones.

Cada sistema operativo da soporte a uno o más sistemas de archivos. Tal como NTFS lo hace para Windows y ext para Linux [7].

### **NoSQL**

Una base de datos *Not-Only SQL* o No solo SQL es una base de datos no relacional, altamente escalable que es tolerante a fallos. Está específicamente diseñada para albergar información no estructurada o semi estructurada. Las bases de datos NoSQL con frecuencia proveen una interfaz de consultas basadas en APIs, las cuales pueden ser llamadas desde aplicaciones. Además, estas bases de datos soportan otros lenguajes de consultas además del SQL, ya que este fue diseñado para información estructurada [7].

## **Sharding**

“*Sharding*” (fragmentación en español) es un procedimiento que consiste en particionar de manera horizontal grupos de datos grandes entre colecciones de grupos de datos más pequeñas y manejables, estas son llamadas “*shards*” (fragmentos). Los fragmentos se distribuyen entonces entre varios nodos, cada fragmento es almacenado en un nodo separado y cada nodo es solo responsable por los datos almacenados en él. Cada fragmento comparte el mismo esquema, de esta manera todos los fragmentos representan de manera colectiva un grupo de datos completo [7].

## **Replicación**

La replicación consiste en almacenar múltiples copias de grupos de datos conocidos como réplicas, esto se hace entre múltiples nodos. Esto brinda escalabilidad y disponibilidad por el hecho de que los mismos datos son replicados hacia varios nodos. La tolerancia a fallas también se logra de esta manera, puesto que la redundancia de datos asegura que estos no se pierdan si un nodo individual falla. Hay dos métodos usados para implementar la replicación:

- Amo-esclavo (*master-slave*)
- Punto a punto (*peer-to-peer*)

En la replicación amo- esclavo los nodos son acomodados en ese tipo de configuración, todos los datos son escritos en el nodo maestro. Una vez guardados los datos, son replicados a los múltiples esclavos. Todas las solicitudes de escritura del exterior incluyendo el insertar (*insert*), actualizar (*update*) y borrar (*delete*) suceden en el nodo amo. Mientras que las solicitudes de lectura pueden ser atendidas por cualquier nodo esclavo. Esta replicación es ideal para cargas de trabajo de intensa lectura más que de escritura, puesto que la demanda de lecturas puede ser administrada al escalar de manera horizontal agregando más esclavos. Por otro lado, las transacciones de escritura son consistentes al ser todas coordinadas por el nodo amo. Sin embargo, habría desventajas de desempeño si la demanda de escritura aumenta. En estos casos aun si el nodo amo falla las lecturas son posibles a través de cualquier nodo esclavo.

En la replicación punto a punto todos los nodos trabajan al mismo nivel. Es decir, no hay relaciones amo – esclavo entre los nodos. Cada nodo – conocido como peer – es igualmente

capaz de manejar lecturas y escrituras de datos. Cada escritura es copiada a todos los demás nodos. Sin embargo, en este tipo de replicación es probable que haya inconsistencia entre los nodos. Esto podría ocurrir como resultado de actualizaciones simultáneas de los mismos datos en distintos nodos. Esto se puede evitar utilizando estrategias de concurrencia optimista o pesimista.

La concurrencia pesimista es una estrategia proactiva que previene la inconsistencia. Esta utiliza bloqueos para asegurar que solo una actualización pueda ocurrir en un momento dado. Sin embargo, esto afecta la disponibilidad al permanecer el registro bloqueado mientras es actualizado.

El enfoque optimista de concurrencia por otro lado es una estrategia reactiva que no usa bloqueos. En vez de eso permite inconsistencia con el conocimiento de que la consistencia de datos sucederá eventualmente después de que las actualizaciones se propaguen [7].

### **Hadoop.**

Hadoop es un marco de trabajo (*framework*) de software abierto para almacenamiento a gran escala y procesamiento de datos que es compatible con hardware construido para procesamiento de datos. Este marco de trabajo se ha establecido como una plataforma para soluciones de Big Data de facto. Puede ser usado como un motor ETL (extraer, transformar y cargar datos) o como un motor de análisis para procesar grandes cantidades de datos estructurados, semi estructurados o no estructurados.

Desde la perspectiva de análisis, Hadoop implementa el marco de referencia de procesamiento llamado *Map Reduce*.

El paradigma de *Map Reduce* provee los medios para descomponer grandes tareas en pequeñas tareas, para correrlas en paralelo y consolidar las salidas de las tareas individuales hacia las salidas finales. Como el nombre lo implica *Map Reduce* consiste de dos partes básicas, un paso de mapeo y uno de reducción.

Map:

- Aplica una operación a una parte de datos.
- Provee una salida intermedia.

Reduce:

- Consolida las salidas intermedias desde los pasos de Map.
- Provee la salida final.

Cada paso utiliza pares de llave/valor - denotados como <llave, valor> - a manera de entrada y salida. Es conveniente pensar en estos pares como simples pares ordenados. Sin embargo, los pares pueden tomar formas bastante complejas. Por ejemplo, la llave podría ser un nombre de archivo y el valor el contenido completo del archivo.

El ejemplo más ilustrativo del *Map Reduce* es un conteo de palabras en el cual la tarea es simplemente contar el número de veces que una palabra aparece en una colección de documentos. En la práctica, el objetivo del ejercicio es establecer la importancia relativa de ciertas palabras como se ejemplifica en la figura 5.

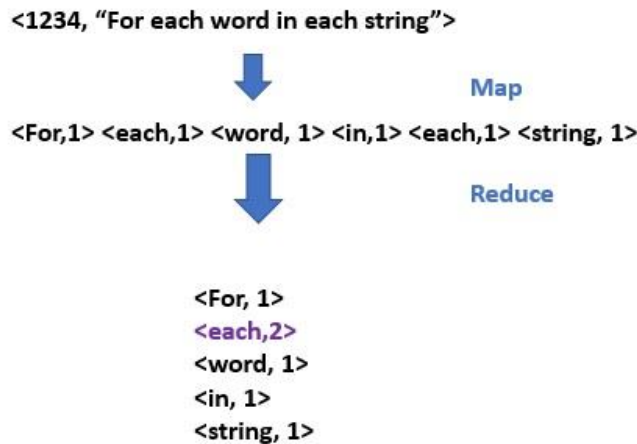


Figura 5 Map Reduce

Expanding el ejemplo del conteo de palabras, la salida final de un proceso de *Map Reduce* aplicado a un grupo de documentos podría tener la llave como un par ordenado y el valor

como una tupla o lista ordenada. En una aplicación típica el *Map* podría aplicarse a millones de líneas de texto y el paso de *Reduce* resumiría los pares de llave/valor.

Aunque *Map Reduce* es en teoría simple de comprender, no es tan fácil implementarlo en un sistema distribuido. Ejecutar un trabajo (job) de este tipo requiere la administración y coordinación de varias actividades:

- El trabajo de *Map Reduce* se debe agendar basado en la carga de tareas del sistema
- Los trabajos necesitan monitorearse y administrarse para asegurar que cualquier error se resuelva de manera apropiada de tal manera que cada trabajo pueda continuar si el sistema falla de manera parcial.
- Los datos de entrada deben repartirse en el clúster.
- El procesamiento del *Map* de la entrada necesita ser conducido a través de los sistemas distribuidos, preferentemente en las mismas máquinas donde residen los datos.
- Las salidas intermedias de numerosos pasos del *Map* necesitan ser recolectadas y repartidas a las máquinas apropiadas para reducir la ejecución.
- Las salidas finales necesitan estar disponibles para ser usadas por otros usuarios, aplicaciones o incluso otro trabajo de *Map Reduce*.

Apache Hadoop (<https://hadoop.apache.org>) es capaz de realizar todas esas actividades, además de contar con servicios para trabajar con sistemas de archivos distribuidos. Adicionalmente, muchas de las actividades se dan de una forma transparente para los desarrolladores, usuarios o científicos de datos.

Por otro lado, existen los HDFS o *Hadoop Distributed File Systems*, estos son sistemas de archivos que proveen la capacidad de distribuir datos a través de un clúster para aprovechar las ventajas del procesamiento en paralelo de *Map Reduce*. Para un archivo determinado HDFS descompone el archivo en bloques de 64 MB y almacena los bloques a través del clúster. Por ejemplo, si un archivo ocupa 300 MB, este es almacenado en cuatro bloques de 64 MB y uno de 44 MB.

Cada vez que es posible HDFS intenta almacenar los bloques de un archivo en máquinas distintas de tal manera que el paso de Map pueda operar en cada bloque de un archivo de



manera paralela. Además, HDFS crea 3 copias de cada bloque a través del clúster para proveer la redundancia necesaria en caso de falla. Para manejar el acceso de datos HDFS utiliza 3 procesos de Java (daemons), NameNode, DataNode y un NameNode secundario.

De esta forma Hadoop se ha convertido en uno de los marcos de trabajo más usados en operaciones de Big Data en todo el mundo, tanto por empresas privadas como por el sector público [4][19].

## Capítulo 2 - Organigrama

En este capítulo se describe la organización en la que se llevó a cabo el proyecto de City of Dallas, misma en la que he laborado por más de 8 años.

### 2.1 GBS AppOps

La organización es una de las empresas de tecnologías de la información más grandes del mundo. Cuenta con una historia de más de 100 años, operaciones en 170 países a nivel global y tiene más de 300, 000 empleados. Su peso en la industria es notable.

A manera organizacional dentro de la empresa se pueden considerar 6 grandes divisiones como se muestra en la figura 6. Tiene una división de servicios profesionales, a manera interna es una de las más grandes a nivel global. Dentro de esta se encuentra la sub división de servicios globales de negocios (*GBS*).



Figura 6 Divisiones de la organización

Además, están las áreas de Finanzas, Comercio, Seguridad, Analítica y las de Operaciones.

En la actualidad el portafolio de negocios de GBS se divide en tres áreas:

- Estrategia digital e iX
- Procesos de transformación cognitivos
- Innovación de aplicaciones en la nube

Dentro de innovación de aplicaciones en la nube se encuentra la línea de servicio *Cloud Application Migration Services (CAMS)*. En esta línea de servicio se encuentra *Application Operations on Cloud (AppOps)* como un servicio de hosting verticalmente administrado, antes *Development Support Team (DST)*. Recientemente cambió el nombre a *Devsecops*.

Dentro de DST (ahora parte de AppOps) se cuenta con la oferta de diversos servicios, incluyendo:

- Hospedaje de ambientes de desarrollo en cloud (a través de Blue Cloud) o de infraestructura tradicional en ambientes virtualizados a través de la propia infraestructura de AppOps, contando con 19 centros de datos alrededor del mundo.
- Administración de Seguridad, soporte a sistemas operativos y administración de redes.
- Soluciones de desarrollo de *Smarter Mobile* para aplicaciones, pruebas y mantenimiento.
- Soporte a *middleware* y aplicaciones incluyendo DB2, Oracle, SAP y WebSphere.
- Soporte y hospedaje de ambientes Rational y DevOps.
- Soporte a servicios de infraestructura personalizada.

En el proyecto de AppOps se atiende a clientes internos, es decir, a proyectos de desarrollo propio de la organización. Pero también a clientes externos, es decir, cuentas comerciales a nivel global.

En ambos casos el foco de los ambientes es de desarrollo y desarrollo/pruebas. Sin embargo, en algunos clientes se da servicio a ambientes pre productivos y productivos.

De manera jerárquica GBS se organiza de acuerdo a lo que se presenta en la figura 7.

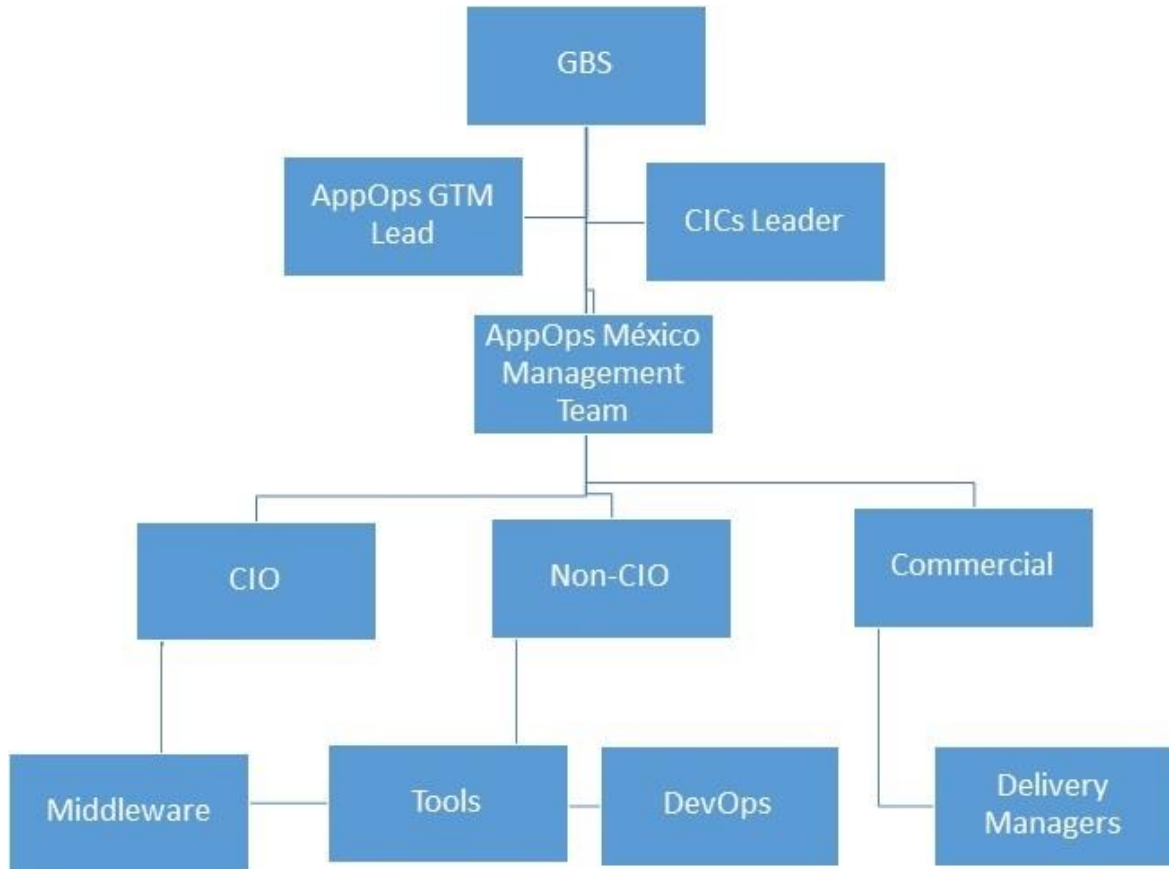


Figura 7 Organigrama de AppOps

*CICs* – *Competence Innovation Center*. Existen centros de este tipo en varios países a nivel global, en el caso de México, este se encuentra en la planta de la organización en Guadalajara.

*CIO* – *Chieff Information Office*

*Commercial* – Se dan servicios a muchas cuentas comerciales con clientes a nivel global. Se incluyen servicios de alojamiento de hardware (hosting), en servers tradicionales y cloud.

Además de soporte a sistemas operativos, seguridad, redes, monitoreo, bases de datos, servidores de aplicaciones y también a soporte de herramientas especializadas entre otros servicios.

De esta manera, en mi caso, la participación más reciente que he tenido ha sido en el equipo de delivery managers. Es un equipo con alcance global donde los miembros somos de diversos países como México, Brasil, Rumania, Canadá, Francia, India y China.

Algunos de los clientes comerciales con los que han habido proyectos son:

Exxon Mobil, City of Dallas, Racetrack, Marriot, Metlife, Electrolux, Bradesco, Santander, American Airlines, PBF Energy, Mercedes Benz, entre otros.

Dentro de AppOps se cuenta con una oferta global teniendo servicio de soporte 8x5, 24x5 y también de 24x7. Para esto se han integrado equipos globales con miembros y especialistas en distintas zonas geográficas como las Américas, Europa y África, así como Asia - Pacífico.

Se dan servicios de hospedaje, soporte a middleware, redes, OS y herramientas especializadas para desarrollo de aplicaciones. Tanto propietarias y licenciadas por la organización como de código abierto.

Los especialistas en soporte y los miembros del equipo de liderazgo de cada área especializada son también de diversos países, pero principalmente de Estados Unidos, México, Brasil, China, Filipinas, India, Egipto, Francia y Rumania.

La organización de AppOps a nivel jerárquico también es global, se cuenta con liderazgos locales para ciertas áreas, pero el soporte se divide a través de todo el mundo, en especial para las cuentas 24x7. Se aplica una práctica llamada localmente "*Follow the Sun*", es decir, seguir al sol. De esta manera si un especialista toma una tarea compleja en América esta puede ser pasada hacia China al final del día, después a Rumania y viceversa. De esta forma hay actividades que se llevan de manera continua y no se detiene el trabajo hasta completar las tareas pendientes. Esto se da con frecuencia cuando se realizan actualizaciones complejas en ambientes distribuidos y de topología de tipo empresarial. También sucede cuando hay un problema que impide la operación de las aplicaciones de los clientes.

El servicio se brinda a través de solicitudes, mismas que se registran por medio de la creación de tickets en uno de los sistemas propios de AppOps. El seguimiento se lleva a cabo por medio del ticket creado donde se asigna un responsable y se informa a los interesados de cada cuenta o proyecto.

La severidad se asigna de acuerdo a los estándares de AppOps pero también se corresponde a los niveles de servicio definidos para cada cuenta.

Es así que AppOps ha podido dar servicios de soporte a clientes de Norteamérica, Sudamérica, Europa, Asia y Oceanía. Esta división de GBS se ha ido adaptando de manera continua a las nuevas tecnologías y busca siempre estar a la vanguardia de los nuevos paradigmas dentro de la industria. En la figura 8 se describen de forma gráfica los diferentes servicios de AppOps.

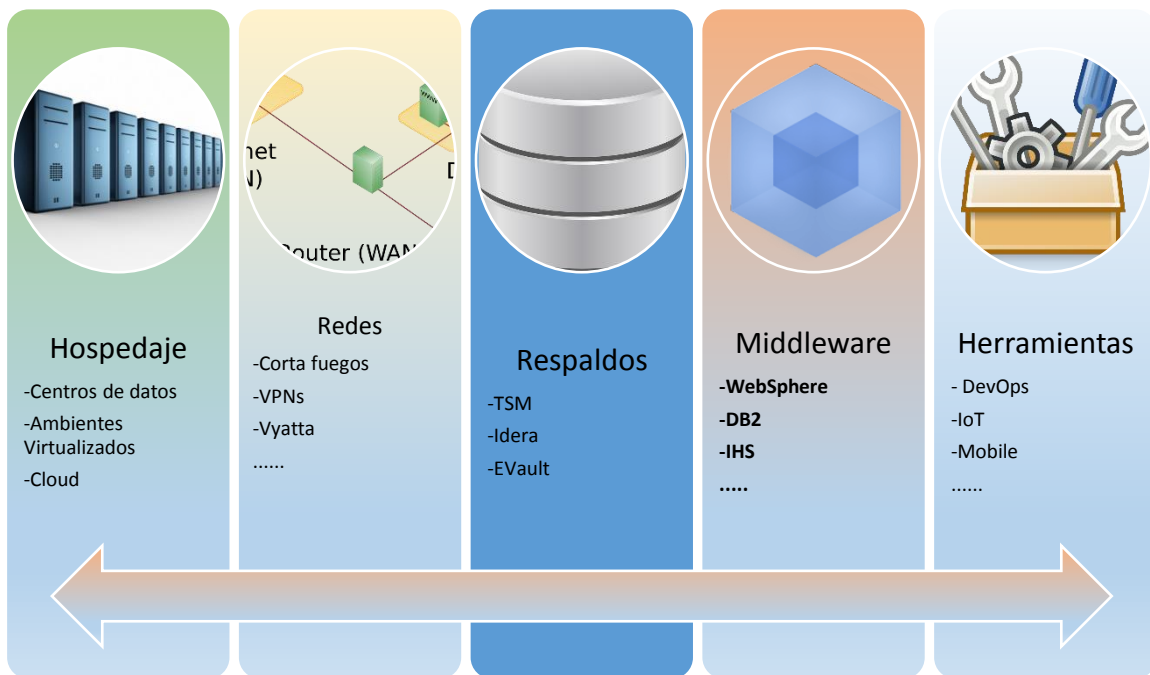


Figura 8 Servicios

## Capítulo 3 – Antecedentes y Descripción de Proyectos

En este capítulo se describen los roles que he desempeñado durante mi participación laboral en la organización; se mencionan los proyectos y las actividades.

### 3.1 Subject Matter Expert

El equipo AppOps se compone de varios roles, tanto administrativos como técnicos. Como es de esperarse para una empresa de TI, la mayor parte de los miembros son de un perfil técnico.

Uno de estos roles técnicos es el de *Subject Matter Expert* (SME). A este rol se asignan personas para dar soporte directamente a sistemas operativos, “*middleware*”, redes, bases de datos o herramientas. La persona asignada a este rol debe tener conocimientos técnicos suficientes para realizar funciones de administrador en algunas de las tecnologías soportadas por AppOps. Por lo regular se asignan personas con conocimientos de administración de sistemas (*sysadmin*), pero también a gente con conocimientos especializados en las herramientas a ser usadas.

Debido a la experiencia obtenida durante mis actividades previas en una empresa nacional, en esta organización se me asignó el rol de **SME para herramientas Rational**. Estas herramientas soportan el desarrollo de software. Comenzaron con el enfoque desde el *Rational Unified Process*, el cual se basaba en desarrollos iterativos, con un marcado énfasis en documentación, desarrollo de requerimientos e integración de herramientas en las disciplinas de RUP. Se basaba en desarrollo de casos de uso para a partir de su clasificación comenzar a construir los sistemas. Después, las herramientas evolucionaron hacia las metodologías ágiles y trabajo de tipo colaborativo para SCRUM y SaFE (<https://www.scaledagileframework.com>). En los últimos años, las mismas herramientas se han ido adaptando cada vez más para servir a proyectos con enfoque de DevOps e

integrándose a trabajar con herramientas de otras suites como Atlassian y HP; también incluyendo varias herramientas de código abierto como Jenkins y Git.

En esa etapa de mi historia laboral daba soporte especializado a ambientes con herramientas Rational, principalmente a CLM (*Collaborative Lifecycle Management*) pero también a herramientas para administración de requerimientos, para generación de Builds (convertir archivos de código fuente a ejecutables), de pruebas de automatización y de desempeño.

El soporte era remoto e incluía servidores ubicados en distintos centros de datos de Estados Unidos, Europa, Asia y unos pocos de Brasil. El proyecto involucraba unos 250 servidores, en su mayoría las herramientas estaban montadas sobre Linux Red Hat Enterprise, aunque había algunos servidores Windows también. La proporción era aproximadamente de 90% para Linux Red Hat y 10% para los servidores Windows.

Como SME se me asignaban tickets para actividades diversas, desde otorgar accesos y permisos, crear nuevos proyectos en las herramientas, instalar actualizaciones, instalación de ambientes pequeños, medianos y empresariales desde cero. En las actividades también se incluía la corrección de problemas complejos de configuración.

Algunas de las herramientas con las que trabajaba estaban basadas en *eclipse* (pues la organización ha sido patrocinador del proyecto) y otras eran desplegadas en web sobre múltiples servidores y trabajando generalmente con WebSphere Application Server o Tomcat. La configuración se llevaba a cabo de acuerdo a los requerimientos técnicos de cada plataforma y ambiente, aunque también influía el presupuesto de los clientes para usar una tecnología más completa u otra más económica.

## 3.2 Competency Lead

Otro de los roles técnicos dentro de AppOps es el de líder de competencia, oficialmente *Competency Lead*. Este rol se asigna a un SME suficientemente especializado en algún área de conocimiento. El objetivo de este rol es de ser el “experto” - dentro del equipo - en el área de conocimiento específica. Esto puede ser una herramienta o grupo de herramientas



relacionadas, un sistema operativo o alguna práctica. Como un ejemplo muy claro se puede mencionar al líder de competencia de Windows Server, de Linux Red Hat o de WebSphere Application Server. Entre las actividades esperadas se encuentra crear documentos específicos de instalación, configuración y de corrección a problemas comunes. Así como orientar a los demás SMEs sobre la competencia en específico, incluso generar nuevo conocimiento dentro del equipo. Es quien debe afrontar las nuevas versiones y las solicitudes más complejas de la herramienta o *middleware* en cada caso.

Después de casi un año y debido a mi experiencia con herramientas de pruebas y de herramientas de construcción o builds (hacía pruebas de desempeño en mi trabajo anterior), me asignaron el rol de líder de competencia (*Competency Lead*) para herramientas de pruebas. En especial para las herramientas Rational de pruebas incluyendo Rational Functional Tester, Rational Performance Tester, Rational Test Virtualization Server, Rational Integration Tester y Rational Performance Test Server. Posteriormente fui involucrado en actividades para actualizar ambientes de Build Forge a las últimas versiones y desplegando sobre WAS, también desarrollé competencias para las herramientas de Builds, incluyendo Rational Build Forge y Jazz Build Engine. En la figura 9 se indican gráficamente las actividades principales del líder de competencia.

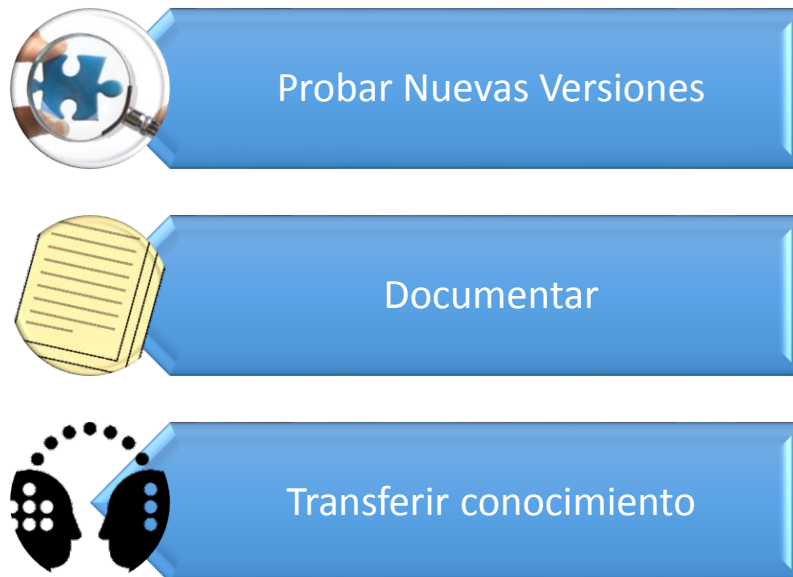


Figura 9 Actividades de Competency Lead

En este rol realizaba las pruebas iniciales de instalación y configuración para las nuevas versiones de las herramientas. Por otro lado, hacía documentación interna sobre los procedimientos tales como los documentos de instalación y configuración, integraciones, migraciones, etc. Adicionalmente trabajaba en la transferencia de conocimiento hacia otros miembros del equipo a nivel global a través de algunas sesiones remotas. Los compañeros a quienes usualmente se les daban entrenamiento y con quienes compartía la información eran principalmente de China, Egipto y Brasil.

Las actividades se orientaban a desarrollar la competencia, investigar sobre integraciones con herramientas Rational y con otras herramientas de pruebas de empresas diferentes a la organización.

Por otro lado, en actividades de soporte del día a día me consultaban solo sobre solicitudes de complejidad media y alta cuando los SMEs, tenían dudas al respecto sobre las herramientas de pruebas. Por ejemplo, cuando se tenía un ambiente tipo empresarial caído y había que restaurarlo tan pronto fuera posible era una complejidad media. O cuando después de varios intentos no podían reestablecer la funcionalidad era complejidad alta. También acudían a mi cuando se pedía algo nuevo por parte del cliente, algo fuera de los procedimientos más comunes y conocidos tales como integraciones, personalizaciones o nueva funcionalidad.

En paralelo desarrollé otros conocimientos y tomé entrenamiento de diversas disciplinas como arquitectura, diseño y mejores prácticas en desarrollo de soluciones para la organización. También tomaba a mi cargo solicitudes complejas de ambientes con otras herramientas Rational como *Collaborative Lifecycle Management (CLM)*. De hecho, esta suite de herramientas era la más utilizada por los clientes internos y comerciales. Se contaba con cerca de 200 ambientes a los cuales se les daba soporte. Los ambientes principalmente se encontraban hospedados en centros de datos en Estados Unidos, Francia, China e India. Me certifiqué en la versión 4 de CLM y era de los SMEs con más conocimiento en la suite por parte del equipo de México y a nivel global.

CLM era la suite más utilizada en ese entonces en todos los proyectos internos de desarrollo y era también de bastante interés para clientes externos.

La suite de CLM incluía:

- Rational Team Concert. Esta herramienta se usaba para administración de proyectos y administración de la configuración. Evolucionó y cambio el nombre a Change and Configuration Managment (CCM).
- Rational Quality Manager. Para administración de la calidad y pruebas. Se integra con herramientas de pruebas automatizadas (RFT) y de pruebas de desempeño (RPT).
- Rational Requirements composer. Enfocado en la administración de requerimientos. Al principio basado en un cliente eclipse, evolucionó a ser totalmente web y cambió de nombre a Doors Next Generation.

Después se adicionaron herramientas para reportes como Jazz Reporting services (JRS) con Data Collection Component (DCC).

Estas herramientas pueden trabajar de manera independiente cada una o integrarse y formar un ambiente robusto para soluciones de tipo empresarial. Se integran sobre la plataforma del Jazz Team Server (JTS) de la organización.

Estas son herramientas web que se despliegan sobre Tomcat y WebSphere Application Server. A partir de la versión 5 se utiliza principalmente la segunda opción

Como tenía aún actividades de SME para CLM se me asignó realizar instalaciones y configuraciones. Esto en ambientes todo en uno (*all in one*) y distribuidos en varios servers o Enterprise Topology, tanto para cuentas internas y comerciales. Prácticamente trabajé con todas las topologías mencionadas en la documentación oficial de CLM.

### 3.2.1 Proyectos como Competency Lead

Como Competency Lead y SME participé en diversos proyectos, a continuación se describen algunos de ellos.

#### 3.2.1.1 Actividades con Herramientas de pruebas

El aseguramiento de calidad de software ha sido parte importante en mi experiencia laboral, durante varios años trabajé con herramientas para diversos tipos de pruebas de software. A continuación se describen algunos de mis proyectos como líder de competencia en este tipo de herramientas.

##### **RFT DEMS**

**Rol desempeñado: Competency Lead,**

**Tiempo de actividad: 1 mes**

Instalación de ambiente con Rational Funcional Tester para pruebas automatizadas para regresión. Actualizaciones a nuevas versiones, restablecer ambiente cuando había errores. Integración con otras herramientas como Quality Manager y Team Concert.

##### **RPT GSA**

**Rol desempeñado: Competency Lead**

**Tiempo de actividad: 1 mes**

Instalación de ambiente con Rational Performance Tester y dos agentes para distribución de carga. El workbench se instaló en Windows para contar con la parte grafica completa y los agentes en Linux para tener mayor eficiencia en el uso de la memoria. Se configuro también un Servidor de licencias y un administrador para trabajar con los usuarios virtuales durante

las pruebas de carga. Adicionalmente se realizaron adecuaciones durante ejecuciones ya que en ocasiones existían limitaciones de recursos con los agentes.

En la figura 10 se puede apreciar la arquitectura general de un ambiente de pruebas con herramientas Rational.

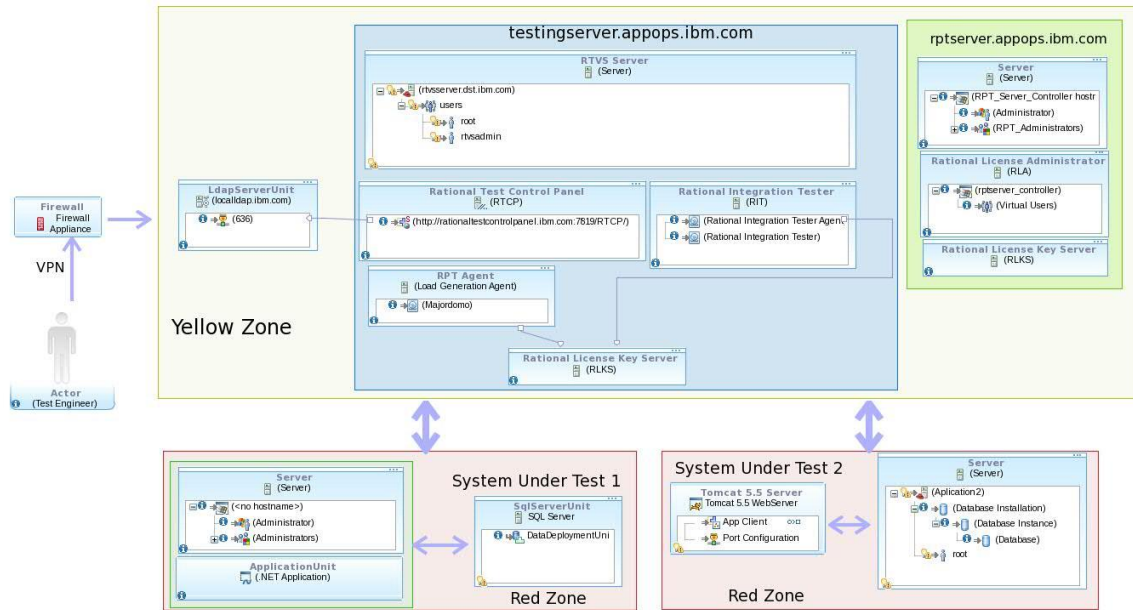


Figura 10 Arquitectura típica de un ambiente de pruebas con herramientas

## GaIN

### Rol desempeñado: SME

### Tiempo de actividad: 1 año.

Las labores estuvieron principalmente relacionadas a ambientes CLM. Se realizaron desde nuevas instalaciones, actualizaciones y solución de problemas. Se trabajó con ambientes “todo en uno” y distribuidos de topología de tipo empresarial donde cada aplicación estaba en un server dedicado y se integraban a través del Jazz Team Server. Los ambientes en su mayoría se desplegaban sobre WebSphere Application Server, pero también algunos en Tomcat.

### 3.2.1.2 Herramientas de Builds

Otra competencia que desarrollé fue sobre las herramientas para construir los empaquetados de software (*builds*), los ambientes debían estar listos para ser usados por los desarrolladores. A continuación se describen algunos de los proyectos relacionados a las herramientas de *builds*.

#### **PIS BRADESCO BuildForge**

**Rol desempeñado: Competency Lead,**

**Tiempo de actividad: 1 mes**

Se realizó una actualización en un ambiente desplegado en WebSphere Application Server. Me tocó abrir una solicitud a especialistas en Estados Unidos ya que se trataba de una instalación poco común. La actualización tomó algunos días ya que se tuvo que revisar bastante información, las guías para hacer la mejora en el sistema tenían algunas inconsistencias. Fue necesario hacer adecuaciones sobre la marcha, por ejemplo, restaurar dos veces las bases de datos desde los respaldos y empezar de nuevo los procedimientos. Fue un proyecto complejo por contar con muy poca documentación y por ser un procedimiento muy poco común dentro del equipo de trabajo.

### 3.2.1.3 Herramientas de CLM

De manera adicional a la competencia de herramientas de pruebas y *builds*, también atendía solicitudes de CLM. Esto por ser de los miembros del equipo con más experiencia y conocimiento en esa tecnología.

En cuanto a estas herramientas CLM, participé en varios proyectos como SME, a continuación se describen algunos de ellos.

## Exxon.

### **Rol desempeñado: SME**

#### **Tiempo de actividad: 3 meses.**

En este proyecto participé en la instalación y configuración de un ambiente de Rational Team Concert para administración de proyectos en cloud (Soft Layer). Fue necesario trabajar de manera cercana con especialistas de cloud para poder hacer la transferencia de archivos de instalación, configurar parámetros de redes y también de respaldos ya que el proveedor de cloud utilizaba tecnologías específicas para esas tareas, como el Aspera de object storage. Me tocó trabajar de manera muy cercana con administradores de bases de datos (DB2) para hacer unas personalizaciones requeridas para los respaldos. La tecnología que se utilizaría en *cloud* (eVault) solo soportaba Oracle de manera directa. En este caso era necesario volcar la información de las bases de datos hacía unos sistemas de archivos y configurar una tarea automática (*cron*) diaria. Para cada Base de Datos se realizó este tipo de configuración y se creó un sistema de archivos para cada una de ellas. De esta forma eVault ya podía tomar el respaldo correspondiente.

## Lenovo.

### **Rol desempeñado: SME**

#### **Tiempo de actividad: 3 meses.**

Instalación y configuración de un ambiente de Rational Team Concert para administración de proyectos en una zona de red intermedia entre la red privada de la organización y permitir comunicación hacia el exterior (*yellow zone*). En este proyecto hubo que organizarse para empalmar los horarios ya que el cliente estaba en Reino Unido y algunos especialistas de Lenovo en China. Me tocó coordinar las reuniones con equipos de infraestructura para establecer las conexiones de red necesarias y tener listo el ambiente para los usuarios.

En la figura 11 es posible apreciar una arquitectura general de un ambiente CLM distribuido donde cada aplicación es instalada en un servidor.

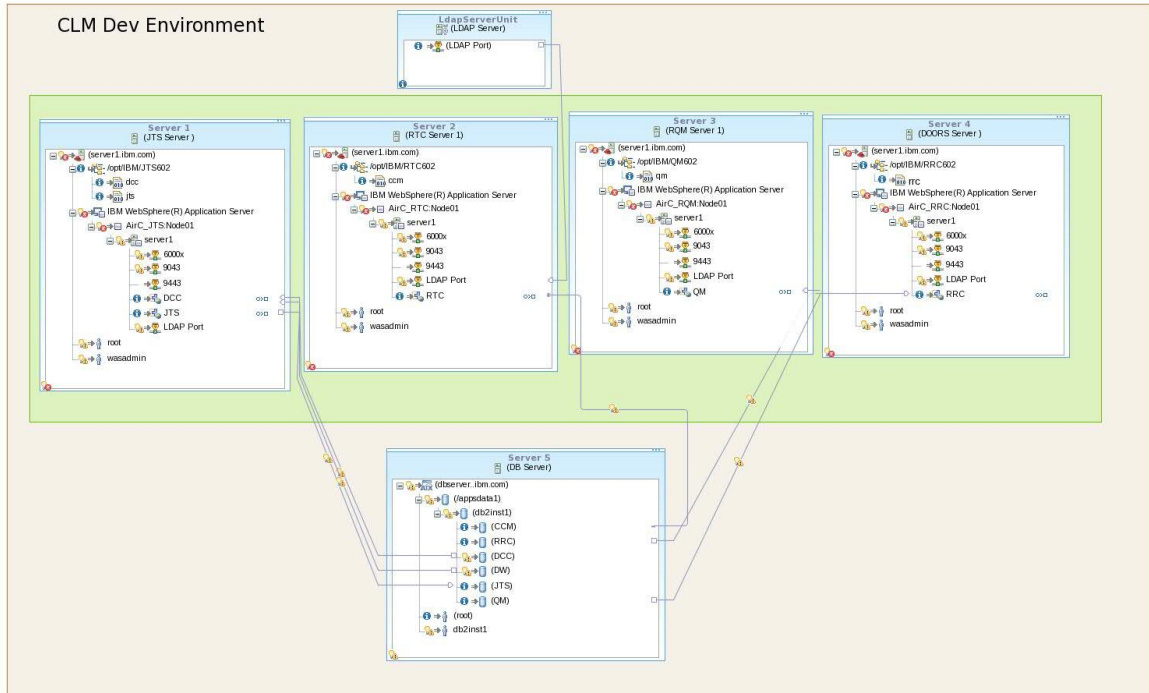


Figura 11 Arquitectura típica de un ambiente CLM distribuido

## DEMS.

**Rol desempeñado: Competency Lead**

**Tiempo de actividad: 1 mes.**

Instalación y configuración de ambiente de pruebas de desempeño con Rational Performance Tester. Se instaló un controlador y dos agentes. El controlador en Windows y los agentes en Linux ya que así el manejo de memoria se comporta de manera más eficiente para este tipo de herramientas.



### 3.3 Solution Designer

El rol de Solution Designer es un puesto técnico que requiere más experiencia y tener un conocimiento amplio de las interacciones de diferentes sistemas para crear una solución completa. Está ligado más a conocimientos de arquitectura y a tener contexto de varias tecnologías, así como de trato con cliente para hacer un levantamiento adecuado de requerimientos.

Este rol consiste en levantar requerimientos técnicos directamente del cliente, funcionales y no funcionales. A partir de la información recolectada diseñar la solución que mejor se adapte a sus necesidades técnicas, pero también a su presupuesto. Se comunica la propuesta, si el cliente acepta la solución y el costo, se lidera la construcción; así como la entrega de la solución.

Debido a mi experiencia con la mayoría de las herramientas de Rational, incluyendo la suite de CLM y las herramientas de pruebas, me asignaron para participar en otras actividades. Se tomó en cuenta también mi interés en temas de arquitectura de soluciones y varios cursos que tomé sobre esa área de conocimiento como por ejemplo “Architecture Thinking” y “Team Solution Design”, por mencionar algunos. Se me asignó al equipo global de Solution Designers a cargo de Christine Poisvert, una especialista y arquitecta de Francia.

En este rol es donde he desarrollado más proyectos desde el enfoque de soluciones completas para los clientes. Me involucré de lleno en el diseño de las soluciones. Empecé con ambientes Rational para todas las herramientas de la suite. Principalmente trabajé nuevamente con herramientas CLM (por ser la que contaba con más ambientes) y de la suite de pruebas para entonces llamada Rational Test Workbench. Esta última incluía alrededor de 5 herramientas especializadas para diferentes tipos de pruebas.

En ocasiones cuando se necesitaban algunos detalles sobre las últimas versiones de las herramientas o información muy específica, era necesario contactar y aclarar información con los líderes de la competencia (Competency Leads) en turno para obtener los datos técnicos. Esto también es común dentro del rol de Solution Designer.

Trabajaba de forma muy cercana con el equipo de Engagements, ellos eran los encargados de definir los costos y de cerrar las ventas con los clientes. Por lo general me contactaban ya con un cliente interesado en adquirir los servicios del equipo de AppOps (entonces DST). La gente de Engagements acudía conmigo ya con la oportunidad y con una solicitud a través de un ticket en una herramienta interna. Ahí se registraba la información a manera muy general de los requerimientos del cliente para sus ambientes.

Al principio yo recababa más información por correo electrónico o haciendo preguntas para obtener más detalles a través del mismo ticket de la solicitud. Sin embargo, al adquirir más experiencia me di cuenta que una llamada ayudaba en gran manera a acelerar las cosas. Normalmente las llamadas eran en inglés, ya que la gran mayoría de los clientes potenciales eran de Estados Unidos, unos pocos de Europa y los menos (solo recuerdo uno) de México o Latinoamérica.

El equipo de Solution Designers se organizaba por zonas geográficas. La gente de Francia (después los de Rumania) atendía las oportunidades de Europa. En mi caso me tocaba atender junto con gente de Brasil las oportunidades de las Américas, incluyendo Estados Unidos. De esta manera, por el tamaño de economía y servicios, nos tocaba trabajar la mayoría de las soluciones, ya que el porcentaje de solicitudes de Estados Unidos era notoriamente grande. Los compañeros de China atendían la zona Asia Pacífico.

Sin embargo, si la oportunidad requería cierta especialización se daba el caso de trabajar con soluciones de zonas geográficas distintas.

En mi caso, trabajé en algunas oportunidades con equipos y clientes de Italia, España y Francia. También trabaje con equipos de Nueva Zelanda, con estos últimos logramos construir soluciones de pruebas.

Parte de mis responsabilidades involucraban tratar de reunir tanta información como fuera posible durante la primera llamada. Desde el número de usuarios concurrentes, el uso que se le daría al ambiente, las integraciones, el tipo de usuarios, la carga de trabajo (basada en el tipo de artefactos que se generarían), si requerían comunicación a internet y tantos detalles como pudiera proporcionar el cliente. Se trataba de información tanto técnica como de la funcionalidad esperada de los ambientes y también de los servicios. Seguía las prácticas de

levantamiento de requerimientos de la ingeniería de software. Fue de gran ayuda la experiencia que tenía sobre aseguramiento de calidad, pues recibí entrenamiento y de hecho me certifiqué como analista de requerimientos y casos de uso. En estos proyectos por tratarse de infraestructura y herramientas ya listas, los requerimientos más importantes eran los no funcionales.

Una vez recabada la información procedía a hacer un “*sizing*”, es decir un dimensionamiento en cuanto al número de servidores que serían necesarios, hardware de cada uno de los servidores, zona de red que correspondía, así como la topología necesaria que correspondería con respecto a la distribución de las herramientas.

Me servía de información técnica disponible en los sitios oficiales de las herramientas, de la experiencia propia con ambientes similares y de conversaciones con los competency leads en turno para cada herramienta. Es decir, analizaba los requerimientos de sistema operativo y de hardware de acuerdo al tamaño y necesidades de los ambientes. Así como también de redes, pues de acuerdo al perfil había que tomar en cuenta ciertos aspectos de seguridad y comunicaciones.

Como se comentó con anterioridad, empecé trabajando básicamente con herramientas de CLM y de pruebas de la suite Rational. Sin embargo, después de un año AppOps involucró ofertas de servicio con otras herramientas para DevOps. Algunas de estas eran Jira, Confluence, Docker, Selenium, Urban Code Deploy.

Durante una buena parte del 2017 la mayoría de las oportunidades que llegaban fueron relacionadas a este tipo de herramientas de código abierto, mismas que han tenido un auge notable para DevOps durante los últimos años. Son herramientas más ligeras que pueden funcionar con menores recursos de hardware y por lo tanto convienen a clientes con presupuestos más moderados. No cuentan con las mismas funcionalidades que las herramientas licenciadas pero sus capacidades, aunque limitadas, alcanzan a cubrir muchos de los requerimientos de los clientes pequeños y medianos. Además, pueden trabajar en conjunto con otras herramientas de software libre y así logran cubrir más de las funcionalidades con las que cuentan las herramientas propietarias. Aunado a lo anterior, es rápido hacer el despliegue y crecer los ambientes (o reducirlos) de una manera sencilla en comparación a las soluciones empresariales de caja.

Tenía mucha comunicación y trabajo colaborativo con el equipo de Engagements para el cierre de ventas y la parte técnica la revisaba con los líderes de competencias. Por estas razones las llamadas telefónicas eran muy comunes al trabajar en este rol. La comunicación era muy importante para puntualizar los detalles y acelerar el trabajo.

Una vez con toda la información técnica y con los requerimientos detallados procedía a hacer el diseño de la solución. Para esto consultaba además informes documentados de *benchmarks* sobre diferentes tipos de hardware y sistemas operativos. Entonces trabajaba en determinar la mejor opción de sistema operativo, ya que algunas herramientas funcionaban mejor en Windows y otras en Linux. Además, identificaba si la mejor opción era un sistema operativo tipo server o si con una versión no server era suficiente. A continuación, procedía a definir la cantidad y tipo de hardware que debía usarse. Con ciertos componentes como con agentes o contenedores la memoria podía ser mejor aprovechada en Linux. Otras herramientas tenían mejor desempeño en un ambiente Red Hat Enterprise Linux. Por otro lado, ciertas herramientas de pruebas tenían la funcionalidad completa solo en Windows y en otros sistemas operativos estaban limitadas. En algunos componentes como en los bancos de trabajo (*Workbenches*) era suficiente una máquina con una versión estándar de Windows. Todos estos aspectos eran tomados en cuenta al diseñar las soluciones.

Era bastante la información a revisar y se debía ser cuidadoso ya que un mal diseño podía disparar costos en etapas posteriores o derivar en ambientes inestables. Además, tomaba en cuenta las plantillas de servidores a utilizar dependiendo de la zona de red en que se necesitara albergar a los servidores. La gran mayoría eran servidores virtuales que se podían crear automáticamente a través de una solicitud en un portal interno, había algunos más llamados tradicionales que a pesar de ser también virtuales eran creados de manera casi manual y en un entorno donde sus recursos estaban más aislados. Por otro lado, los servidores de cloud eran creados a partir de una solicitud específica en Soft Layer. Contando así con servidores cloud estándar y de tipo *bare metal* dependiendo del *hiper visor* que se utilizaría. También se tomaban en cuenta los servicios y la confidencialidad de la información que produciría y guardaría el cliente. Todo esto podía modificar los costos e influir en redimensionamientos.

De manera especial se debía analizar el tema de seguridad. Si no se contaba con la aprobación de los especialistas en la materia una solución podía demorarse de manera considerable, modificarse, o incluso cancelarse. En mi caso agendaba llamadas con el equipo interno de seguridad una vez que contaba con un diseño preliminar de la solución.

En la figura 12 se puede observar el flujo completo para llegar a una solución

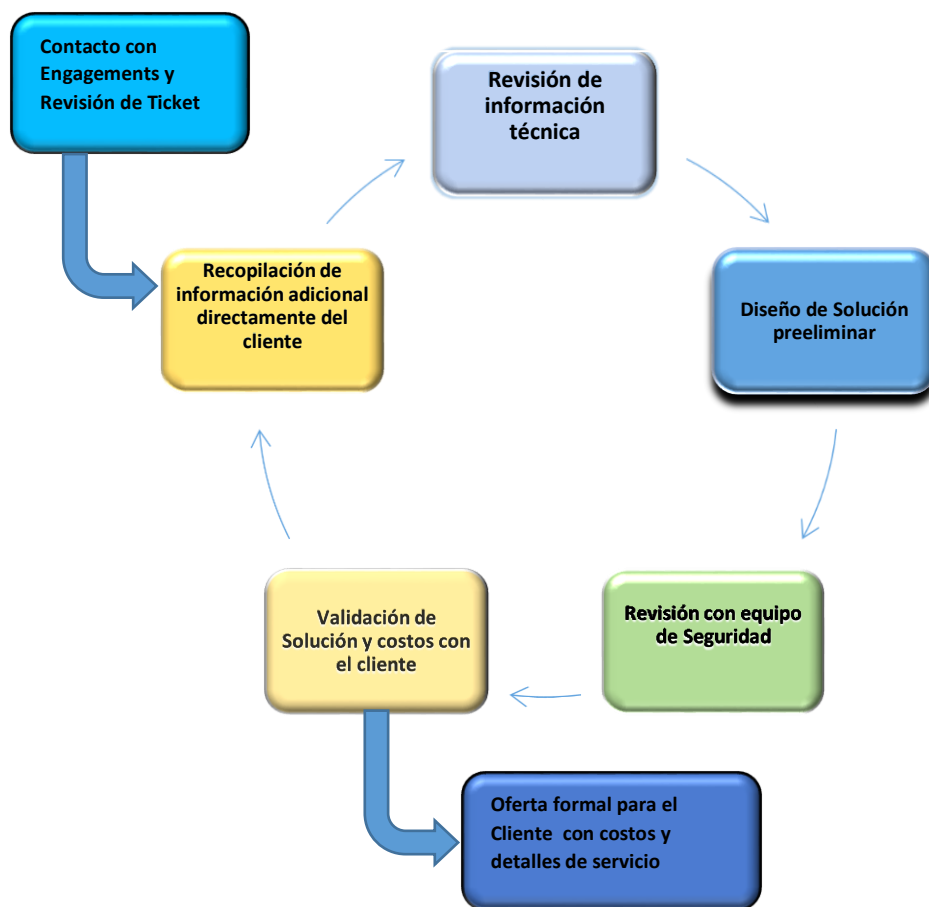


Figura 12 Flujo de soluciones

### 3.3.1 Proyectos con Herramientas de Pruebas

Con respecto a las herramientas de pruebas, trabajé con todas las que eran parte de la suite de Rational. A continuación se describen algunos de los proyectos.

#### *ANZ*

**Rol desempeñado: Solution Designer**

**Tiempo de actividad: 2 meses.**

En este proyecto participé haciendo la instalación y configuración de uno de los primeros ambientes de Rational Test Virtualization Server. Esta herramienta se utilizaba para hacer pruebas a través de “*stubs*”, es decir, emular componentes de software por medio de servicios virtualizados antes de que estuviera listo el ambiente. Se instaló un server en Windows y otro en Linux, además de un agente de Rational Integration Tester para poder ejecutar los “*stubs*”. De forma adicional se instaló la parte del Rational Test Control Panel para que los usuarios pudieran generar su ambiente virtualizado de pruebas.

#### *State of California*

**Rol desempeñado: Solution Designer,**

**Tiempo de actividad: 2 meses.**

Este proyecto se relacionó a ambientes CLM para el gobierno del estado de California en Estados Unidos. Después de tener llamadas con el cliente se hizo el diseño de una solución que incluiría dos ambientes de CLM. Estos ambientes se crearon e instalaron para administración y desarrollo de proyectos de software. Se generó un ambiente en red “intermedia” llamada Yellow Zone para que pudieran salir los clientes hacia internet y a su vez tuvieran comunicación con algunos servidores de la organización. En este caso fue necesario establecer comunicación a través de enlaces directos por IP, para esto estuve trabajando con el equipo de redes. Por otro lado, se generó otro ambiente en la red interna, en aquel estarían trabajando solo empleados de la organización que estarían dando servicios al cliente. El proyecto comenzó en 2016 y siguió activo hasta 2018.

## *Watson*

### **Rol desempeñado: Solution Designer y SME**

#### **Tiempo de actividad: 1 mes.**

Para este proyecto se me pidió construir unas plantillas para ambientes virtualizados internos. A partir de estas se generarían grupos de servidores para ejecutar unas pruebas de desempeño con cargas bastante grandes. Tuve reuniones con un arquitecto de Watson (la solución de inteligencia artificial de la organización) y con los encargados de grabar los scripts y ejecutar las pruebas. Las plantillas debían ser para la última versión de Rational Performance Tester (RPT), en ese entonces (8.7), ya que la intención era hacer pruebas de desempeño muy robustas para una arquitectura con Watson. Generé un servidor virtual con Linux Red Hat e instalé la versión más completa de RPT incluyendo agentes. Después trabajé de manera cercana con especialistas del equipo de virtualización para probar que la plantilla pudiera generar servidores de manera automática y que estos funcionaran bien. Al término de algunas pruebas la plantilla se desplegó como disponible solo en algunos centros de datos específicos de Estados Unidos. Esto se hizo para evitar afectar ciertas zonas de red debido al gran tráfico que se estaría generando. Se crearon 10 servidores para trabajar como controladores y 100 servidores para funcionar como agentes. Adicionalmente se configuraron servidores de licencias para trabajar con 100, 000 usuarios virtuales. El ambiente estuvo activo solo dos semanas por los altos costos de tener un ambiente de pruebas tan grande. Al final el ambiente resultó muy útil para ese proyecto interno y generó buenos ingresos para AppOps en poco tiempo.

### 3.3.2 Proyectos con Herramientas de CLM

Una parte importante de mi trabajo con la suite Rational fue la interacción con el Collaborative Lifecycle Management (CLM), un conjunto de herramientas para administrar requerimientos, proyectos de desarrollo y calidad. A continuación se describe mi participación en uno de los proyectos con CLM.

#### *HCS Secret*

##### **Rol desempeñado: Solution Designer**

##### **Tiempo de actividad: 2 meses.**

Se me contactó para una oportunidad con un cliente interno de Canadá. El primer contacto fue a través de una *project manager* que había tenido proyectos como cliente interno con AppOps. La experiencia técnica y los servicios que se le habían dado contribuyeron a que recomendara trabajar con nuestro equipo. Estuve de manera inicial a través de chats con ella y con otros miembros de su equipo de Canadá. Con base en los requerimientos planteados di una estimación básica, pero se acordó hacer llamadas para detallar más la información necesaria. Durante dos llamadas subsecuentes se involucró más gente del proyecto quienes puntualizaron algunos requerimientos y pidieron información detallada acerca de cómo se crearía la solución.

A partir de esas reuniones preparé un documento detallado con la información correspondiente a la solución incluyendo:

- Información de la topología CLM de tipo empresarial para la solución.
- Diagrama de arquitectura.
- Listas de verificación para la instalación de las aplicaciones CLM.
- Lista de verificación para instalación del servidor de base de datos (DB2).
- Detalle de los servidores a ser creados con información de los recursos de hardware, directorios de instalación y aplicaciones contenidas en cada uno.



- Lista de usuarios requeridos para la instalación y configuración de las aplicaciones y bases de datos.
- Información de los perfiles requeridos y los directorios a ser creados para el servidor de aplicaciones (WebSphere Application Server).
- Pasos para la instalación y configuración del ambiente.
- Requerimientos y configuración de red para hacer el despliegue de las aplicaciones, incluyendo los puertos a ser abiertos para comunicación bidireccional.
- Información del licenciamiento requerido.
- Recomendaciones y opciones para realizar los respaldos diarios (el cliente se haría cargo de esas labores).

Verifiqué una parte de la información con algunos líderes de competencias, en este caso para CLM y Urban Code Deploy.

La solución fue aceptada por el cliente y se generó un documento de entendimiento. AppOps estuvo a la espera por varios meses ya que el cliente contaba con recursos de hardware limitados. Finalmente se creó el ambiente y permaneció funcional por alrededor de ocho meses.

## Capítulo 4 - Contexto de la Participación Profesional

En este capítulo se describe el rol que desempeño hasta el presente; así como mi participación en el proyecto de City of Dallas. Dicho proyecto es la parte central del presente informe.

### 4.1 Delivery Manager

A mediados de 2017 se me ofreció un cambio de rol y pasar de Solution Designer a Delivery Manager, un rol más relacionado al liderazgo a nivel global dentro de AppOps. Esto era un requisito para poder tener una promoción dentro del equipo. Con anterioridad había tomado entrenamiento formal de fundamentos de *project management*, así que contaba con educación sobre ese tipo de actividades. Debido a esto, por mi experiencia en otros roles y por mi involucramiento cada vez mayor en actividades de tipo estratégico dentro del equipo, se me asignó trabajar con 3 cuentas. Empezando con un proyecto de Big Data para City of Dallas, adicionalmente un proyecto de *cloud* para Electrolux (la empresa sueca de electrodomésticos) y otro con middleware para desarrollo de aplicaciones móviles.

Se trató de un nuevo reto a asumir, donde sobre la marcha tuve que desarrollar nuevas habilidades y capacidades. En este rol estaría teniendo trato directo con los clientes comerciales y liderando a los especialistas de las herramientas y soporte de infraestructura.

Mi interacción personal fue desde entonces mucho mayor, ya que de ser uno de los participantes principales pasé a ser el organizador de las reuniones y llamadas con los clientes. Principalmente de Estados Unidos y en particular de la ciudad de Dallas, donde trabajé con City of Dallas, el proyecto que ha tomado más de mi tiempo y dedicación.

Este rol ha involucrado muchos retos, ya no tanto desde el punto de vista técnico, sino que de liderazgo y negociación. He desempeñado estas actividades por más de dos años y ha sido mi labor principal hasta el cuarto trimestre de 2020.

De las primeras cuentas que se me asignaron, dos de ellas ya estaban en actividades de soporte desde hacía meses, es decir, tomé una transición y las cuentas eran relativamente sencillas.

Sin embargo, la cuenta de City of Dallas apenas estaba comenzando y varias cosas estaban por definirse. De hecho, los ambientes y servidores aún no estaban configurados y no había ni siquiera personal dentro del equipo para soportar las herramientas requeridas. En este contexto nadie de mi equipo había trabajado con cuentas de Big Data, tecnología relativamente nueva. Nadie tenía experiencia con las herramientas y muchas cosas eran nuevas para todo el equipo global de AppOps.

Se trataba entonces de un reto aún mayor, el primer proyecto que me tocaría administrar como líder principal era de una complejidad mayor en comparación con el promedio de todas las otras cuentas de AppOps y con tecnología poco conocida en ese entonces.

## Capítulo 5 - Análisis y metodología empleada

En este capítulo se describe el comienzo del proyecto, la información inicial recibida y las razones que me llevaron a decidir administrar el proyecto con ciertas prácticas en cuanto a metodología.

### 5.1 Despliegue

Con base en el entrenamiento que había tomado, revisé que opciones podría aplicar para llevar la administración del proyecto. Aun cuando la metodología ágil estaba muy promovida dentro de la organización y tenía mucho impulso, no parecía adecuarse por completo. El proyecto no era de desarrollo de software. De esta forma los roles, actividades y dinámicas planteadas en metodología ágil no se adaptarían más que en algunas pocas cosas. Por ejemplo, no era práctico usar historias de usuario ya que los requerimientos estaban muy definidos y en realidad no había un dueño de producto (*product owner*) que solicitara cambios, el software se instalaría de caja.

El alcance definido para mi equipo era de soporte a la plataforma de BigData de City of Dallas, en específico a las herramientas. Sin embargo, como se verá más adelante, tuve que tomar actividades adicionales sobre la marcha.

Mi alcance original estaba enfocado a dar soporte rutinario a las herramientas, desde reparar errores, instalar correcciones menores, cuestiones de permisos, hasta hacer actualizaciones sencillas.

Partiendo de lo anterior se planeó hacer revisiones semanales con el equipo, hacer un plan de monitoreo a largo plazo y planear cada actualización de acuerdo a las distintas versiones de las herramientas a manera de una corta iteración. Para cada cambio se generaría un pequeño plan de trabajo de tipo clásico, incluyendo fechas, responsables, etc. Además, agendaría previamente llamadas con el cliente para revisar los días y horas que mejor se adecuaran. Se trataba de una cuenta con soporte 8x5, de lunes a viernes, de 9 de la mañana a las 6 de la tarde. De esta forma todo parecería ser rutinario. Sin embargo, más adelante se podrá notar que enfrenté muchos retos durante todo el proyecto. Además de que me delegaron mayores

responsabilidades, de hecho, todo un rol adicional, por todo esto fue necesario modificar mi estrategia de trabajo original

Sin embargo, el mundo entero es de cambios, en especial la industria de las tecnologías de la información. También como parte de mi análisis decidí aceptar los retos que llegaron, puesto que todo eso me permitiría desarrollar nuevas habilidades

Las primeras adecuaciones a mis actividades llegaron muy pronto y se trataba de labores que realmente no esperaba. En su primera fase, las labores se enfocarían en hacer el despliegue de los ambientes de desarrollo y producción. Una vez terminado el despliegue, lo que seguiría sería el soporte a la plataforma de Big Data. De acuerdo a la información disponible al momento, mi equipo estaría a cargo de 4 herramientas:

- BigInsights
- BigIntegrate
- Cognos
- SPSS

En mi caso además de la educación de administración de proyectos, contaba con experiencia en el modelo iterativo de RUP y aseguramiento de calidad de productos y procesos. Sin embargo, en este caso los requerimientos estaban ya establecidos. Así que, lo más parecido al modelo iterativo era tomar el despliegue como dos iteraciones y seguir las especificaciones técnicas para asegurar una apropiada entrega de los ambientes.

Por otro lado, una parte del despliegue la estaría llevando el equipo de Lab Services, uno de los equipos más experimentados en despliegues dentro de la organización. La segunda parte mi equipo, puesto que era muy costoso para el proyecto mantener a Lab Services en el proyecto por más tiempo. Para entonces ya había un plan de proyecto creado por el administrador del proyecto. Ese plan incluía toda la parte de despliegue y una de transición hacia el cliente y también a mi equipo.

De esta manera, el plan de proyecto ya establecido y acordado con el cliente adoptaba el modelo clásico. Fue actualizado para incluir las actividades de despliegue en producción y responsables, así cubría desde el inicio del despliegue hasta la puesta en marcha del soporte a la plataforma. Sin embargo, el despliegue del ambiente de producción que se asignó a mi equipo tenía varias actividades macro y nos daban tres semanas para completar todas las actividades.

Fue así que se me pidió planear a detalle todas las actividades y también llevar el seguimiento de ciertos costos, por solicitud del cliente el despliegue de producción se llevaría a cabo en las instalaciones del cliente en Dallas, Texas.

En este caso decidí trabajar en la planeación elaborando el plan detallado de despliegue en producción, usando el modelo clásico y manejarlo a manera de una iteración. Aun cuando no se estaría haciendo desarrollo si no solo despliegue. La base para esto era una iteración de despliegue en un modelo iterativo, en caso de que hubiera fallas. Era lo que se adecuaba más de acuerdo a las circunstancias del momento. Fue así que descompuse todas las actividades macro en tareas muy específicas para las instalaciones. Basado en las dependencias técnicas de la plataforma de Big Data agendé algunas actividades en paralelo y otras en secuencia, asignando en cada caso a los responsables técnicos.

Por otro lado, en cuanto a ejecución, llevé el seguimiento basado en el plan de despliegue que había preparado. Sin embargo, decidí incorporar algunas prácticas de método ágil que me parecieron útiles. En este caso fue realizar un “*scrum*” diario para verificar dependencias, obstáculos y establecer los objetivos de cada día. Al principio lo realicé de manera remota a través de llamada, una vez estando en las instalaciones del cliente fue presencial. Una vez en las instalaciones del cliente fue aún más útil incorporar ese tipo de prácticas, pues el cliente nos asignó una sala de juntas y reuní a todo el equipo a trabajar ahí lado a lado. De esta forma la interacción fue muy colaborativa y el apoyo mutuo era muy directo, esto funcionó muy bien en algunos casos cuando surgieron dificultades técnicas.

También de esta manera podía llevar un seguimiento más detallado y puntual de las cuestiones financieras, tales como los viáticos, realizando cortes por semana. Adicionalmente podía facilitar la comunicación con los ejecutivos en caso de ser necesario de su apoyo.

En cuanto al cliente, cada semana les daba un estatus de manera presencial. Además, tenía otras varias reuniones con ellos para revisar y acordar temas relacionados al soporte que se daría una vez listos los ambientes.

## 5.2 Soporte

En cuanto a las actividades posteriores a la puesta en marcha de las aplicaciones, fue en gran parte soporte a la plataforma de Big Data, de hecho, era el alcance original. Debía asegurarme de cumplir con los requerimientos y mantener operativas las herramientas. Para esto hice planes a mediano y largo plazo, identificando las actualizaciones y los cambios importantes que podrían afectar a la plataforma. Por ejemplo, las fechas de fin de soporte, defectos identificados y vulnerabilidades. En estas actividades macro también adopté el modelo clásico, pues eran cuestiones de meses. Sin embargo, cada semana teníamos una llamada con el cliente para poder revisar esos temas.

No obstante, algunas circunstancias especiales se presentaron y requirieron de varios cambios antes de lo esperado. Por un lado, el cliente solicitaba una funcionalidad nueva y argumentaba que estaba limitado, por otro, ciertos defectos se encontraron en algunas herramientas.

Es así que durante los primeros meses de soporte se necesitaron varias actualizaciones en algunas herramientas de la plataforma. Además, estos cambios debían reflejarse en ambos ambientes, es de entenderse que el cliente solicitaba los cambios primero en el ambiente de desarrollo y después en el de producción.

En estos casos también preparé los cambios adoptando un modelo clásico a manera de una iteración de despliegue, haciendo la planeación de ambas iteraciones y acordando fechas con el cliente. La primera iteración fue para el ambiente de desarrollo, la segunda para producción. Una vez completado y verificado el ambiente por el cliente, se agendaba la segunda iteración para hacer los cambios en producción.

De esta forma hice la planeación, ejecución y seguimiento de los cambios, actualizaciones y aplicación de correcciones (*fixes*) que se necesitaron durante las actividades normales de

soporte. De hecho, preparé la documentación necesaria para tener un proceso de cambios aceptado por el cliente.

Retos adicionales se presentaron para la plataforma de Big Data que estaba soportando mi equipo. La organización anunció que BigInsights dejaría de tener soporte a partir de junio de 2019, era necesario tomar las medidas necesarias e informar al cliente.

Una vez enterado el equipo de City of Dallas, se revisó el tema y se establecieron negociaciones para involucrar más gente de la organización y preparar los cambios. En este caso implicaba hacer una migración completa de la plataforma. Se involucraron arquitectos y un administrador de proyectos complejos. Junto con ellos se realizaron talleres de Design Thinking para identificar las mejores opciones para la plataforma de Big Data de City of Dallas, en mi caso participé de manera remota.

Como resultado de los talleres de Design Thinking se identificó que lo recomendado era hacer la migración, crecer la plataforma, agregar más componentes de inteligencia artificial y aumentar el hardware. Desafortunadamente el cliente tenía limitaciones de presupuesto y en ese momento solamente se podría hacer la migración.

De esta manera trabajé con el administrador de proyectos complejos para elaborar el plan de la migración, nuevamente se decidió por el modelo clásico por facilitar una estructura muy ordenada. Esta vez sería al principio secuencial, pues no afectaba a todos los componentes. Sin embargo, el mayor impacto era en el “centro” de la plataforma, migrando de BigInsights a HDP. Para esto se involucró nuevamente a un experto de Lab Services.

Por razones de costos, el administrador de proyectos complejos ya no pudo participar y toda la ejecución de la migración debí tomarla yo. Decidí aplicar la estrategia que usé en el despliegue e incorporé prácticas de la metodología ágil una vez más. Sin embargo, en esta ocasión todos estaríamos trabajando remotos, de esta forma los “*scrums*” fueron a través de llamadas al inicio del día. El trabajo colaborativo “lado a lado” fue a través de largas sesiones webex donde se compartía pantalla y todos los especialistas involucrados participaban. La estrategia funcionó muy bien y se terminó en tiempo y forma la migración.

Un par de meses después el cliente pidió actualizar la plataforma a las últimas versiones disponibles de las herramientas, en particular a HDP 3.x. De esta manera nuevamente se



necesitaba una migración. Sin embargo, en esta ocasión más cambios eran necesarios, el impacto era mayor y todas las herramientas debían ser actualizadas.

En esta segunda migración toda la parte de planeación residió en mí y toda la parte de ejecución fue efectuada por mi equipo de especialistas. Esta vez no había dinero adicional para involucrar a Lab Services o más especialistas.

Revisé todas las dependencias con mi equipo, pues todas las herramientas debían ser actualizadas, pero debían ser compatibles. Es decir, no necesariamente debía actualizarse a la última versión en cada caso. Para esta documentación utilicé un archivo de inventarios que había generado para las actividades normales de soporte.

Con toda la información recolectada elaboré un plan de proyecto que cubría todas las actualizaciones y migraciones. Nuevamente la idea fue utilizar el concepto de iteraciones en despliegue. En cada caso el resultado de la iteración era terminar con el componente actualizado en cada caso. De esta forma incluí varias iteraciones En el mismo plan, incluyendo los ambientes de desarrollo y producción. Sin embargo, había que agregar una complejidad adicional, el hypervisor de VMware y el sistema operativo de Linux debían ser actualizados también. Los agregué a manera de iteraciones previas y coordiné todo con los responsables correspondientes.

Nuevamente incorporé las prácticas ágiles, pero esta vez involucrando equipos adicionales al que yo lideraba. Fue así que coordiné también las actividades de actualización para hypervisor y sistema operativo. De esta forma agendé una llamada diaria con los otros equipos involucrados para revisar los avances, pendientes y obstáculos (*stand up*); además se documentaba mandando reportes de avance todos los días.

En cuanto a las actividades de mi equipo, seguí con las prácticas de tener el “scrum” por llamada todas las mañanas al empezar el día, llevar las actividades propias de tipo colaborativo por medio del webex.

De esta forma fue posible concluir la segunda migración de manera satisfactoria para el cliente.

Los detalles y resultados de la metodología elegida se mencionarán en los capítulos siguientes.

## Capítulo 6 - City of Dallas Big Data Platform - Participación profesional

En este capítulo se describen más detalles de mi participación en el proyecto de City of Dallas. También se menciona información técnica adicional y los retos que se enfrentaron para llevar a buen término las actividades durante diferentes etapas del proyecto (despliegue, soporte y migraciones).

### 6.1 Inicios como Delivery Manager e incorporación al proyecto

Al convertirme en Delivery Manager mis actividades fueron muy distintas a lo que tenía por costumbre. Pasé de hacer análisis y actividades meramente técnicas a llevar tareas puramente administrativas. Sin embargo, mi experiencia técnica me ayudó mucho a entender la interacción de las herramientas y componentes involucrados; incluso me sirvió para elaborar documentos de arquitectura y a revisar que los documentos de soporte tuvieran la información necesaria.

Este proyecto se relacionó a dar soporte a dos ambientes con herramientas especializadas en Big Data. Estos ambientes son los siguientes:

- **Desarrollo.** En estos servidores los especialistas de City of Dallas desarrollan sus consultas, estadísticas, artefactos, reportes y por supuesto hacen la ingestión de datos desde sus sistemas internos.
- **Producción.** En este ambiente los especialistas despliegan su sistema de Big Data en tiempo real para aprovechar la gran cantidad de información recabada por el gobierno de la ciudad de Dallas, esto en relación a las revisiones que hacen a los restaurantes y proveedores de servicios de alimentos de la ciudad. Además, llevan estadísticas sobre incidentes de tránsito locales.

## 6.2 Herramientas

Las herramientas que se utilizan en el proyecto se indican a continuación:

**Big Insights®** (después HDP). Esta fue una solución muy completa de Big Data, lista para ambientes de tipo empresarial, capaz de funcionar en distintos tipos de infraestructuras. Estaba basada para trabajar en clústeres con Hadoop y Apache Spark. Útil para escalar ambientes de manera rápida y fácil.

**BigIntegrate.** Es una Solución de Big Data que provee conectividad superior, rápida transformación y confiabilidad. Cuenta con características de entrega de datos con facilidad de uso que se ejecutan sobre nodos de datos en clústeres de Hadoop. Provee una plataforma flexible y escalable para transformar e integrar datos de Hadoop.

\*Hadoop es un proyecto de software abierto de Apache que permite el proceso distribuido de grandes conjuntos de datos en clúster de servidores.

**SPSS®.** Es una plataforma que ofrece análisis estadístico avanzado, una vasta biblioteca de algoritmos de aprendizaje de máquina (inteligencia artificial), análisis de texto, extensibilidad de tipo código abierto, integración con Big Data y despliegue continuo a aplicaciones. Es una plataforma fácil de usar, flexible y escalable.

**Cognos Analytics.** Es una suite integral de inteligencia de negocio desplegada en web. Esta provee un conjunto de herramientas para reporte, analítica, así como monitoreo de eventos y de métricas. Puede tomar información de varios tipos de fuentes de datos. Es muy útil puesto que en esta herramienta se crean los tableros de información (*dashboards*) que se presentan a los usuarios de negocio de City of Dallas.

## 6.3 Inicio de Actividades

Mi participación en el proyecto comenzó integrándome a llamadas y listas de correo, fui involucrado por una especialista para venta de soluciones en Estados Unidos. Ella también había participado durante la oferta del servicio y tenía conocimiento de los documentos de entendimiento (DOU) y de la solicitud de trabajo (SOW). Además, tenía el rol de arquitecta de soluciones dentro de AppOps aunque ella no fue quien diseñó la solución.

Al comienzo conté con el apoyo de la líder global de Delivery Managers, una persona de Brasil. Ella me fue presentando con las personas involucradas en el proyecto y me otorgó los accesos a sitios internos y a la documentación estándar para los proyectos desde el enfoque de entrega del servicio (Delivery).

Me contactaron también con el líder técnico de nuestro equipo, un especialista de Data Stage (una solución de integración de datos de la organización) quien viajaría en pocos días a Dallas para que le hicieran transferencia de conocimiento. Esto se realizaría mientras se ejecutaba el despliegue de uno de los ambientes. La instalación del ambiente de desarrollo se llevaría a cabo por parte de dos equipos de la organización:

- Lab Services. Especializados en Cognos, SPSS, Cognitive y otros servicios.
- Stampede Group. Especialistas en BigIntegrate, BigInsights y ciencia de datos.

De forma adicional y por contrato, estos especialistas trabajarían con los analistas de datos de City of Dallas (cliente) en una prueba de concepto, de esta manera el cliente esperaba involucrar a sus especialistas en las primeras labores con las herramientas. El equipo de City of Dallas no tenía mucha experiencia con las tecnologías de Big Data de la organización.

De acuerdo al contrato, la organización se haría cargo de la instalación y despliegue, así como de asesoría para un caso de uso determinado por los analistas de datos de Dallas, posteriormente del soporte a la plataforma. Por otro lado, en cuanto al hardware, este sería proporcionado como servicio por otra compañía. La otra compañía era un asociado de negocios de la organización. El asociado de negocios suministraría los servidores virtualizados y se haría cargo de los respaldos a través de una solución de VMware

previamente contratada por el cliente. Además, proporcionaría el soporte de los servidores a nivel de sistema operativo.

Aun cuando la organización es un productor de hardware, por practicidad decidieron involucrar al asociado de negocios; esa compañía es un proveedor de hardware local y está bien establecido en Dallas. En circunstancias de este tipo es conveniente trabajar con el asociado de negocios local en vez de mover recursos desde alguna otra ciudad e incrementar costos para el cliente.

Este proyecto comenzó en febrero de 2017, varios meses antes de que yo fuera involucrado. Al principio participaron especialistas para ofrecer una solución de acuerdo a las necesidades del cliente. Se realizó un análisis y levantamiento de requerimientos de tipo iterativo por parte de especialistas de GBS; los arquitectos tuvieron varias reuniones con el cliente y basados en eso se dieron las iteraciones pasando por análisis de requerimientos y diseño de la solución cada vez. Se refinaba la solución de acuerdo a las necesidades de City of Dallas, pero también se re dimensionaba de acuerdo al presupuesto que tenía planeado el cliente para esta plataforma de Big Data. La solución que cumplía con todos los requerimientos identificados del cliente era bastante robusta, pues incluía componentes para análisis masivos de información, automatización, administración de proyectos y de código.

Las herramientas adicionales eran:

- **Rational Team Concert.** Administración de proyectos y administración de la configuración
- **Rational Test Workbench.** Pruebas automatizadas.
- **Intelligent Video Analytics.** Una herramienta de análisis de video que manda alertas en tiempo real sobre actividades en videos de seguridad.
- **InfoSphere Big Match.** Herramienta que ayuda a analizar cantidades masivas de información en clústeres de Hadoop.

Sin embargo, el cliente pidió reducir costos. Al final, después de otra iteración se definió la solución (reducida) que estaba acorde con las limitantes de presupuesto del cliente.

Al tratarse de productos de software completamente terminados y “de caja”, la fase siguiente en la iteración era el despliegue. De esta manera se definieron las actividades para hacer el despliegue en el ambiente de desarrollo y posteriormente en el de producción.

## 6.4 Instalación del Ambiente de Desarrollo

Mi incorporación al proyecto se dio a unos días de comenzar la instalación del ambiente de desarrollo como parte de la fase de despliegue. Por esta razón me tocó entonces hacer las primeras acciones administrativas y gestionar el viaje del líder técnico. Como mencioné anteriormente, él estaría directamente en las instalaciones del cliente adquiriendo el conocimiento del ambiente por parte de los especialistas de Lab Services y Stampede group. De acuerdo a lo planeado serían 3 semanas.

Además, me contactaron con el administrador del proyecto, también era una persona de Estados Unidos quien además fue el arquitecto del proyecto. De parte de él comencé a recibir información relacionada al plan de trabajo y de la gente involucrada por parte del cliente. Además, me incluyó en las juntas semanales por teléfono con el cliente, en estas se planteaba el estado y avance del proyecto, así como las inquietudes del cliente. En realidad, se trataba de una junta semanal de estatus que ha seguido durante todo el proyecto por casi 3 años.

Al principio me dediqué básicamente a escuchar y comprender el avance del proyecto, identificar a los involucrados, conocer el alcance y los objetivos del proyecto. Tomaba muchas notas, preguntaba de manera interna al equipo y pedía documentación adicional cuando esta era generada. Se trataba de hacer una transición desde el equipo a cargo del despliegue hacia el equipo de soporte, del cual yo sería el líder como Delivery Manager.

Como se mencionó previamente, la organización se haría cargo de dar soporte especializado en los dos ambientes a las herramientas. Sin embargo, adicionalmente se instalaría una solución de seguridad para monitoreo de los accesos y administración de información. Esta suite es llamada Guardium. Aun cuando también se trataba de un producto de la organización, era otro equipo quien se encargaría de la instalación, configuración y soporte. Esta herramienta es parte de una suite de seguridad de datos de la organización. Es útil para

configurar políticas de riesgo e identificar intentos de acceso inválidos; así como para registrar el tipo de actividades que realizan los usuarios que trabajan en los clústeres. Por otro lado, tal como se indicó, el asociado de negocios proporcionaría el hardware. En total se trataba de 28 servidores Linux Red Hat Empresariales. Este mismo proveedor proporcionaría el soporte a nivel sistema operativo y gestionaría toda interacción con Red Hat en caso de requerir soporte más especializado.

De esta forma tendríamos dependencia en cuestiones de infraestructura con el asociado de negocios y por otro lado con el departamento de seguridad y redes de City of Dallas.

Es así que el soporte a los ambientes de Big Data para City of Dallas fue dividido como se indica gráficamente en la figura 13.



Figura 13 Equipos a cargo del soporte para los ambientes de City of Dallas

Cada equipo era responsable de tareas específicas. En el caso de AppOps el alcance se enfocaba en dar soporte a las herramientas de Big Data mencionadas con anterioridad.

Varias actividades tuvieron que ser llevadas en paralelo. Una de ellas fue formar el equipo de especialistas que estaría dando soporte a las herramientas de forma remota. Revisé currículos de algunos postulantes y pedí opiniones en cada caso al líder técnico. Como los especialistas propios de Big Data no eran tan abundantes, entre los aspectos que se consideraron aceptables estaban:

- Conocimientos avanzados de Linux
- Experiencia en Data Stage, Business Intelligence o DBMS
- Un perfil de administrador de sistemas era deseable
- Un perfil de desarrollador también era aceptable
- Ubicación en Estados Unidos o facilidad de radicar en ese país en muy corto tiempo
- Que el costo por hora se adecuara al presupuesto que se tenía en la oferta

Al final seleccionamos a dos personas de Lansing Michigan para estar en tiempo completo. Por tratarse de un cliente gubernamental, los especialistas que estarían accediendo continuamente a los servidores debían estar ubicados físicamente en Estados Unidos.

El líder técnico tenía acceso al ambiente de desarrollo por medio de una VPN que le proporcionaron durante su estancia en Dallas City Hall (el centro de gobierno de la ciudad de Dallas). Sin embargo, necesitaríamos acceso para todos los miembros del equipo. Contacté a gente de parte del cliente y acordamos establecer comunicación a través de un “*Jump Server*”. Un *Jump Server* es un sistema dentro de una red que se utiliza normalmente para acceder y administrar dispositivos en una zona de seguridad separada. Para el caso particular de este proyecto, el *Jump Server* sería un servidor tipo máquina virtual y en realidad ayudaría a conectar la red interna de la organización con la red local de City of Dallas. De esta forma el equipo de AppOps podría acceder remotamente a los sistemas del cliente. El servidor sería proporcionado por la organización y la conexión se llevaría a cabo a través de un túnel VPN a la red de City of Dallas. De esta manera los especialistas podrían acceder con cuentas propias creadas dentro del *Jump Server* a los servidores de Dallas. Este servidor sería creado y administrado por AppOps.



Para revisar los detalles técnicos de la comunicación con City Of Dallas fue necesario contactar al líder de infraestructura de AppOps, a especialistas de seguridad y gente del equipo de redes. En conjunto determinamos el centro de datos adecuado para albergar al *Jump Server* y el tipo de zona de red necesario para nuestros requerimientos. La organización cuenta con varias zonas de red, algunas se pueden comunicar al exterior y otras están altamente restringidas. Para este cliente necesitábamos una zona de red “mixta”; se ubicó un centro de datos en Atlanta que contaba por un lado con comunicación hacia la red interna de la organización, pero por el otro era capaz de comunicarse al exterior. Además, geográficamente se encontraba relativamente cerca de Dallas.

Procedí a crear el servidor virtualizado en la zona de red acordada, con los recursos de hardware necesarios y a realizar los procedimientos internos estándar y adecuados al proyecto. Lo siguiente era configurar la comunicación del *Jump Server* con la red de City of Dallas.

Conseguí los contactos de gente del equipo de seguridad y de redes de Dallas. Al principio nos comunicamos por correo electrónico, pero después agende llamadas telefónicas involucrando a gente del equipo de redes de AppOps y al equipo de redes e infraestructura de City of Dallas. Prácticamente reuní a nuestro equipo de redes con especialistas de redes de Dallas en videoconferencia, esto para que entre especialistas de redes el entendimiento fuera más claro y directo.

Después de una semana de llamadas, de revisión técnica y de sesiones con audio y video en teleconferencia, fue posible completar la configuración del túnel VPN. Entonces tuvimos el *Jump Server* listo para acceder al ambiente de Big Data de Dallas en forma remota.

Se proporcionaron cursos obligatorios para los especialistas poco después de que el líder técnico había tomado la transferencia de conocimiento. Por mi parte también tomé el entrenamiento. Aunque mi rol no sería técnico en este proyecto, deseaba entender cómo trabajaban las herramientas.

El entrenamiento básico se componía de varios cursos de la Big Data University y de la organización, incluía:

- Big Data Fundamentals (3 cursos)
- Hadoop Fundamentals (4 cursos)
- BigInsights Foundations
- Cognos

Adicionalmente pedí la documentación a los especialistas de Stampede y Lab Services, pues esa información ayudaría al equipo en las futuras labores de soporte. Les solicité elaborar los documentos con la información técnica necesaria para el soporte remoto (*Runbooks*).

Después de 3 semanas se me proporcionó parte de la documentación. Al regresar el líder técnico de Dallas tuve varias conversaciones con él a través de chat y teléfono, ya que habitaba en una ciudad diferente de México (Torreón Coahuila). El líder técnico se incorporó de manera más activa en el entrenamiento de los especialistas. Para que el conocimiento pasará a ser más práctico se creó un ambiente de pruebas con servidores internos. Ahí fue donde se instalaron las mismas herramientas que se desplegaron con el cliente. Este ambiente estuvo activo desde agosto de 2017 y siguió trabajando de manera intermitente hasta 2020, fue útil para probar ciertas actualizaciones y nuevas correcciones en las herramientas (*fixes*).

En este caso se crearon también servidores Linux, todos en la red interna de la organización. Se instaló un clúster reducido, con menos nodos y menos hardware también. En este ambiente solo se probaría hacer instalaciones, aplicar *fixes* y actualizaciones. Las tablas del clúster se crearon con lo básico y no se inyectaron datos, ya que no se pretendía generar cargas de trabajo de tipo ciencia de datos.

Como se puede observar en la figura 14, inicialmente se crearon 4 nodos de administración y 3 nodos de datos para la parte principal (*core*) de Big Data; este fue inicialmente BigInsights y después HDP. Además, se creó un server para instalar SPSS y otro para instalar Cognos. Las instalaciones iniciales se hicieron con exactamente las mismas versiones que tenía City of Dallas en su ambiente de desarrollo.

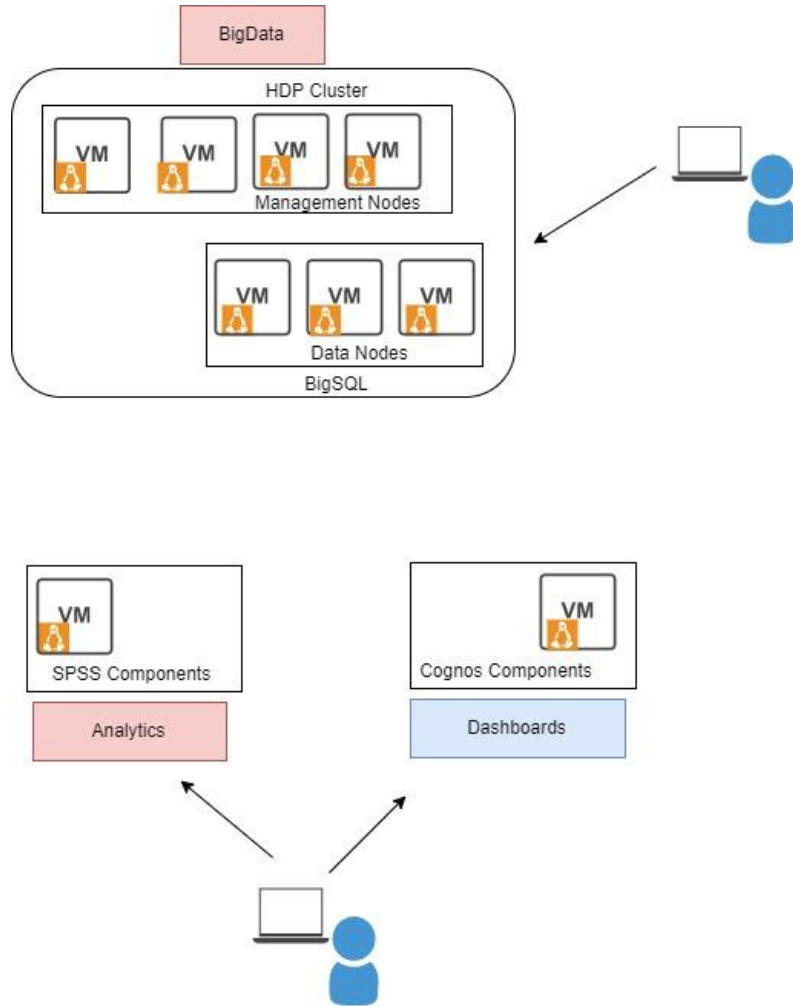


Figura 14 Ambiente interno para pruebas

Se realizó un plan de transferencia para hacer la entrega del ambiente de desarrollo desde Stampede y Lab Services hacía al equipo que yo estaría liderando. Insistí en pedir la documentación ya que por las agendas tan apretadas de algunos especialistas de Stampede group había documentos faltantes de completarse.

Entre fines de agosto y principios de septiembre de 2017 el cliente hacía muy poco uso del ambiente de desarrollo. De hecho, se concentraron más en detallar los requerimientos con Stampede group para un caso de uso que pensaban llevar a cabo. La gente de City of Dallas estaba muy interesada en implementar por completo la funcionalidad de las herramientas. Sin embargo, tenían conocimientos muy básicos sobre las mismas; por esta razón insistían en tener un acompañamiento completo por parte de los especialistas de la organización en Big Data.

Debido a sus fechas compromiso internas, el equipo del cliente insistía mucho en obtener guía y asesoría de los especialistas de Stampede. Los especialistas de City of Dallas querían tener todo el apoyo posible y gente dedicada a resolver sus dudas. Si bien se brindaba el servicio de acuerdo a lo estipulado en el contrato, por parte del cliente se pedía más y más. Esto debido al hecho de tener la implementación de las herramientas como una de sus metas para el año 2017.

## 6.5 Instalación del Ambiente de Producción

Me mandaron un plan de trabajo actualizado donde se indicaba que el ambiente de producción debía ser instalado en el mes de octubre. Este plan había sido elaborado por el arquitecto de la solución quien era además el administrador de proyecto. Adicionalmente se me hizo la transferencia de la mayoría de las actividades de administración del proyecto por ser yo el encargado del soporte. También se me dio más información técnica con la cual hice algunos de los documentos técnicos relacionados a la aplicación y los diagramas de arquitectura.

Para mediados de agosto el tema a revisar era quién se encargaría de hacer la instalación del ambiente de producción. El candidato natural era Stampede group, ya que este equipo se hizo cargo de las actividades del ambiente de desarrollo. Además, eran los que tenían más experiencia con las herramientas de Big Data.

Sin embargo, por cuestiones de costos se hizo notar que no era muy viable involucrar a Stampede group, pues el precio por hora de sus especialistas era bastante elevado. Además de que ellos seguían trabajando con la gente de City of Dallas en el caso de uso para el ambiente de desarrollo.

A mediados de septiembre de 2017 me contactó la líder de soluciones encargada de la parte de cierre de ventas y costos para este proyecto. Ella organizó una reunión con la ejecutiva del proyecto, una persona de Atlanta. Durante la llamada me preguntaron sobre el tiempo que tardaría mi equipo en instalar el ambiente de producción, así como de las limitaciones y riesgos que habría que considerar. Adicionalmente me pidieron elaborar un plan de trabajo detallado para las actividades relacionadas a la instalación del ambiente de producción, el tiempo contemplado era de 3 semanas. Prácticamente lo mismo que le tomó a Stampede group y a Lab Services instalar el ambiente de desarrollo.

Mi respuesta fue que el tiempo contemplado era corto, resalté el hecho de que Stampede group – equipo de expertos en Big Data con mucha experiencia - había tardado justo esos días para las actividades. Hice notar que, si bien mi equipo había tomado entrenamiento y se había auto capacitado, aún no contaban con una experiencia comparable a Stampede Group.

La ejecutiva del proyecto me pidió ir revisando lo necesario en caso de realizar el viaje a Dallas. Organicé una llamada con mi equipo y les informé de la posibilidad de estar trabajando 3 semanas en las instalaciones del cliente. Esto era importante ya que implicaba un viaje medianamente largo para todos los involucrados. Todo esto sucedió entre el 29 de septiembre y el 3 de octubre.

Para el día 5 se confirmó que nosotros estaríamos a cargo de la instalación del ambiente de producción. De hecho, la encargada de la parte económica y la ejecutiva del proyecto me indicaron el presupuesto, resaltaron la importancia de no sobrepasar el monto y ser muy cuidadosos puesto que económicamente el proyecto estaba muy ajustado. Incluso se involucró a gente de finanzas y a personas con roles de liderazgo a nivel local y global tales como líderes de analítica y *cognitive* del área sur de Estados Unidos, el ejecutivo global para proyectos complejos de AppOps, la ejecutiva para clientes del sector público de Texas, entre otros. Todos ellos querían dar seguimiento al proyecto, por esta razón nuestro trabajo tendría mucha visibilidad en AppOps y en otros equipos dentro de GBS. Esto implicaba un reto

adicional, pues usualmente el rol de Delivery Manager en AppOps no involucra cuestiones financieras.

Avisé a mi equipo, les solicité comenzar a buscar vuelos y hacer reservaciones. Esto indicándoles ubicaciones recomendadas para no tener que rentar auto y desde entonces comenzar a evitar sobrecostos. Había varios hoteles cercanos al Dallas City Hall, a un par de cuadras, nos hospedamos en uno de ellos.

En paralelo gestionaba la logística con el cliente sobre nuestros accesos, el lugar donde nos ubicaríamos dentro de sus instalaciones, los contactos. Adicionalmente revisaba con ellos los detalles del plan de trabajo. En realidad, el rol que estaría llevando a cabo para esas actividades era el de Deployment Manager. Así que mi intención era revisar cada detalle alrededor de las labores de mi grupo de trabajo.

De manera interna pedí a mi equipo conectarse remotamente a Dallas y comenzar a descargar el software hacia los servidores de producción, también les pedí hacer revisiones técnicas a nivel sistema operativo para asegurar que la configuración estuviera lo más avanzada posible incluso antes de viajar. Estas revisiones incluían verificar que existiera comunicación hacia los sitios de la organización para descargar el software, revisar que no hubiera parches de sistema operativo pendientes, que el *firewall* no bloqueara la comunicación entre los servidores y que cumplieran con los requerimientos de sistema para las instalaciones. Además, les solicité revisar la disponibilidad de los repositorios de Red Hat para que en caso de ser necesario se pidiera apoyo al equipo encargado del soporte de hardware y sistema operativo.

El equipo hizo las reservaciones correspondientes, cada quien de manera personal. Serían 2 personas de Estados Unidos, en específico de Lansing Michigan, una persona de Sao Paulo en Brasil, el líder técnico de Torreón y yo de la Ciudad de México. Partiendo de países, ciudades y zonas horarias diferentes no era posible llegar exactamente en la misma fecha y hora. Además, por el corto tiempo – reservamos 4 días antes de viajar – había pocos vuelos y poco hospedaje.

Los primeros en llegar fueron los especialistas de Lansing, el 9 de octubre de 2017. Después llegó el Líder técnico el 10 por la mañana, luego yo, el mismo 10 por la tarde. Finalmente, la persona de Brasil llegó el 12 de octubre al mediodía.

Les pedí buscar a su llegada a nuestro contacto en City of Dallas, la persona que nos apoyó con la VPN y quien tramitaría nuestros accesos.

Como lo mencioné, las personas de Lansing llegaron desde el primer día, obtuvieron sus accesos muy rápidamente y de hecho tuvieron una reunión con la ejecutiva del proyecto, quien por llevar la parte comercial y la relación más cercana con el cliente estaba muy interesada en conocer al equipo.

El segundo día llegó el líder técnico por la mañana a quien encargué asistir a la primera junta con el cliente. En mi caso llegué por la tarde y de inmediato busqué a nuestro contacto para presentarme personalmente. De hecho, hice lo mismo incluso con mi equipo pues el trato con ellos había sido totalmente remoto hasta entonces.

La misma tarde del segundo día pedí el estatus a cada uno de los miembros del equipo y les solicité ir avanzando todo lo posible e intentar adelantar algunas actividades ese mismo día, para empezar las actividades de pre instalación de BigInsights. Mis intenciones eran tratar de adelantar ciertas tareas para completar un poco más del trabajo planeado para cada día, pues así tendríamos cierta holgura en caso de encontrar contratiempos durante las actividades.

Comencé de inmediato a aplicar las técnicas de administración de proyectos de ingeniería de software (manejo de riesgos, administración de personal y trabajo de equipo), vigilando cuidadosamente los costos. Solo que en este caso las prácticas se enfocaban básicamente a despliegue, pues la parte de desarrollo no aplicaba al tratarse de software instalado “de caja”.

Al tercer día asistí por primera vez de manera presencial a la junta semanal con el equipo del cliente. Tenían muchas preguntas sobre el estatus y sobre las actividades, como se mencionó tenían fechas compromiso. Había bastante presión y solicitaban saber cómo se compensaría lo que consideraban retrasos en la implementación de las herramientas, ya que ellos hubieran deseado que empezáramos una semana antes. Por estas razones querían confirmar que teníamos la capacidad para salir en la fecha indicada. Fue una reunión llena de preguntas y cuestionamientos que colaboró a aumentar la presión y los “ojos” sobre nuestro equipo.

Para el cuarto día llegó el último miembro del equipo desde Brasil, trabajé para integrarlo lo más pronto posible y así fue.

Pasó la primera semana y llevábamos buen avance de acuerdo al plan, de hecho, incluso habíamos completado tareas que estaban planeadas para la segunda semana de actividades. Ya se habían instalado el clúster de BigInsights, Cognos y 2 de los 3 componentes de SPSS; este último estaba contemplado para la segunda semana. Como lo he mencionado desde el inicio, traté de adelantar trabajo con el equipo para tener cierta holgura de tiempo y atenuar los riesgos. Se notaban esas horas adicionales con las que contaríamos en caso de ser necesarias.

Sin embargo, más actividades se agregaron a mi carga de trabajo debido a que se me pidió llevar el control de los gastos de manera más detallada de lo esperado; pues debería llevar el rastreo financiero de todo el equipo para asegurar que no nos pasáramos del presupuesto. De hecho, también tenía que dar seguimiento a las horas y costos registrados a nivel gerencial, todo esto no estaba contemplado en mi rol original. Es decir, se me estaba pidiendo tomar las actividades completas de la administración del proyecto.

Para la segunda semana (16 de octubre) me pedían estatus de parte del cliente, así como del equipo ejecutivo y financiero. Todo iba en tiempo hasta el cuarto día de la segunda semana cuando aparecieron unos problemas en la configuración. La arquitectura de producción no era exactamente igual que la de desarrollo. El clúster de BigInsights tenía más nodos de administración y eso causó un cambio en la configuración que no era conocido en el equipo. Como la instalación se hacía en modo clúster, un nodo adicional generaba adecuaciones más allá de ese solo servidor. Cada nodo tenía varios servicios y había que configurar adecuadamente para poder balancear la carga de trabajo.

En este caso les pedí a los especialistas abrir un ticket directamente con el soporte de las herramientas a nivel global (PMR), para que de ser necesario pudiéramos ser atendidos por el equipo de desarrollo de esas herramientas. Esta situación la tuvimos con 2 de las cuatro herramientas incluidas en el clúster, BigInsights y SPSS. Al final de la semana la situación había cambiado y en lugar de ir adelantados estábamos justo en tiempo y con dos problemas de alto impacto sin resolver. Aun cuando estaban instaladas las herramientas, algunos servicios no se podían poner en línea (nodos de BigInsights y Analytic Server). Establecí



esas situaciones como riesgos y lo comuniqué al equipo interno. Había preocupación por la visibilidad del proyecto, como lo mencioné tanto los ejecutivos de la organización como los del cliente querían tener estatus de lo que sucedía. El equipo seguía trabajando.

Los problemas tardaron en ser resueltos y para mediados de la tercera semana ya incluso teníamos retraso de 2 días, la situación se había revertido por completo. El trabajo de equipo era crucial para solventar los retos que estábamos enfrentando. Adicionalmente salió un problema de desempeño a nivel sistema operativo para lo cual se abrió otro PMR. Aparecía un mensaje de *CPU lock* en los servidores que ejecutaban el servicio de BigSQL. Esta situación causaba que el consumo de CPU en los servidores llegara casi al 100% en la mayoría de los nodos del clúster. Era algo grave ya que en ese momento las herramientas no tenían carga de trabajo. Es decir, tan solo con levantar los servicios los procesadores se estaban saturando, no era normal.

En esos días algunos miembros del equipo del cliente nos pasaban a visitar con cierta frecuencia y se daban cuenta de la ardua labor del equipo para poder completar el ambiente. Eso ayudó a mejorar la comunicación con el cliente. Sin embargo, también notaban que había cosas que estaban tomando más de lo esperado. Había decidido que si para el cuarto día de la tercera semana (26 de octubre) al final del día no quedaban resueltos los problemas, informaría de manera formal del retraso, pues ya no habría manera de salir en el tiempo esperado. Esto era parte de la administración de los riesgos que tenía identificados.

Si no era posible resolver los problemas tendríamos un fuerte impacto en varios aspectos, desde la percepción del cliente, las fechas de entrega se retrasarían y pondríamos en riesgo sus objetivos para el año. El cliente monitoreaba continuamente el avance y pedía estatus frecuentemente. También tendríamos un impacto significativo en las finanzas planeadas para el despliegue, pues eso implicaba estar más días en Dallas aumentando el costo de hospedaje y de cambios de vuelos. De ser así el costo al finalizar sería mayor que el valor ganado del servicio.

Sin embargo, el esfuerzo colaborativo de todo el equipo dio resultado, se hizo un gran trabajo en equipo. Fue posible resolver los problemas después de muchas llamadas con los especialistas de PMR's, muchas revisiones a marchas forzadas y varios cambios. Para el mismo jueves 26 en la tarde ya estaban configuradas las herramientas faltantes. Hacia el final

de la semana el ambiente de producción estaba completo y funcional. Adicionalmente el proyecto estaba terminando en tiempo, forma y dentro del presupuesto. De hecho, sobró alrededor de un 15% del dinero que se tenía contemplado para gastos.

Poco antes de terminar también tuve reuniones con el equipo de City of Dallas para aclarar el manejo de incidentes, los procedimientos y niveles de servicio. Es decir, los detalles del servicio de soporte que proveeríamos. Al cumplirse los objetivos para el despliegue de producción nos fue posible cerrar esa fase del proyecto.

Durante esos últimos días también daba el seguimiento a la instalación de los agentes de Guardium, la solución de seguridad que monitoreaba los accesos a las herramientas de consulta y administración de datos. Aun cuando esto sería soportado por otros especialistas, me quise asegurar que las labores avanzaran de acuerdo a lo planeado.

El día 27 de octubre por la mañana tuve una última reunión con el equipo de City of Dallas en sus instalaciones. En esta junta se les hizo el anuncio formal del término de actividades de mi equipo, regresé a México ese mismo día por la tarde. Igualmente, ese día regresaron a sus ubicaciones los otros 2 especialistas que no habitaban en Estados Unidos, el líder técnico y el especialista de Brasil que trabajo con SPSS y Cognos. Informé a los especialistas de Lansing que ellos se quedarían hasta el 31 de octubre de la siguiente semana. Había decidido que estuvieran algunos días más solo para revisar en caso de que algún defecto pudiera surgir y para que el cliente se sintiera acompañado por un poco más de tiempo. Como su viaje era local su desplazamiento era más sencillo que el del resto del equipo.

Al finalizar las actividades, el cliente podía planear con seguridad sus actividades propias de negocio y enfocarse en ello. Por otro lado, de nuestra parte era posible continuar con lo planeado y enfocarnos en el soporte. De hecho, la ejecutiva del proyecto agradeció a todo el equipo el compromiso y dedicación

Una vez que todos los miembros del equipo regresamos a las ubicaciones locales en Estados Unidos, Brasil y México, le solicité a los especialistas elaborar los documentos de soporte (*Runbooks*) para el ambiente de producción. Estos artefactos los incluí en un sitio interno para contar con la información necesaria de los ambientes de desarrollo y producción de la solución.

De esta forma las iteraciones estaban concluidas y pasábamos por completo a la etapa de soporte a aplicaciones.

En la figura 15 se ilustra la implementación final del ambiente productivo:

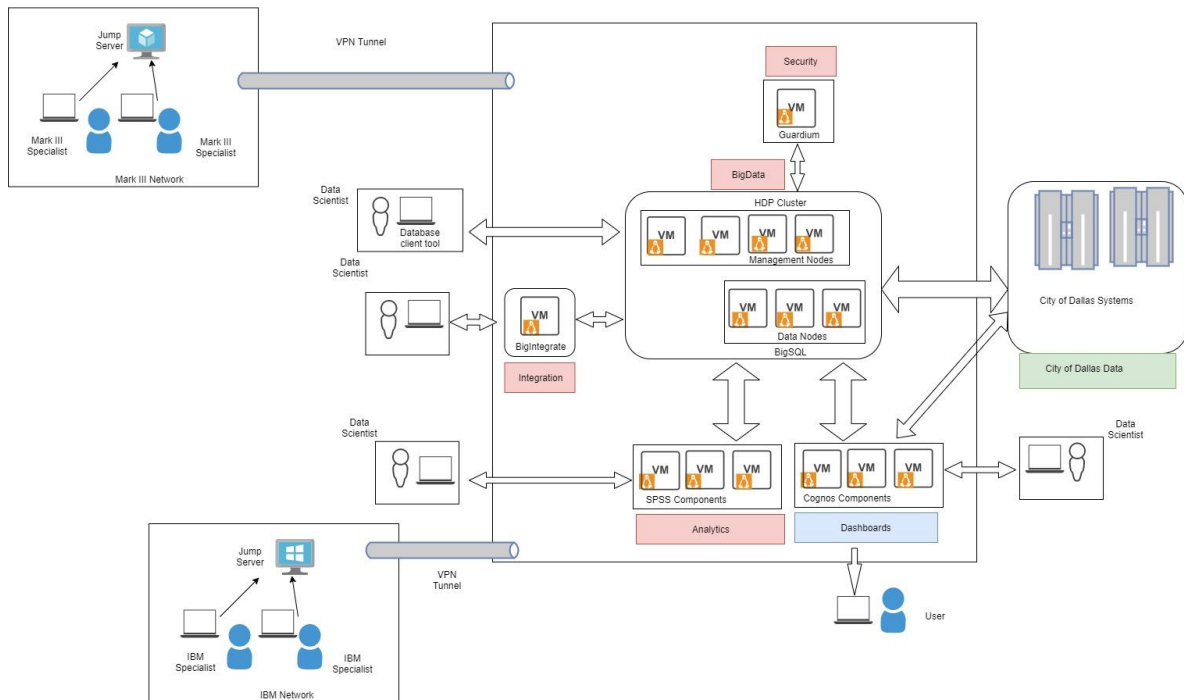


Figura 15 Arquitectura del ambiente de producción

Como puede verse en la figura 6.3, para BigInsights que es el centro del clúster se instalaron 4 nodos de administración y 3 de datos. En cada uno de los nodos se instalaron HDFS, Hive y BigSQL como servicios principales. Otros servicios secundarios como Zookeeper, Flume y sqoop solo se instalaron en algunos de los nodos.

BigIntegrate que se utilizaría para la ingestión de datos se instaló también en un solo server.

Por otro lado, en relación a Analytics, SPSS se instaló distribuido en 3 servidores. Uno para Modeler, otro para CDS y finalmente uno para Analytic Server. Cada uno de ellos desempeñaba diferentes funcionalidades que los demás.

En cuanto a lo que sería la capa de presentación, Cognos también se instaló en tres servidores, cada componente en una máquina virtual. Uno para el aplicativo, otro para Gateway y el tercero para Content Manager. De esta manera se podría distribuir la carga de trabajo y tener mejor desempeño.

La arquitectura para el ambiente de desarrollo era bastante parecida, la diferencia es que tenía solo 2 nodos de administración, es decir la mitad que el ambiente de producción.

Todas las máquinas virtuales estaban repartidas en 8 servidores físicos. Los servidores físicos estaban ubicados en las instalaciones del cliente (*on premise*). En esos servidores se instaló VMware como hypervisor y a partir de ahí se crearon las máquinas virtuales correspondientes y se asignaron recursos de hardware de acuerdo al diseño original de GBS.

## 6.6 Transición a Soporte

En la primera semana de noviembre continué dando seguimiento a las actividades de Guardium para asegurar que la parte relacionada al ambiente de producción estuviera lista. Aun cuando se trataba de otro equipo, me involucré activamente agendando llamadas con el cliente y los especialistas de Guardium. Al final también les solicité documentar el ambiente desde el enfoque de la solución de seguridad.

Pasadas unas semanas, a mediados de noviembre de 2017, comenzamos a recibir solicitudes del cliente. Al principio se trataba de cosas básicas más relacionadas a administración de usuario y configuración de permisos. Después, conforme los analistas de datos de Dallas fueron aprendiendo más de las herramientas, las solicitudes fueron más específicas.

Seguimos en llamadas semanales con el equipo de Dallas, el asociado de negocios y Stampede group. Muchas de las peticiones eran direccionadas hacia Stampede group, ya que ellos estaban asesorando a los analistas de datos en un caso de uso. La gente de Dallas lo que

pedía una y otra vez era un acompañamiento para completar un flujo completo con su información a través de todas las herramientas. Algunos de ellos se especializaban en la ingestión de información, otros en el manejo de estadísticas y unos más en la capa de presentación relacionada a los reportes.

De esta manera comenzamos en paralelo a atender solicitudes tanto en producción como en desarrollo. La mayoría de las actividades se relacionaban aún a desarrollo, ya que ahí era donde los especialistas de Dallas estaban trabajando el primer caso de uso con Stampede group.

Debido a algunos requerimientos, fue necesario actualizar parte de las herramientas como Cognos, ya que cierta funcionalidad que deseaban usar los especialistas de Dallas solo estaba disponible en versiones más recientes. De igual manera, conforme los especialistas de Dallas realizaban acciones más complejas y necesitaban funcionalidad más detallada, salieron a la vista algunas limitaciones en las herramientas. Por esta razón también fue necesario actualizar SPSS a una versión más reciente.

Durante la segunda mitad de noviembre y principios de diciembre nuevamente el cliente tenía muchas preguntas y hacía muchas solicitudes. La gente de Dallas sentía la premura de terminar sus actividades en fechas específicas. Todos los cambios que se aplicaban en desarrollo se replicaban también en producción en corto tiempo. El objetivo era tener desarrollo y producción con las mismas versiones de las herramientas.

Durante los últimos días del año 2017 nos informaron que el equipo de especialistas del cliente estaría trabajando en el primer caso de uso, esto ya directamente en el ambiente de desarrollo y con su información. Hasta entonces habían estado haciendo solo pruebas de concepto con ejemplos básicos. Ellos esperaban que nosotros estuviéramos al tanto de sus labores para brindar el soporte adecuado a la brevedad.

A mediados de diciembre de 2017 comenzamos a recibir solicitudes relacionadas a desempeño, ya que de alguna manera ciertos reportes tomaban más tiempo de lo esperado para cargarse por completo. Nuestro especialista de Cognos identificó en específico las consultas que estaban relacionadas a los reportes con lentitud y logro aislar las que más impactaban. Se ubicaron algunos parámetros que podían causar demora y se revisó el tema

con el cliente. Al parecer algunos filtros causaban el retraso, pero se corroboró que había algo más. Lo que hacían los filtros era armar consultas de BigSQL (uno de los servicios en el clúster) con base en los parámetros que se especificaran. Se identificó una consulta en particular, después de revisarla con más detalle se rastreó hasta BigSQL y fue posible ejecutarla directamente ahí antes de llegar a Cognos. Era notorio que desde BigSQL había problemas de desempeño. Revisando notas técnicas se identificó que incrementar la memoria disponible para BigSQL sería la solución adecuada en ese momento. Informé al cliente de las implicaciones y afectaciones, el equipo de City of Dallas aceptó el cambio y nuestros especialistas procedieron a aumentar la memoria disponible de BigSQL. Después de esa actualización el desempeño mejoró considerablemente.

La gente de City of Dallas siguió trabajando y continuaban mandando algunas solicitudes semanalmente, aún principalmente relacionadas a permisos o pidiendo orientación sobre algunos cambios que requerían hacer.

Debido a que el arquitecto y administrador de proyecto tenía otras actividades, se me delegaron por completo las tareas de administración a fines de 2017. Fue un acuerdo entre GBS y AppOps. Desde entonces estuve a cargo de coordinar todas las actividades de soporte de parte de la organización y del socio de negocios proveedor del hardware.

Para las primeras semanas del año 2018 recibimos una solicitud nuevamente relacionada a desempeño de reportes. Sin embargo, en esta ocasión era una cuestión diferente ya que no había mucho margen para aumentar la memoria disponible. El cliente estaba realmente preocupado, pues se acercaban fechas compromiso relacionadas a sus objetivos y sus tableros de reportes (*dashboards*) tardaban bastante en cargar, eso no era aceptable para sus usuarios. Les pedí a los especialistas de nuestro equipo abrir PMRs relacionados a Cognos, BigSQL y SPSS. Dimos seguimiento en paralelo con los desarrolladores de las herramientas mientras atendíamos otras solicitudes. Debido a la urgencia de City of Dallas, se agendaron unas reuniones telefónicas diarias para informar al cliente del estado y avance del problema de desempeño. De hecho, tenía dos llamadas con ellos durante esos días, una por la mañana y otra en la tarde. La preocupación del cliente llevó a escalar la situación y al mismo tiempo me tocaba liderar al equipo de especialistas, informar al cliente y comunicarme a niveles gerenciales y ejecutivos ya que el proyecto tenía mucha visibilidad tanto de parte de City of

Dallas como de nuestro equipo dentro de la organización. Mucha gente quería saber cómo iba el avance para resolver el problema.

Después de 4 días de análisis, los especialistas de los PMR's no habían encontrado algo que pudiera resolver el problema. La solución surgió del mismo equipo de soporte cuando el líder técnico de nuestro grupo identificó que no estaba activa una funcionalidad (*auto analyze*) que ayudaba a indexar las tablas y hacer más eficientes las consultas. Se activó la funcionalidad necesaria y el cambio fue notorio. Los reportes funcionaron con un desempeño mucho mejor y ya aceptable para el cliente. Esa situación quedó documentada y ha sido muy útil para aplicar el mismo procedimiento en otras ocasiones cuando después de parches y actualizaciones se ha repetido la lentitud en las herramientas.

La gente de Dallas pedía estar al tanto de los detalles y se les dio la información técnica del cambio efectuado. Desde ese momento ya no se presentaron más errores o problemas de desempeño. Unos días después el arquitecto de City of Dallas nos comunicó que sus especialistas de datos fueron capaces de terminar el primer caso de uso de forma completa. De esta manera fue posible cumplir con los objetivos de negocio establecidos por el cliente.

Durante los días siguientes me enfoqué en revisar los procedimientos de incidentes con el arquitecto del cliente, esto con el objetivo de depurar detalles e implementar mejoras que se adaptaran a los procesos internos de City of Dallas.

Por otro lado, continuamos revisando procedimientos para instalar parches por cuestiones propiamente de mantenimiento y de seguridad. También me enfoqué en mejorar las habilidades y conocimientos del equipo de soporte solicitando ciertos cambios con el efecto de tener gente preparada para actividades más complejas. Uno de los miembros del equipo de Lansing no había desarrollado las habilidades necesarias de manera adecuada, su conocimiento era demasiado básico. Por esta razón solicité un cambio de especialista, mismo que me concedieron unas semanas después.

Conforme transcurrieron las semanas, ya con el primer caso de uso funcionando y con más conocimiento de parte de los especialistas de Dallas, las solicitudes fueron más específicas en cuanto al uso e interacción con las herramientas. Así también los especialistas comenzaron a trabajar con cierta regularidad en otros servicios y componentes que no habían utilizado

tanto. Es así que se regularizaron las actividades, tomando ya un ritmo normal de soporte y enfocándose a buscar áreas de mejora y automatización de algunas actividades. Por otro lado, también se dio más enfoque en pulir los procesos y procedimientos que llevaríamos en conjunto con el asociado de negocio que soportaba el hardware, con Guardium y con City of Dallas.

En resumen, hasta principios de marzo de 2018 cuando se estabilizaron los ambientes, la distribución de solicitudes para soporte por herramienta era la que se puede observar en la figura 16.

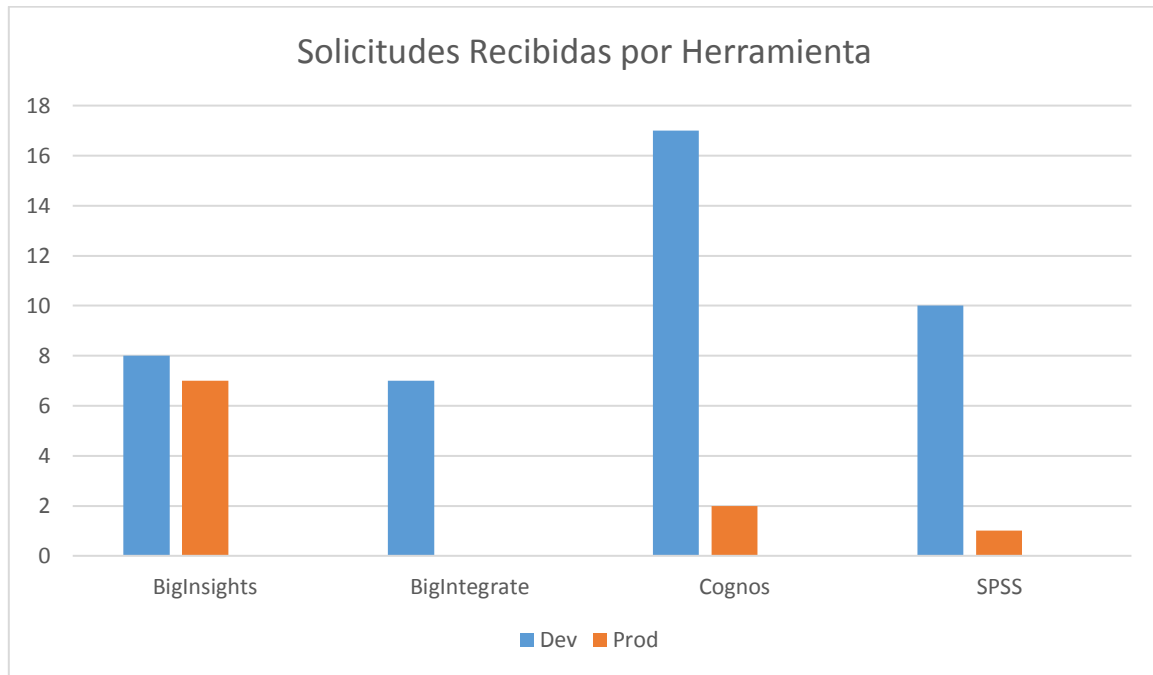


Figura 16 Solicitudes

En gran medida esas solicitudes iniciales se relacionaron a depurar los detalles que permitieron tener los ambientes estables para la operación normal. Como puede observarse hubo un mayor número para Cognos, ya que esa herramienta requirió actualizaciones de software, de configuración y *fixes* (reparaciones a través de actualizaciones). Adicionalmente es notorio que para desarrollo llegaron más solicitudes que para producción. La mayoría de



los cambios se realizaron ahí inicialmente, se aplicaron después en producción y así cuando el cliente trabajo en este último ambiente todo estaba mucho más estable.

Poco después, a fines de marzo, el equipo de City of Dallas recibió una nueva arquitecta. Durante el 2018 ellos planearon comenzar a implementar no solo Big Data a partir de información estática sino buscar opciones para trabajar con base en diferentes flujos de desarrollo (*streams*) y con tecnología de internet de las cosas (*IoT*).

Había todo un Framework que ellos estarían revisando y que a su vez nuestro equipo tendría que conocer. Este proyecto ha incluido varios retos, pero experiencias muy interesantes y enriquecedoras.

Después de marzo de 2018 los ambientes permanecieron estables por varios meses. Se recibió una media de unas 10 solicitudes cada mes, principalmente relacionadas a los servicios dentro del clúster de BigInsights, solo unos pocos para Cognos y SPSS. Sin embargo, en octubre de ese mismo año 2 servidores de producción tuvieron problemas a nivel de hardware (virtual). Las máquinas virtuales que albergaban 2 servidores de producción se apagaron sin previo aviso. Al tratarse de los nodos de administración todo el clúster se vio afectado.

Los especialistas del asociado de negocios fueron los encargados de revisar, pues se trataba de un problema de hardware. Después de más de 14 horas fueron capaces de reestablecer los servidores, se pidió hacer un análisis de causa raíz. Al pasar algunos días se determinó que la causa probable era la falta de un parche a nivel del hypervisor de VMware, por esa razón la eficiencia en alto volumen de transacciones podría no ser la óptima. De cualquier manera, se incrementó el monitoreo para evitar saturaciones en el clúster. El especialista en soporte de hardware activó notificaciones y aumentó el muestreo para que pudieran recibir alertas mucho más rápido.

Por nuestra parte se implementó una herramienta de monitoreo adicional llamada New Relic® y se instalaron agentes en cada uno de los servidores, también se configuraron alertas. De esta forma podíamos monitorear hardware, procesos, almacenamiento, red, JVM y algunos procesos de DB2.

## 6.7 Migración a HDP

Por razones comerciales la organización dejó de tener una herramienta integrada con Hadoop. Por este motivo la suite sobre la que trabajaba el clúster de City of Dallas inicialmente (BigInsights) dejaría de tener soporte a partir del 30 de junio de 2019. Comunicué al cliente esta limitante técnica desde octubre de 2018, aunque era un hecho generalmente conocido en la industria.

A partir de ese momento era claro que debía hacerse no solo una actualización a la suite, sino una migración completa al clúster. La continuidad de la suite estaría soportada en la Hortonworks Data Platform (HDP).

El cliente solicitó hacer la migración lo antes posible. Sin embargo, se trataba de una labor compleja en varios aspectos. Para empezar, mi equipo solo estaba a cargo de cambios que involucraban hasta 16 horas de trabajo continuo. La migración en sí misma tomaría una semana como mínimo. Adicionalmente, algunas de las otras herramientas debían ser actualizadas, de otra manera no serían compatibles con la nueva versión. De manera general el clúster alberga la información y la ingestión de datos se hace desde otros servicios dentro del clúster, posteriormente usan algunos de esos servicios como fuentes de datos para crear reportes y explotar la información de otras formas. Por ejemplo, tableros de reportes (*dashboards*) en Cognos. Es por eso que un cambio en el clúster impacta en todas las otras herramientas, hay mucha dependencia entre las herramientas.

Tuve que contactar a mis líderes y a algunos ejecutivos de la cuenta de Estados Unidos, la labor de migración requeriría presupuesto adicional y el soporte de otros especialistas. Principalmente solicité un arquitecto de Big Data para revisar todo el impacto que podíamos tener, además algunas herramientas tal vez requerirían más hardware para tener mejor desempeño.

Después de varias llamadas se acordó la participación de especialistas de otros equipos de la organización. Se incorporarían de manera temporal especialistas de Lab Services, arquitectos

de Big Data, algunos ejecutivos para conversar con el cliente y hasta un administrador de proyectos complejos.

Todo lo anterior sucedió entre mediados de noviembre y principios de diciembre de 2018. Para esas fechas, como lo mencioné, ya me habían delegado todas las actividades de project manager, aun cuando mi rol no era ese originalmente. Sin embargo, por motivos prácticos fue necesario que tomara esa responsabilidad.

Así pasaron los últimos meses de 2018, negociando y planeando varias actividades con respecto a la migración para principios de 2019.

En enero de 2019 se comenzaron varias actividades con el cliente. Se agendaron varios talleres basados en Design Thinking (metodología para potenciar la creatividad colectiva) para revisar una vez más los requerimientos con City of Dallas, para ese entonces el cliente ya tenía varios casos de uso y estaba utilizando bastante las herramientas. Además, tenía varios planes para los dos ambientes. En mi caso me conecté por *webex* por si necesitaban información de mi parte. La idea era identificar junto con el cliente si era recomendable hacer cambios a la plataforma de Big Data.

Los talleres se efectuaron en Dallas durante una semana. Para esto solicité que fuera un miembro de mi equipo de Lansing, para él sería más fácil el traslado por tratarse del mismo país. En los talleres además participaron especialistas de requerimientos, ejecutivos y hasta el administrador de proyectos complejos (*complex PM*) recientemente involucrado.

Fue una semana bastante positiva y se hizo un entendimiento más claro de los planes a futuro del cliente.

Posteriormente, con base en el análisis de los resultados de los talleres, se recomendó hacer adecuaciones e incluso agregar más hardware. Sin embargo, por cuestiones presupuestales del cliente, los ambientes debían permanecer con los mismos componentes por algún tiempo adicional. Al final lo positivo fue brindarle al cliente una idea más clara hacia a donde podía perfilar su negocio.

Una vez terminados los talleres y bien definidos los objetivos, se procedió a crear un plan de trabajo detallado para la migración de los ambientes, de esto se encargó en un inicio el *complex Project manager* a quien apoyé. Además, se definió una solicitud de cambio del

proyecto porque se necesitaría involucrar más especialistas y definir el alcance de la migración.

Tomó algo de tiempo negociar y aclarar algunas partes de la solicitud de cambio con el cliente. Aun cuando no tuvo costo adicional, fue necesario detallar y negociar varias cosas con City of Dallas. Adicionalmente se actualizó el plan de proyecto para la migración

Una vez aprobada la documentación requerida se decidió confirmar la participación de un especialista de Big Data de Lab services. Este compañero temporal estaría a cargo de guiar la migración desde la parte técnica y estaría trabajando muy cercanamente con mi equipo.

Por cuestiones prácticas y sobre todo tratar de reducir costos, se decidió que las actividades se efectuarían de manera remota. El especialista de Big Data desde Texas. Mi equipo desde Michigan, Brasil y México. Trabajaríamos a través de *webex* para ver todos los detalles en directo compartiendo pantallas.

A mediados de mayo de 2019 se comenzó con las actividades previas a la migración, se trataba de unas pruebas requeridas por el especialista de Big Data para determinar el estado general del ambiente (scripts y actividades que verificaban la salud del clúster). Como es práctica usual el primer ambiente sobre el que se trabajo fue en desarrollo.

A partir de este momento prácticamente me encargué por completo de las actividades de la administración del proyecto de migración, además de las otras relacionadas a la cuenta. Aun cuando se había involucrado un *complex Project Manager*, las actividades fueron delegadas hacía mí nuevamente. Procedí a negociar con el cliente las fechas, mandar boletín informando de las actividades (el clúster no estaría disponible) y agendar las reuniones por *webex*. Así comenzamos con las primeras actividades relacionadas a la migración.

En las primeras pruebas se tuvieron algunos problemas con las máquinas virtuales, al realizar el reinicio completo del hardware un par de servers no respondieron favorablemente (el *boot* no funcionaba). Después de casi 20 horas de trabajo, el soporte de hardware fue capaz de reestablecer los servidores que presentaban problemas. Una vez listas las máquinas virtuales, procedimos a levantar las aplicaciones con mi equipo. Al terminar estas actividades se determinó que se podía continuar con la migración en desarrollo.

El comienzo propiamente de la migración en desarrollo fue en la última semana de mayo de 2019.

Para esto agendé varias sesiones diarias de *webex*:

- Una interna con el equipo encargado de la migración a las 8:00 am (conforme a la metodología ágil scrum)
- Una con el cliente para dar estatus del avance a las 10:00 am (básicamente un *stand up*)
- La sesión para dar seguimiento propiamente a las actividades a las 10:30 am como parte del trabajo de equipo.

Además, tenía comunicación continua con el equipo encargado de los respaldos. Por lo crítico que era cada cambio se decidió trabajar con imágenes de VMware (*snapshots*), de esta forma se podían reestablecer los ambientes fácilmente. Normalmente los ambientes eran respaldados de manera incremental diariamente, ahora se cambiaba a *snapshots*. Se tomó una imagen inicial de todos los servidores y se generaría una cada que se hiciera un “corte” de avance, es decir, un cambio significativo completo. Trabajar con *snapshots* era práctico ya que, en caso de necesitar restaurar, el procedimiento tomaba unos 10 minutos, de otra forma por medio de los respaldos tomaría hasta 4 horas.

Así comenzamos con un *Kick-off* el último lunes de mayo. Se empezó por desinstalar los servicios del clúster de BigInsights, configurar los repositorios y borrar directorios específicos. Durante los dos primeros días las cosas avanzaban normalmente. Sin embargo, al tercer día se procedió a hacer la actualización a HDP, debía realizarse en 7 de los 13 servidores de desarrollo, se trataba de estos servers:

- Los 2 nodos de administración
- Los 3 nodos de data
- BigIntegrate
- SPSS Analytic

En la figura 17 se puede apreciar a la arquitectura del ambiente de desarrollo a nivel de componentes. Como puede verse, SPSS Analytics y BigIntegrate debían considerarse como parte del clúster debido a las dependencias de las herramientas.

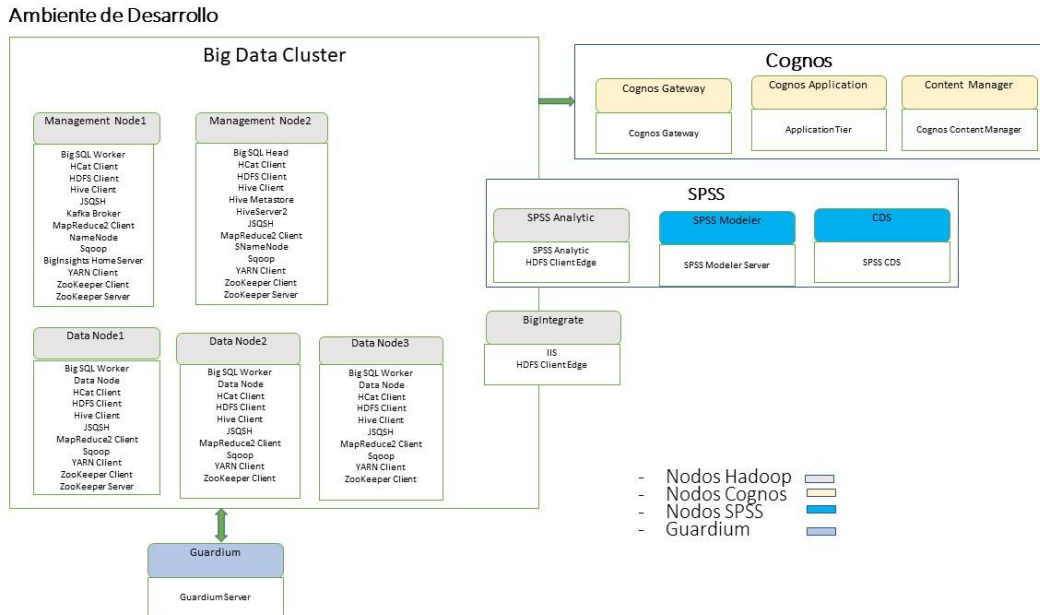


Figura 17 Arquitectura general del ambiente de desarrollo a nivel de componentes

La actualización en SPSS Analytic no funcionó correctamente. Al ser un clúster y levantar los servicios no se tuvo sincronización completa, era necesario revisar detalladamente. Abrimos un caso a soporte de Software Group (una división de la organización), un PMR. La respuesta fue que era necesario borrar otros directorios de SPSS de forma manual, de otra manera SPSS no reconocería la nueva versión de los servicios (HDP 2.6.4).

Informé al cliente y pedí restaurar a partir de la última *snapshot* (había pedido tomar una después del primer día). El problema se resolvió, pero nos demoró casi un día. Cada día adicional podía ser un problema pues los usuarios esperaban el clúster listo en una fecha determinada (a las 2 semanas).

A partir de ahí también se involucró a otro especialista de Lab Services, también un experto en BigData y en particular en migraciones de HDP. Tuvimos jornadas extendidas de más de 12 horas para poder terminar en tiempo. Debido al esfuerzo del equipo fue posible terminar la migración en la fecha esperada para el 31 de mayo de 2019. No obstante, aún faltaba más, Cognos y SPSS debían ser actualizadas o no serían compatibles con HDP.

Durante 2 días Cognos fue actualizado a la última versión, otros dos días después y con algunos contratiempos SPSS también estaba listo.

Tomamos un día más para hacer pruebas internas, le solicité a mi equipo correr algunos jobs, revisar los tableros de reportes y la funcionalidad en general. Después de 2 semanas y dos días entregamos el ambiente de desarrollo al cliente. Ellos revisaron y se aceptó a la migración y las actualizaciones como satisfactorios.

Una vez concluido todo para el ambiente de desarrollo se revisaron las fechas para la migración en producción, esta comenzó a mediados de julio. Al igual que para la primera etapa, agendé las sesiones de inicio internas, con el cliente y las relacionadas propiamente a las actualizaciones del clúster y las otras herramientas.

Para esos momentos contábamos con las experiencias de la migración de desarrollo. Se tomaron las medidas pertinentes para evitar los contratiempos y errores previos. De esta manera las actualizaciones en producción fueron mucho más estables y directas. La experiencia en las semanas previas fue muy útil.

En el caso de producción, la arquitectura era muy similar, aunque como puede apreciarse en la figura 18 se tenían dos nodos de administración adicionales. Se tomaron las medidas pertinentes de acuerdo a esas diferencias.

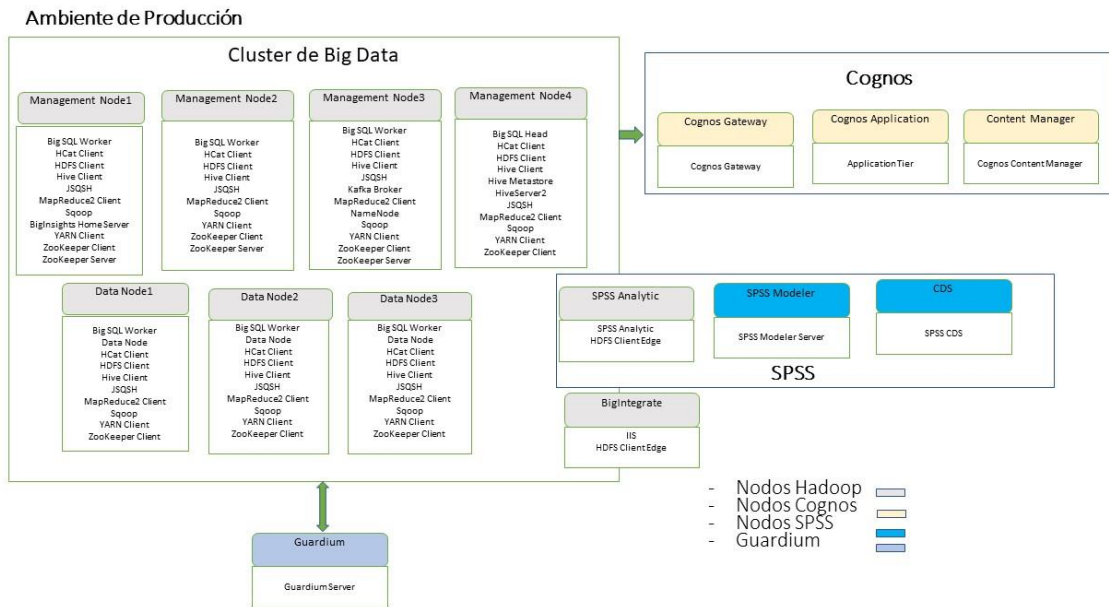


Figura 18 Arquitectura general del ambiente de producción a nivel de componentes

Después de 2 semanas la migración en el ambiente de producción estaba completa, también se habían actualizado Cognos y SPSS.

Realizamos las mismas pruebas que previamente habíamos ejecutado para el ambiente de desarrollo durante la etapa de despliegue, a su vez el cliente también ejecuto sus revisiones. Para la tercera semana de julio se aceptaron los ambientes y se pudo declarar como satisfactoria la migración para los dos ambientes de Big Data de City of Dallas.

Los temas de formalización fueron llevados por unos ejecutivos de la cuenta, uno de Dallas y otro de Florida. A ellos también les informaba de manera regular del avance, con ellos también tenía comunicación frecuente y por supuesto también les informé de la finalización satisfactoria de las actividades.



## 6.8 Actualización a HDP 3.x

En agosto de 2019 el principal contacto con los usuarios de City of Dallas me comunicó sus intenciones de actualizar una vez más los ambientes. El plan era tener la última versión disponible de HDP (3.1.x) en el clúster antes de que terminara el año.

El equipo de City of Dallas lo clasificó como un cambio prioritario y comenzaron a pedirme un plan de proyecto completo para hacer las actualizaciones correspondientes en los dos ambientes.

Por cuestiones de presupuesto en esta ocasión no habría posibilidad de incluir a especialistas de Lab Services. Mi equipo debía hacerse cargo de todas las actualizaciones en ambos ambientes.

Nuevamente nos fue de mucha utilidad el ambiente de pruebas que teníamos creado en servidores internos de la organización. El equipo a mi cargo estuvo haciendo actualizaciones de prueba durante unas 3 semanas. Al principio surgieron errores al intentar actualizar BigSQL y abrimos PMRs para solicitar ayuda. Se aclararon algunas dudas y finalmente fue posible hacer pruebas de actualizaciones que fueron satisfactorias.

Nuestro especialista de SPSS y Cognos también hizo pruebas satisfactorias. Fue necesario actualizar estas herramientas pues había dependencias de compatibilidad, aun cuando el cliente básicamente quería el cambio en HDP.

En el caso de BigIntegrate el cliente pidió no trabajar en la actualización pues ese componente no estaba en los planes del cliente para los próximos meses. Además, les hice notar que el tiempo era muy apretado por lo que prefirieron dedicar el esfuerzo a las otras herramientas.

Preparé un plan de trabajo donde incluía las actividades para actualizar los dos ambientes (incluyendo HDP, Cognos y SPSS), lo mandé al cliente, se revisó y se hicieron adecuaciones un par de veces. Teníamos también dependencia de la disponibilidad de los especialistas y el equipo del cliente, pues las fechas empalmaban con días festivos, básicamente el día de gracias en Estados Unidos.

Se determinó que las actividades en el clúster de desarrollo se llevarían a cabo durante la última semana de noviembre, teniendo 3 días para HDP. Sin embargo, se efectuarían otras actividades previamente:

- Aplicar parches a nivel de VMware a principios de noviembre
- Trabajar en la actualización de Cognos a mediados de noviembre

La actualización de SPSS se efectuaría en paralelo a la actualización de HDP, pues la parte de Analytic server se instalaba al final sobre la consola de Ambari (administración general de los servicios de Big Data) como un componente más del clúster.

De manera preventiva se pidió el apoyo de los especialistas de soporte de hardware para sacar imágenes de las máquinas virtuales antes de los cambios y pedía una imagen adicional cada vez que se lograba un avance significativo, es decir, trabajamos nuevamente con *snapshots* por la utilidad que tuvieron en la migración anterior. Por ejemplo, generábamos *snapshots* en estos casos:

- Después de la actualización de HDP
- Después de la actualización de BigSQL
- Al completar los cambios en el clúster, pero antes de instalar SPSS en Ambari.

De esta manera en caso de tener algún problema durante las actualizaciones nos sería posible pedir restaurar la imagen previamente guardada y comenzar de nuevo.

Llegaron las fechas y se efectuó el trabajo conforme a lo planeado. Tuvimos un par de inconvenientes que se resolvieron con ayuda de PMR's. Igualmente fue necesario pedir restaurar la imagen una vez durante los trabajos en HDP.

Se logró completar la actualización en el ambiente de desarrollo conforme a lo planeado.

A partir de aquí se confirmó que los trabajos en el ambiente de producción se efectuarían en la segunda semana de diciembre, teniendo igualmente 3 días para HDP. Los parches de VMware y la actualización de Cognos se trabajaron con anterioridad, tal como se realizó para desarrollo.

De esta forma el ambiente de producción quedó listo en la segunda semana de diciembre de 2019, el cliente hizo pruebas y encontró unos detalles menores que debieron corregirse por medio de un jar personalizado de sqoop, específicamente para City of Dallas. De eso se encargarían los especialistas que trabajaron en un PMR creado en especial para esa situación.

Así fue posible actualizar la plataforma de Big Data por segunda vez, en esta ocasión durante finales de 2019. El cliente debería trabajar en ese entonces sin problemas por lo menos durante más de un año, pues la versión disponible en esos días para el clúster (HDP 3.1) tenía soporte de Cloudera (compañía estadounidense) hasta diciembre del 2020.

## Capítulo 7 - Resultados.

Durante el tiempo que he laborado en esta organización he podido cumplir con las expectativas de los roles que he desempeñado. Por esa razón, me ha sido posible ir escalando posiciones de manera gradual y asumir actividades con más liderazgo y complejidad, tanto técnica como administrativamente.

El proyecto de City of Dallas me ayudó a desarrollar nuevas habilidades y retomar otras que había adquirido en trabajos con otras empresas, tales como las habilidades suaves o *soft skills* cuando se trata con cliente de manera personal. Aunque en este caso se tuvo el elemento adicional de colaborar en una lengua diferente y con unas culturas tan particulares como la tejana y la india.

En cuestión de las labores, el proyecto se ha dado de acuerdo a las expectativas de los recursos con los que contaba. Se desplegó en tiempo y forma el ambiente de producción completando al 100% las actividades del plan de trabajo, además se brindó el soporte al ambiente de desarrollo y también de producción apejándose al documento de entendimiento (contrato). Fue posible cumplir con los requerimientos del cliente cada vez y así entregar el servicio esperado, adicionalmente llevar una relación proactiva de colaboración con los especialistas.

Si bien había tomado entrenamiento de fundamentos de administración de proyectos, el trabajo directo como líder de proyecto en la parte de Delivery me ayudó a enfrentar de manera práctica una parte de esos conocimientos. Aún más durante el tiempo que desplegamos el ambiente de producción directamente en Dallas. El trabajar de manera continua y día a día liderando gente de Estados Unidos, México y Brasil fue una experiencia enriquecedora. Eso me permitió ver de manera directa su cultura laboral y también notar detalles de sus costumbres en general.

Como Delivery Manager en este proyecto me ha tocado cubrir prácticamente todos los procedimientos y generar todos los entregables estipulados en el procedimiento interno de Delivery. En muchas cuentas comerciales por la naturaleza y alcance de los proyectos solo se completan algunas partes. Sin embargo, City of Dallas ha sido un proyecto lleno de retos de diversos tipos. No solo he trabajado como Delivery Manager sino como Deployment

Manager y hasta Project Manager general del proyecto. Me han asignado actividades incluso de la parte financiera y cobranza.

La gente encargada de la parte financiera me delegó algunas actividades al inicio, pero después me tocó llevar toda la parte de la administración del proyecto incluyendo las finanzas. Para el despliegue de producción tuve que llevar el seguimiento de los costos del proyecto y de los gastos generados por el viaje para todos los miembros del equipo. Después tuve que dar seguimiento a esa parte para todo el proyecto. Además de todas las actividades de administración del proyecto y de liderazgo he tenido que asegurarme que la parte financiera no cayera en pérdidas para la corporación. A pesar de contar con varios años de experiencia laboral nunca me había tocado llevar los costos completos de un proyecto. En este caso tuve que enfrentar esa parte por primera vez y en otro país, con políticas y términos financieros algo distintos.

Para el despliegue fue necesario crear un formato propio desde cero y una manera propia de llevar ese seguimiento financiero. El formato que generé funcionó para City of Dallas y al final del despliegue del ambiente de producción sobró dinero del presupuesto, así no solo evitamos caer en números rojos sino conservar parte del presupuesto de contingencia.

Después, al llevar la administración de todo el proyecto, trabajé con unas herramientas internas de la organización para Project Management. Tanto para el seguimiento de finanzas, los planes, evaluaciones de proyecto e incluso para la cobranza.

Durante la mayor parte del proyecto he conseguido tener números positivos. Sin embargo, para la migración el proyecto cayó temporalmente en pérdida con respecto a las finanzas, esto porque se involucró a especialistas y ejecutivos que no estaban contemplados en el presupuesto original. El cambio de BigInsighst a HDP no se esperaba cuando inició y se firmó la cuenta. Afortunadamente poco después de la migración los números volvieron a ser positivos, aunque temporalmente con un margen de ganancia bastante pequeño.

En cuanto a los requerimientos y objetivos del cliente los resultados fueron concretos:

- El ambiente de producción fue entregado en la fecha esperada, en 3 semanas.
- El ambiente fue desplegado dentro del presupuesto esperado, sin ningún sobre costo y de hecho sobraron recursos del presupuesto original.
- El proyecto ha sido documentado con la información técnica y de administración del proyecto de acuerdo a los estándares de AppOps.
- Este fue el primer proyecto donde la documentación de transición pudo ser completada bajo los nuevos estándares de 2017 para el aseguramiento de calidad.
- La retroalimentación del líder local de Delivery Managers ha sido positiva, así también los comentarios de la líder global de Delivery Managers.
- Me ha permitido obtener calificación de “excede” en mi revisión anual con mi gerente.
- Mis evaluaciones posteriores también se clasificaron como “excede”, al menos un par de veces más, pues la administración completa del proyecto no estaba contemplada en mi rol original.
- La primera migración fue completada satisfactoriamente para los dos ambientes.
- El trabajo mostrado permitió empezar negociaciones para extender el contrato.
- Los trabajos para la segunda actualización de los ambientes también concluyeron satisfactoriamente.

El proyecto de City of Dallas ha sido muy enriquecedor en varios aspectos, tanto de manera laboral como personal. Al tener tanta visibilidad por parte de ejecutivos, líderes y gerentes también me ayudó a ser considerado para ser parte de nuevas iniciativas de mejora dentro de la organización, para roles adicionales y para planear nuevos retos de mejora durante 2018, 2019 y 2020.

## Conclusiones

Durante mi participación laboral en la organización he podido desarrollar más habilidades y competencias técnicas y de liderazgo, además me ha permitido trabajar de lleno en un ambiente multicultural y global.

Los roles que he podido desempeñar me han abierto la posibilidad de participar en otro tipo de proyectos y me han ayudado a crecer como especialista técnico y en roles de liderazgo.

La formación técnica recibida en la Universidad y la formación humanística de la misma me han ayudado a través de mi historia laboral a tener una mente abierta y dispuesta al cambio. Esto ha sido tanto técnicamente como personalmente. Todo me ha provisto de las herramientas técnicas y las habilidades blandas necesarias para enfrentar cambios, asumir nuevos retos, adquirir nuevos conocimientos y desarrollarme en la industria de tecnologías de la información. Desde los conceptos de organización de computadoras, el diseño digital y las demás materias propias de computación, hasta la estadística y ciencias básicas, todo me ha ayudado a adaptarme a diversos roles y asumir retos de manera constante.

El proyecto de City of Dallas significó asumir diversos nuevos retos al mismo tiempo. Implicó comenzar en un nuevo rol con un proyecto complejo, pasar de roles principalmente técnicos a uno de liderazgo y de administración de proyectos. También tener trato directo en otro idioma siendo “el rostro” de la organización para el cliente, pues, aunque ya tenía experiencia internacional con proyectos en Sudamérica, mi rol trataba brevemente con los clientes. Además, me significó realizar labores que jamás había hecho y con el agregado de ir “contra reloj”. Todo esto me ayudó a acelerar el aprendizaje con el fin de cumplir los objetivos de negocio, cosa que finalmente fue posible conseguir.

La experiencia acumulada hasta este proyecto me será útil en caso de que se dé la oportunidad de asumir mayores retos, con más complejidad y con grados más elevados de responsabilidad.

Como consecuencia de este proyecto se me asignaron más cuentas como Delivery Manager, en especial cuentas donde el trato con el cliente debía ser más personalizado y se necesitaban

más habilidades de comunicación. Mis líderes me han asignado otras cuentas (consideradas retadoras o complejas) basados en las aptitudes de comunicación que han notado en mí.

El rol de Delivery Manager me ha permitido conocer más a fondo la parte administrativa y financiera dentro de la industria. Además de ayudarme a desarrollar mucho más las habilidades de liderazgo, conciliación y negociación. Tanto con el cliente como con los asociados de negocios. De esta manera me será posible tener más conocimiento y habilidades que me permitan asumir nuevos retos para proyectos en el futuro, aun cuando vuelva a la parte técnica.



## Glosario

Ambari. – Es un proyecto de Apache orientado a facilitar la administración de Hadoop al crear software para provisionar, administrar y monitorear los clústeres.

AppOps. - Application Operations on Cloud es un área de servicio de GBS dedicada a brindar servicios de hosting, infraestructura, redes, middleware y soporte para clientes internos y comerciales de la organización.

Bare metal. – Se refiere a la parte física del disco duro. Sin embargo, términos más generales se relaciona a servidores físicos, lo opuesto a los servidores virtuales. Por ser servidores tradicionales suelen dar servicio a un solo cliente o consumidor.

Benchmarks. – En términos computacionales se relaciona a ejecutar programas, grupos de programas, operaciones o aplicaciones con el objetivo de evaluar su desempeño con base a ciertos parámetros como el consumo de memoria o procesador.

Build. - Se refiere al proceso por el cual el código fuente de un programa es empaquetado y convertido en una forma tal que pueda ser ejecutado. Uno de los principales pasos en la generación de un build es el proceso de compilación donde el código fuente es transformado en código ejecutable.

CLM.- Solución Rational para Collaborative Lifecycle Management (Administración de ciclo de vida colaborativo), es un conjunto de aplicaciones integradas para llevar a cabo la administración de ciclo de vida de aplicaciones. Estas herramientas se pueden instalar y configurar para trabajar como una sola.

Cloud. – Literalmente significa nube en español, se refiere a entrega de servicios de computación bajo demanda desde aplicaciones hospedadas en centros de datos. Los centros de datos trabajan brindando los servicios a través de redes mundiales de servidores interconectados.

DevOps. - Development and Operation es una cultura de desarrollo de software enfatizada en la interacción e integración de áreas de TI a través de despliegues continuos de software y automatización.

Eclipse. – Es un ambiente de desarrollo integrado que se usa en programación para el lenguaje java. Sin embargo, puede trabajar con otros lenguajes y con otras funcionalidades a través de distintos conectores personalizados.

ETL. – Son las siglas de extraer, transformar y cargar. Es un procedimiento de integración de datos que utiliza las acciones citadas para consolidar información de diferentes fuentes de datos en una ubicación centralizada.

Firewall. – En español cortafuegos, es un dispositivo de red para seguridad que monitorea y filtra la comunicación de salida y la de entrada basado en las políticas de seguridad definidas.

GBS. - Global Business Services es una de las principales ramas de negocios de la organización, principalmente relacionada a brindar servicios profesionales de Tecnologías de la Información a empresas en todo el mundo.

Hadoop. - Proyecto de software abierto de Apache que permite el proceso distribuido de grandes conjuntos de datos en clúster de servidores. Utilizado como plataforma de BigData para muchas soluciones comerciales de la industria.

HDFS. – Es un sistema de archivos distribuidos que ayuda a manejar grandes grupos de datos, es usado para escalar clústeres de Hadoop.

Herramientas Rational. - Suite de herramientas de desarrollo de software de manera iterativa y adaptadas a las mejores prácticas de la industria.

Hypervisor. – Es software o hardware que a través de emulación crea y ejecuta máquinas virtuales.

ITIL (Information Technology Infrastructure Library). - Es un grupo de prácticas detalladas desarrolladas en Inglaterra para administración de servicios de tecnologías de la información.

PMR. - (Problem Management Report) es una solicitud de soporte especializado de la organización, se brinda asistencia sobre problemas de software de las herramientas o de usabilidad.

Rational Unified Process (RUP). - Proceso de desarrollo iterativo creado por Rational Software.

Middleware. – Es software que reside entre el sistema operativo y las aplicaciones tal como lo son los servidores web y los servidores de aplicaciones.

SaFE. – Las siglas significan *Scaled Agile Framework Enterprise*, es decir, marco de trabajo ágil escalado. Es una metodología ágil para organización empresarial.

Scrum. – Es un modelo de trabajo que se utiliza para el desarrollo de software de manera ágil. Hace uso de buenas prácticas y la colaboración constante entre los miembros del equipo de trabajo.

Scrum master. – Es un facilitador para equipos que trabajan utilizando la metodología de scrum. Se asegura de que el equipo se adhiera a las prácticas y que trabaje de manera autónoma y auto organizada.

Spark. - Framework basado en clústeres, originalmente creado en la Universidad de California. Desarrollado para mejorar la eficiencia de algoritmos de tipo MapReduce muy utilizados en BigData.

Sprint. – Es un mini proyecto, típicamente toma entre 1 y 4 semanas, durante este tiempo se genera software utilizable y potencialmente entregable. Esta práctica se utiliza en las metodologías ágiles como Scrum.

Sqoop. - Es una herramienta diseñada para transferir de manea eficiente grandes cantidades de datos entre Apache Hadoop y fuentes de datos estructuradas como bases de datos relacionales.

Streaming. - Retransmisión de contenido multimedia, en gran medida de videos, pero también de música. El contenido se va reproduciendo conforme se va descargando

TI.- Tecnologías de la Información

Webex. – Es una empresa estadounidense que desarrolla y vende servicios para conferencias y videoconferencias. Ahora es parte de Cisco systems.

# Referencias y Bibliografía

## Ingeniería de Software

- [1] Pressman R. S. (2010). *Ingeniería del Software, Un enfoque Práctico*. New York, NY, Estados Unidos de América: Mc Graw Hill.
- [2] Sommerville I. (2005). *Ingeniería del Software*. Madrid, España: Pearson.
- [3] Tsui F., Karam O., Bernal B. (2018). *Essentials of Software Engineering*. Burlington, MA, Estados Unidos de América: Jones & Bartlett Learning.

## Big Data

- [4] Hoglund, A. (2018). Big Data. Recuperado el 18 de marzo de 2020, de <https://www.usfca.edu/magazine/summer-2016/big-data>
- [5] “Research.” Oxford Big Data Institute. (n.d.). Big Data Institute. Recuperado el 18 de marzo de 2020, de <https://www.bdi.ox.ac.uk/research>
- [6] Marr, B. (2015). Big Data: 20 Mind-Boggling Facts Everyone Must Read. Recuperado el 14 de agosto de 2019, de <https://www.forbes.com/sites/bernardmarr/2015/09/30/big-data-20-mind-boggling-facts-everyone-must-read/#4eef396117b1>
- [7] Khattak, W., Erl, T., & Buhler, P. (2020). *Big data fundamentals concepts, drivers & techniques*. Boston, MA, Estados Unidos de América: Pearson.

[8] EMC Education Services (2015). *Data Science and Big Data Analytics*. Indianapolis, IN, Estados Unidos de América: Wiley.

### **Project Management**

[9] PMI (2017). *A Guide to the Project Management Body of Knowledge*. Newton Square, PA, Estados Unidos de América: Project Management Institute, Inc.

[10] “What is Project Management?” (n.d.). Recuperado el 9 de octubre de 2020, de <https://www.pmi.org/about/learn-about-pmi/what-is-project-management>

### **BigInsights y Herramientas relacionadas**

[11] IBM (n.d.). IBM Open Platform with Apache Hadoop 4.0 documentation. Recuperado el 18 de marzo de 2020, de [https://www.ibm.com/support/knowledgecenter/en/SSPT3X\\_4.2.0/com.ibm.swg.im.infospere.biginsights.welcome.doc/doc/welcome.html](https://www.ibm.com/support/knowledgecenter/en/SSPT3X_4.2.0/com.ibm.swg.im.infospere.biginsights.welcome.doc/doc/welcome.html)

[12] IBM. (n.d.). IBM Db2 Big SQL. Recuperado el 18 de marzo de 2020, de <https://www.ibm.com/analytics/us/en/technology/biginsights/>

[13] IBM (n.d.). IBM BigIntegrate. Recuperado el 18 de marzo de 2020, de <https://www.ibm.com/us-en/marketplace/ibm-biginsights-bigintegrate>

## **SPSS**

[14] IBM (n.d.). IBM SPSS Modeler V18.1.0 documentation. Recuperado el 18 de marzo de 2020 de [https://www.ibm.com/support/knowledgecenter/en/SS3RA7\\_18.1.0/modeler.kc.doc/clementine/knowledge\\_center/product\\_landing.html](https://www.ibm.com/support/knowledgecenter/en/SS3RA7_18.1.0/modeler.kc.doc/clementine/knowledge_center/product_landing.html)

[15] IBM (n.d.). IBM SPSS Statistics. Recuperado el 18 de marzo de 2020, de <https://www.ibm.com/products/spss-statistics>

## **Cognos**

[16] IBM. (n.d.). IBM Cognos Analytics 11.0 documentation. Recuperado el 18 de marzo de 2020, de [https://www.ibm.com/support/knowledgecenter/en/SSEP7J\\_11.0.0/com.ibm.swg.ba.cognos.cbi.doc/welcome.html](https://www.ibm.com/support/knowledgecenter/en/SSEP7J_11.0.0/com.ibm.swg.ba.cognos.cbi.doc/welcome.html)

[17] IBM (n.d.). IBM Cognos Analytics. Recuperado el 18 de marzo de 2020, de <https://www.ibm.com/products/cognos-analytics>

## **City of Dallas**

[18] Dallas City Hall (n.d.). Welcome to Dallas. Recuperado el 18 de marzo de 2020, de <http://dallascityhall.com/Pages/default.aspx>

## **Hadoop**

[19] Apache (n.d.). Apache Hadoop. Recuperado el 18 de marzo de 2020, de <http://hadoop.apache.org/>

## **Hortonworks**

[20] Cloudera, Inc, & Apache Software Foundation. (2020, March 5). El data Data Cloud empresarial. Recuperado el 18 de marzo de 2020, de <https://es.cloudera.com/>

# Anexos

## Implicaciones del Big Data

Desde hace mucho tiempo el valor de la información ha sido sumamente importante para el desarrollo de las organizaciones e incluso de las naciones. Desde principios de la década de los 80's se tuvo un acelerado crecimiento de la información en formato digital y por ende el crecimiento de la explotación en ese formato fue exponencial.

Las bases de datos relacionales han tenido gran auge y varios sistemas manejadores de bases de datos han tenido una gran importancia y notoriedad desde los 80's y también durante los 90's. La mayoría de las empresas y organizaciones utilizaron ese paradigma desde los 90's y durante la primera década de los 2000 para almacenar la información de sus sistemas.

Sin embargo, las cosas tuvieron un cambio sensible conforme las tecnologías de la información se hicieron más accesibles. Esto por el gran aumento de cobertura de internet y de las redes de comunicaciones. En particular el gran crecimiento de las redes de datos colaboró a generar mucha información, todo esto ayudó a acelerar la generación de datos de todo tipo en gran medida.

En la actualidad la mayor parte del contenido creado en todo el mundo ya nace digitalizado. Desde información estructurada hasta la no estructurada. Cada día se genera una cantidad impresionante de información. Para mediados de 2019 se tenían unos 4, 390 millones de usuarios en internet, todas estas personas en conjunto son capaces de generar una cantidad enorme de datos que nunca antes se había visto en la historia de la humanidad.

Las redes sociales, los sitios de videos y *streaming* han colaborado a aumentar aceleradamente la generación y el flujo de datos a través de internet. Si se considera también el crecimiento planeado del internet de las cosas podemos esperar aún más información durante los años venideros.

Hoy en día el Big Data incluye todo tipo de información, estructurada y no estructurada. Se incluyen archivos de texto, información de sensores, audio, video, servicios de *streaming*, archivos de logs y más.



Cada día son generados cerca de 2.5 millones de terabytes en todo el mundo, el crecimiento tan acelerado y exponencial se refleja en que el 90% de la información en internet fue generada en los últimos 2 años como se puede apreciar en la figura 19.

En estos días la información que inunda al mundo es en su mayor parte no estructurada. La capacidad de categorizar, analizar y explotar esa información tiene gran potencial de producir grandes beneficios para las empresas. Pero también para gobiernos, científicos y organizaciones no lucrativas. El rol de los científicos de datos se vislumbra como una necesidad creciente en gobiernos y en todo tipo de organizaciones de notable tamaño.

El Big Data apenas está comenzando a entenderse u explotarse, pero en unos pocos años parece que será una necesidad fundamental para todas las organizaciones medianas y grandes de este mundo globalizado y tan digitalizado. De hecho, la organización esperaba que la demanda por científicos de datos creciera a un 28% para el año 2020.

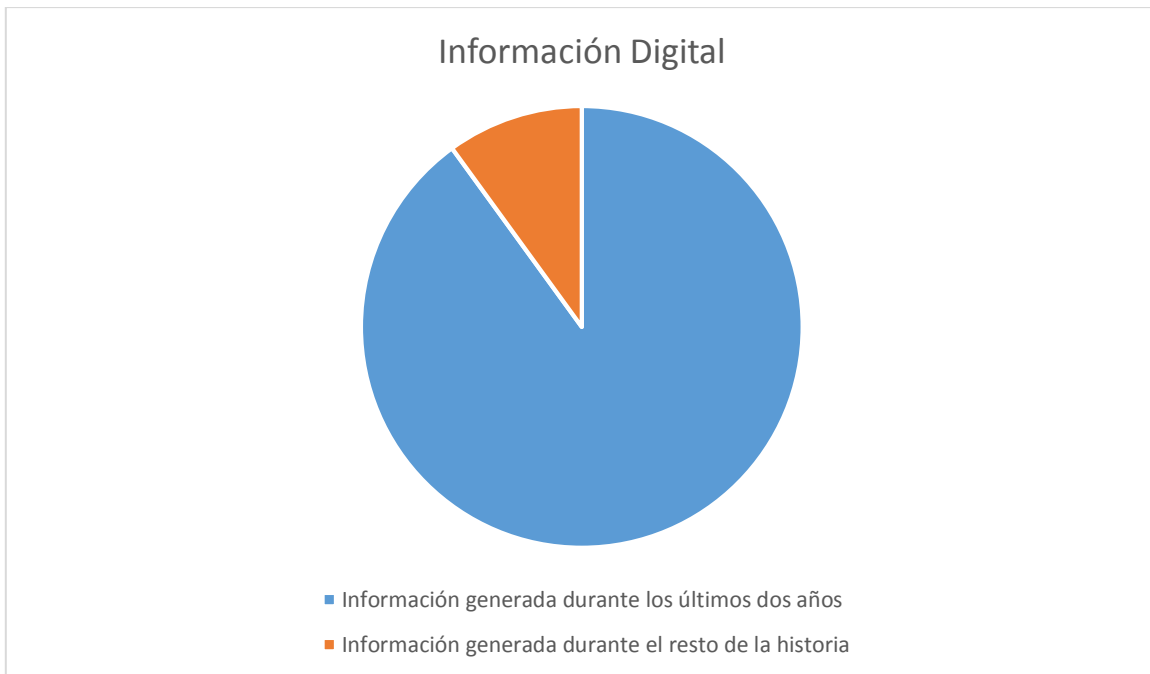


Figura 19 Información generada en el mundo

Las inversiones de las empresas y gobiernos en Big Data son millonarias. De acuerdo a la revista Forbes tan solo para Hadoop se esperaba un crecimiento de un 58% para el 2020, sobrepasando un billón de dólares para estas fechas.

Si bien algunas personas aún piensan en el Big Data como algo nuevo, la realidad es que ya está aquí, llegó para quedarse y avanza a una gran velocidad. Es cada vez más, la tecnología que conduce las estrategias de marketing para muchas empresas.