



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

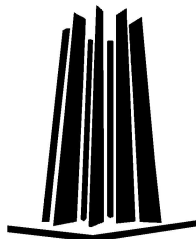
---

**FACULTAD DE ESTUDIOS SUPERIORES  
ARAGÓN**

**“EXPERIENCIA PROFESIONAL EN LA  
ADMINISTRACIÓN DE BASE DE DATOS ORACLE  
(2001-2005)”**

**T R A B A J O E S C R I T O  
EN LA MODALIDAD DE INFORME DEL  
EJERCICIO PROFESIONAL  
QUE PARA OBTENER EL TÍTULO DE:  
INGENIERO EN COMPUTACIÓN  
P R E S E N T A :  
M A R T I N L Ó P E Z  
G U A D A R R A M A**

**ASESOR: M. EN C. MARCELO PÉREZ MEDEL**



**MÉXICO, 2006.**



Universidad Nacional  
Autónoma de México



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

<b>Índice</b>	<b>Página</b>
Introducción .....	iii
Capítulo 1 Softtek® .....	1
1.1. Historia y objetivo .....	1
1.2. Proceso de selección e inducción .....	1
1.3. Asignación a proyectos .....	3
1.3.1. Procedimiento .....	3
1.3.2. Diseño de reportes .....	4
Capítulo 2 Soporte a aplicaciones .....	10
2.1. Antecedentes .....	10
2.2. Soporte a Profiler .....	15
2.2.1. Funciones y actividades .....	16
2.3. Soporte al Dataware House .....	22
2.3.1. Funciones y actividades .....	22
Capítulo 3 Soporte a la plataforma de Broadvision .....	26
3.1. Antecedentes .....	26
3.2. Funciones y actividades .....	30
3.2.1. Tareas de Gestión y Mantenimiento .....	31
3.2.2. Tareas de Soporte .....	37
Reflexiones .....	49
Bibliografía .....	54

## Introducción

Cuando se inicia la vida de profesionista hay que enfrentar una serie de obstáculos, empezando por la fuerte competencia que existe y la falta de experiencia. Con este informe se describen las experiencias que afronta un recién egresado, esperando que sirvan de apoyo para aquellos profesionistas que están por concluir o han concluido sus estudios.

Se conocerá un poco de lo que es Softtek® como organización, las diferentes formas de empezar a laborar y la manera de trabajar de las consultorías. Se podrá ver que se puede ingresar a una empresa sin tener la experiencia que piden en la mayoría de los empleos y que ser becario es una buena opción, además de que los cursos que imparten ayudan mucho al crecimiento profesional y con lo cual en el futuro ayudan a abrir otras puertas.

También se verá la experiencia adquirida al trabajar con diferentes personas en distintos soportes y los problemas que se pueden llegar a presentar, así como la forma de solucionarlos y conseguir el reconocimiento de los usuarios. Con lo cual se puede obtener un mejor puesto con más responsabilidades.

Finalmente, al tener la experiencia necesaria, se puede conseguir un mejor puesto y a la vez enfrentar nuevos retos, y lograr un mayor crecimiento profesional dentro del mundo laboral.

Lo importante es que quien lea este informe obtenga un panorama de lo que es la carrera de sistemas y de las áreas que requieren de esta profesión para después integrarse al mundo laboral con una idea más clara y de esta forma ser exitosos.

# Capítulo 1

## Softtek®

### 1.1 Historia y objetivo

Softtek® nace en 1982 como proveedor de servicios de *software*<sup>1</sup> para satisfacer las necesidades de desarrollo y soporte de las medianas y grandes empresas; integrando soluciones inteligentes con tecnología de vanguardia.

Su finalidad es la atención a clientes de acuerdo a los sectores de la industria y a las diversas tecnologías de información. Se elaboran soluciones que abarcan el desarrollo de sistemas a la medida o la implantación de *software* comercial, hasta la integración crítica del uso de tecnologías de Internet para mejorar y transformar los procesos clave de las empresas.<sup>2</sup>

Actualmente Softtek® tiene presencia en varios países, ofreciendo servicios a empresas nacionales y trasnacionales. También cuenta con Centros de Desarrollo, con los cuales cubre las necesidades de las empresas, sin tener que estar físicamente en las oficinas de éstos. De esta manera, se da un gran avance en los servicios de soporte de aplicaciones y desarrollo de sistemas, logrando cumplir con las expectativas de los clientes y dando servicios de primera calidad.

### 1.2 Proceso de selección e inducción

Existen varias maneras de ingresar a Softtek®, por algún conocido (que trabaje en la empresa), un anuncio en el periódico o por Internet (en páginas dedicadas a anuncios de empleo). El primer paso es enviar el Currículo y esperar a ser llamado para una entrevista con Recursos Humanos; durante la reunión las

---

<sup>1</sup> Conjunto de programas que puede ejecutar el hardware para la realización de las tareas de computación.

<sup>2</sup> [www.softtek.com](http://www.softtek.com)

preguntas van dirigidas a los estudios que realizó el aspirante, sus expectativas y acerca de su familia, después realizan pruebas de tipo: psicométrico<sup>3</sup> (los cuales sirven para tener un marco de referencia de sus habilidades para razonamiento y atención), conocimientos y habilidades (éstos son especializados en el área para el puesto requerido y miden las habilidades técnicas y las fortalezas); después de éstas evaluaciones informan al prospecto si cumplió con el perfil solicitado, el cual puede ingresar para formar parte de un proyecto o como becario.

Si el aspirante es aceptado, toma una plática de inducción en la cual dan una reseña de lo que es la empresa, los objetivos, las obligaciones y prestaciones a las que es acreedor. De acuerdo a su perfil, se asigna a un proyecto, se le informa sobre el cliente, las actividades que va a realizar, el lugar donde se debe presentar (puede ser en las instalaciones de la misma empresa o en las del cliente) y a quien debe dirigirse. Posteriormente el líder del proyecto le dará más detalle.

En caso de que inicie como becario, recibe una plática de bienvenida, donde se presentan a los instructores, el objetivo de cada curso, la manera de evaluar en cada uno de ellos y las consecuencias que tiene salir mal en las evaluaciones, esto es, van a depurar dentro del grupo a las personas que no están cumpliendo con las expectativas que se tiene de ellos.

Los cursos que se imparten tienen como finalidad formar recursos que puedan desempeñar actividades como desarrolladores y cubran las necesidades del proyecto en el que van a trabajar, los lenguajes de programación más comunes son: *SQL*, *PL/SQL*, *Unix*, *Shell Script* y herramientas como son: *sqlplus* de Oracle<sup>4</sup>.

En el curso de *sqlplus*, se explican los comandos necesarios para el formateo de la salida de las consultas que se realizan, generación de reportes usando instrucciones propias de la herramienta.

---

<sup>3</sup> es.[wikipedia.org/wiki/Psicom%C3%A9trico](http://es.wikipedia.org/wiki/Psicom%C3%A9trico)

<sup>4</sup> Herramienta que permite introducir instrucciones *SQL* y bloques de *PL/SQL* mediante línea de comandos.

En *SQL*<sup>5</sup> se ven los principales comandos, funciones, cláusulas y operadores usados para realizar consultas a la base de datos, la diferencia con los comandos de *sqlplus*.

En el de *PL/SQL*<sup>6</sup>, se ven las estructuras de control, secuencias, cursores<sup>7</sup>, tipos de datos, manipulación de la información, *triggers*<sup>8</sup>, funciones, paquetes, procedimientos almacenados y bloques anónimos<sup>9</sup>.

En el curso de *Unix*, se estudia la arquitectura del sistema operativo, comandos básicos, herramientas para edición de texto, programación *Shell*, estructuras de control y monitoreo de procesos.

Al final de cada curso se aplican exámenes prácticos y teóricos de lo que se vio en cada modulo, sólo los que aprobaron son aceptados, aunque también depende de la demanda de recursos que haya en ese momento. Terminada la capacitación son asignados a proyectos.

### 1.3 Asignación a Proyectos

Dentro de la empresa, existen varias cuentas, en cada una hay un gerente de operación, el cual solicita al área de recursos humanos gente que cubra alguna plaza o para algún proyecto, éstos últimos se encargan de administrar el personal, canalizan a las personas en las cuentas y de ésta manera se van asignando los recursos a proyectos.

#### 1.3.1 Procedimiento

Cuando es solicitado un consultor, buscan a la persona que tenga el mejor perfil, para cubrir las necesidades del proyecto al que será asignado,

---

<sup>5</sup> Lenguaje estructurado de consultas, sirve para manipular registros de bases de datos relacionales.

<sup>6</sup> Lenguaje de programación utilizado para acceder a bases de datos Oracle desde distintos entornos.

<sup>7</sup> Se emplea para procesar múltiples filas extraídas de la base de datos, con una orden select.

<sup>8</sup> Es un bloque de *PL/SQL* que ejecuta automáticamente instrucciones al insertar, actualizar o borrar.

<sup>9</sup> Conjunto de instrucciones en *PL/SQL* que se construyen de forma dinámica y se suelen ejecutar una sola vez.

consideran la experiencia, conocimientos de las herramientas a utilizar, dominio de algún idioma (generalmente el inglés), hecho esto, le informan al mejor candidato que será enviado a alguna de las cuentas a un proyecto y le indican con quien debe dirigirse.

Cuando el recurso está asignado, tomará el horario de la cuenta donde va a trabajar, hay que mencionar que laborar en sistemas es tener un horario flexible, ya que no se tiene una hora de salida pero si de entrada, a esto hay que sumar que cada empresa tiene sus políticas y se debe acoplar a éstas cuando no esté en las oficinas de la organización.

### **1.3.2 Diseño de reportes**

Dentro de los proyectos que tiene la empresa, necesitan una persona que tenga conocimientos en manejadores de base de datos, programación y creación de formas y reportes. De los recursos disponibles buscan a quien cubra con el perfil requerido, y es enviado con el líder de proyecto que hizo la solicitud. Cuando se presenta con él, se le explica en que consiste el proyecto y cuales son las tareas a realizar, así mismo, es presentado con los miembros de la organización que también están asignados a la cuenta.

Dentro de la cuenta, en una de las áreas a las que dan soporte, se tiene un equipo de recursos, los cuales atienden las solicitudes de los usuarios involucrados, éstos requieren modificaciones a los componentes activos en el ambiente productivo. Como reciben muchas solicitudes, el líder decidió que se realizara una aplicación para llevar un mejor control de cada uno de los requerimientos recibidos y estadísticas del servicio proporcionado, para lo cual se creó una base de datos para almacenar la información de las peticiones del cliente; ya se tenía un avance de esta aplicación y se realizaron los siguientes puntos:

- a. Crear formas para capturar la información de cada requerimiento, en las cuales se considera lo siguiente: el usuario que hace la solicitud, fecha y hora en la que se solicita, descripción del problema, nombre del recurso que llevará a cabo la petición, tiempo estimado de resolución, prioridad, fecha y



- hora de termino del requerimiento, tiempo que tomó llevar a cabo la tarea, estatus.
- b. Generar reportes donde se muestren los requerimientos que tienen asignados los recursos.
  - c. Generar reportes donde se muestre con un semáforo el estatus que tienen los requerimientos, utilizando los siguientes colores: verde para los requerimientos que aún no sobrepasan la fecha de estimación; amarillo para los requerimientos que están sobre la fecha estimada y rojo para aquellos que ya rebasaron la fecha de estimación.
  - d. Generar un reporte donde se muestre una descripción de cada uno de los puntos que califica el cliente del servicio que están recibiendo, para lo que se toma en cuenta: el número de requerimientos resueltos a tiempo, cuantos fueron resueltos después de la fecha de estimación, cuales aún no se resuelven y cuya fecha de estimación aún no se sobrepasa, los que no se han resuelto y sobrepasaron la fecha de estimación. Para estos casos se representa con un color y se despliega un porcentaje para cada uno de los puntos a calificar, los colores son: verde para los que han sido exitosos; amarillo para los que han salido en el tiempo estimado y el esperado por el cliente y rojo para aquellos que han tenido retraso y que sobrepasaron el tiempo esperado por el cliente.

Se debe terminar de construir la base de datos, para que cumpla con los requerimientos del proyecto, pueda almacenar la información requerida, y así, concretar los puntos descritos con anterioridad.

El primer paso a seguir es analizar cada uno de los puntos para el desarrollo de la aplicación, ya que han sido entendidos, iniciar por el más importante, en este caso, revisar que tablas existen, cuales son sus atributos, las relaciones. Con base al análisis previo, determinar si hacen falta componentes, si es así, crear los objetos necesarios, con la finalidad de tener una base de datos consistente y que ayude a cumplir con los puntos del requerimiento.

El siguiente paso es crear las formas para llenar las tablas, se debe buscar un diseño amigable y fácil de utilizar, para que al usuario final no se le dificulte trabajar con ellas. Se deben presentar al menos 3 formatos diferentes al

interesado para que decida cual le parece más adecuado, hecho esto se hacen las formas para las tablas que las necesitan. Algo importante que no se debe olvidar, es que cumpla con todos los puntos que se especifican en el requerimiento. También se puede dar el caso que tengan formatos establecidos y deben considerarse para realizar las formas.

Una vez terminadas las formas y habiendo realizado las pruebas correspondientes, se procede a llenar las tablas con información para contar con datos para los reportes del proyecto.

El primer reporte es sobre los casos asignados a cada recurso, para esto, se deben buscar los requerimientos que está atendiendo cada recurso; ya que se tiene esta información, entonces se crea, éste puede ser gráfico o sólo con los nombres de las personas que están trabajando y el total de solicitudes que tiene cada uno, si es gráfico puede mostrarse un esquema de barras que indique el nombre y el total de casos que atiende, con esto podremos verificar la cantidad de requerimientos que está atendiendo tanto el equipo de soporte como cada recurso.

El siguiente reporte debe mostrar el número de requerimientos, la descripción del caso, la fecha en que se abrió la solicitud, el momento en que se empezó a atender, el día estimado para entregarlo, el estado (éste no puede ser cerrado para ser tomado en cuenta), así como el nombre del recurso que lo atiende; con base en las fechas se va a determinar la situación del requerimiento, dos datos importantes que debemos tomar en cuenta son: la estimación para la solicitud, la cual llamaremos: fecha de estimación, y cuando se genera el requerimiento, aunque esta última no se muestra en la información presentada, la debemos considerar, ya que nos dará la pauta para obtener el estado en que se encuentra la solicitud, esto es, el semáforo que están pidiendo, ésta funcionalidad va a indicar si la atención es rápida o es tardía, por lo tanto, se va a comparar la fecha actual contra la del día de la solicitud, con lo cual determinaremos lo siguiente:

- a. Si la estimación es mayor a la fecha actual, entonces el requerimiento tendrá color verde, lo que quiere significa que está en tiempo y que no hay retraso.

- b. Si la estimación es igual a la fecha actual, entonces se pintará un color amarillo lo cual indica que está en tiempo y que debe solucionarse de inmediato el requerimiento.
- c. Si la estimación es menor a la fecha actual, entonces el semáforo estará en color rojo, lo cual quiere decir que se tiene un retraso e implica que no se está dando la atención necesaria, además puede repercutir en la percepción del cliente hacia el servicio que se está dando.

De esta manera se tiene un reporte que mostrará cada uno de los casos que aún no están cerrados, y por medio de un semáforo se indicará el estado de atención en el que se encuentran, esto ayudará a tener una idea más amplia del trabajo que se está desarrollando y si se atiende correctamente.

Este último reporte será mensual y mostrará los puntos que califica el cliente del servicio que se está ofreciendo y deben considerar los siguientes puntos:

- ❖ Tipo de requerimiento.
- ❖ Descripción
- ❖ Porcentaje del nivel de atención (éste punto deberá funcionar como semáforo)

Para el número de requerimientos resueltos a tiempo, se va a tomar el total de solicitudes hechas durante el mes y el total de cerrados antes de la fecha de estimación, obtenidos estos datos se deberá aplicar la siguiente fórmula:  $(\text{Total de cerrados a tiempo}) \times (\text{cien}) / (\text{total de solicitudes hechas})$ ; si el resultado es mayor a 85, entonces se encenderá en color verde el porcentaje; si es mayor a 75 y menor a 85, entonces se prenderá en color amarillo y si es menor a 75 entonces mostrará un color rojo.

El porcentaje de los requerimientos que fueron cerrados después de la fecha de estimación se calcularán de la siguiente manera: se obtendrá el total de solicitudes abiertas en el mes y el total de cerrados después del día estimado, con estos datos se deberá aplicar la siguiente fórmula:  $(\text{Total de cerrados después de la fecha de estimación}) \times (\text{cien}) / (\text{Total de solicitudes abiertas})$ ; si el resultado es mayor a cero y menor a 15, entonces se encenderá en color verde;

si es mayor a 15 y menor a 40, entonces se prenderá en color amarillo y si es mayor a 40 entonces se mostrará en color rojo.

Para los requerimientos que aún no se ha cerrado y la estimación no ha expirado, se tomarán los siguientes valores: Total de solicitudes abiertas en el mes, el total de requerimientos sin cerrar y cuya fecha de estimación aún no se pasa, teniendo esta información se aplicará la siguiente fórmula:  $(\text{Total de requerimientos sin cerrar}) \times (\text{cien}) / (\text{Total de solicitudes abiertas})$ ; si el resultado es mayor a 80, entonces se encenderá en color verde; si el resultado es mayor a 60 y menor a 80, entonces se prenderá en color amarillo y si el resultado es menor a 60 entonces se mostrará en color rojo.

Finalmente, para los requerimientos que aún no se cierran y que sobrepasaron la fecha de estimación, se deben obtener los siguientes datos, total de solicitudes abiertas en el mes y el total de requerimientos sin cerrar cuya fecha de estimación ya se sobrepasó, con estos datos se deberá aplicar la siguiente fórmula:  $(\text{Total de requerimientos sin cerrar}) \times (\text{cien}) / (\text{Total de solicitudes abiertas})$ ; si el resultado es mayor a 0 y menor a 10, entonces se encenderá el color verde; si es mayor a 10 y menor a 20, entonces se prenderá en color amarillo y si es mayor a 20 entonces se mostrará en color rojo.

Con esto se finaliza el requerimiento solicitado, pero antes de ser utilizado se debe probar para mostrarlo al usuario que hizo la solicitud, para esto se prepara una serie de revisiones, que muestren el funcionamiento de la base de datos, los formularios y los reportes hechos, para lo cual se debe realizar lo siguiente:

- a. Agregar información desde los reportes realizados y verificar que estén en las tablas correspondientes.
- b. Ejecutar el reporte de los casos asignados por recurso, para revisar que la información que este mostrando sea correcta.
- c. Ejecutar el reporte para el estatus de los requerimientos, y validar que la información esté bien.
- d. Finalmente, ejecutar el reporte con la información que se va a mostrar al usuario y con la cual va a evaluar el servicio que se está ofreciendo, en este

caso se debe obtener un ejemplo de cada uno de los casos para validar que todo este funcionando correctamente.

Si estas pruebas se realizan correctamente entonces se avisa al usuario y se muestra la aplicación que se realizó, de lo contrario se corrigen los posibles problemas que se puedan presentar, con la finalidad de mostrar una aplicación confiable y estable.

El usuario es quien determina si la aplicación requiere de más cambios y si es lo que esperaba, las modificaciones se deben hacer de acuerdo a la lógica de programación o al diseño que presenta la aplicación. Finalmente, durante la vida del sistema se van a estar realizando cambios, los cuales van a ir en función a las necesidades del usuario y de la información que se necesite.

Con todo esto, se entrega una aplicación donde se puede capturar información de requerimientos de un equipo de soporte, se muestran reportes que ayudarán a detectar los puntos en los que se está fallando y conseguir las mejoras necesarias para que el servicio sea el adecuado y el que está esperando el cliente.

## Capítulo 2

### Soporte a aplicaciones en Oracle

#### 2.1 Antecedentes

En una de las cuentas que se dedica a equipos de cómputo existe la necesidad de una persona que cubra las ausencias de los titulares de ciertas aplicaciones, éstas pueden ser por vacaciones, enfermedad e inclusive para apoyar en caso de carga de trabajo o contingencia dentro de la aplicación. Para lo cual debe tener conocimientos técnicos en: *SQL*, *PL/SQL*, sistema operativo *Unix* y programación *Shell*<sup>10</sup>.

Al cual se le conocerá como *backup* y la idea principal de tenerlo es para no descuidar el servicio que están recibiendo en los casos antes mencionados, deberá ser capacitado por los titulares de las aplicaciones, enseñándole todas las tareas que se llevan a cabo para dar el soporte, es importante resaltar que al inicio no cubrirá al 100% las actividades del titular, pero al menos podrá realizar las actividades más importantes dentro de cada aplicación, con el tiempo irá adquiriendo experiencia y esto le ayudará a dar un mejor soporte.

Las aplicaciones a las que se les da soporte son: *Profiler* y el *Dataware House*, el *backup* se entrenará durante las primeras semanas para conocer las tareas principales, por lo que deberá organizar su tiempo con ellos para evitar algún contratiempo y en el peor de los casos deberá esperar a que le indiquen cuando debe presentarse, esto implica estar disponible en cuanto lo requieran.

Esta etapa es muy importante ya que se verá la capacidad de aprendizaje y organización del recurso, y dará la pauta para saber que conocimientos técnicos le hacen falta, para que sean reforzados durante la capacitación. En cada aplicación se tomará una introducción de su funcionalidad, se verá un

---

<sup>10</sup> Entorno de programación que proporciona al usuario herramientas necesarias para automatizar la administración de sistemas y facilitar la labores cotidianas.

panorama general de las actividades que se realizan y cuales serían las consecuencias en caso de algún error o problema.

*Profiler*: esta aplicación tiene guardado el perfil de cada una de las compañías, revendedores y contactos para la empresa en la que se está trabajando, al principio lo más importante es conocer la estructura de la plataforma, para esto se tienen dos ambientes: uno de pruebas y uno de producción, cada uno está dividido en dos partes, *Backend*<sup>11</sup> y *Frontend*<sup>12</sup> (ver figura 1). En el primero están todos los procesos utilizados para hacer las cargas de información que llegan a ésta, y también es donde reside la base de datos; en el segundo existe la aplicación de Internet con la cual se hacen las consultas a la información almacenada en la base de datos, por lo que en ambos entornos se tendrá algo similar.

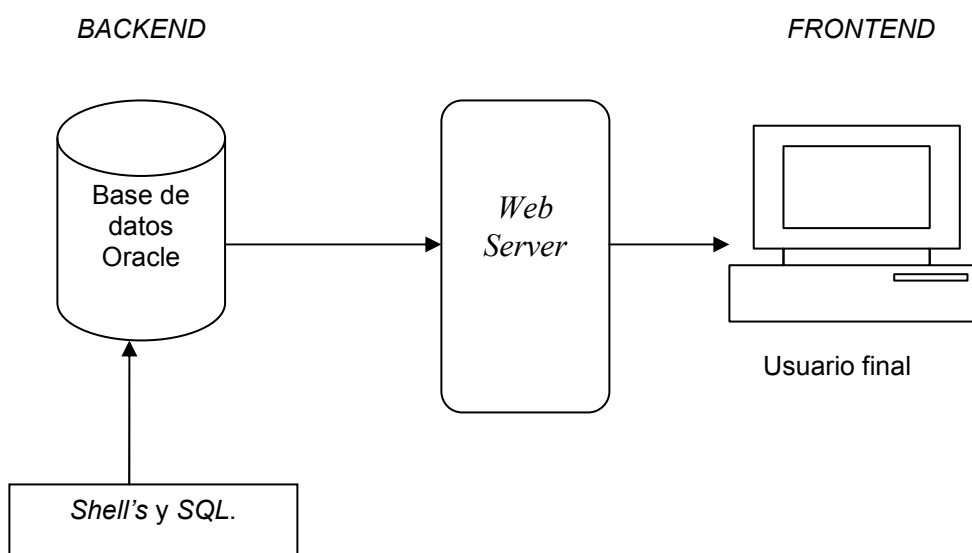


Figura 1. Diagrama de arquitectura de los servidores

La función del ambiente de pruebas, es ser el entorno donde se podrán hacer modificaciones a los objetos que conforman la aplicación, además crear nuevos elementos para alguna nueva funcionalidad, principalmente de la parte de *Backend*, para después ser puestas en producción, siempre y cuando hayan

<sup>11</sup> Encargado de procesar la información que el usuario ingreso desde el frontend.

<sup>12</sup> Responsable de recolectar los datos de entrada del usuario, que pueden ser de muchas y variadas formas y procesarlas de una manera conforme a la especificación que el backend pueda usar.

sido aprobadas y se haya validado su funcionamiento. Se recibe información de otras aplicaciones, y se procesa para ser mostrada desde la aplicación de Internet, también se generan datos para enviarse a otras aplicaciones. Esto dará una idea de cómo se trabaja en la plataforma y las actividades que se realizan cotidianamente.

Iniciemos con la base de datos<sup>13</sup>, se deberá hacer un análisis de todas las tablas que existen, buscando las relaciones que existen entre ellas, los programas que las puedan estar afectando (por ahora sólo se buscarán los *triggers* que han sido creados), con el objetivo de construir un diagrama entidad-relación el cual ayude a entender mejor la constitución de la base y así generar consultas de la información almacenada. Esto ayudará a saber como pueden verse afectados los objetos al realizar cambios dentro de la base de datos, cuando llegue a ser requerido por algún desarrollo, modificaciones al esquema e inclusive cuando se realice alguna transacción.

Ahora vayamos con el *Backend*, primero se verá la estructura de directorios que existe para poder identificar donde se encuentran los procesos de carga y extracción de datos, los archivos que se reciben de otras aplicaciones y los que son generados por los procesos del sistema, así como los *logs*<sup>14</sup> que crean éstos últimos, los cuales nos ayudarán a identificar muchos de los problemas que se llegan a presentar.

El siguiente punto está relacionado con los procesos que se ejecutan desde el *Backend*, esto es, analizar *Shell's* y procedimientos de *SQL*, para conocer su funcionamiento, ya que por lo general los usuarios cuando solicitan la ejecución de alguno de éstos, usarán un nombre funcional, por lo que es importante saber identificarlos, para lo cual se explicarán cada uno de los procesos involucrados, usando el ambiente de pruebas para ver como trabajan cada uno de ellos y saber de que forma los pueden llamar.

---

<sup>13</sup> Conjunto de datos que pertenecen al mismo contexto almacenados sistemáticamente para su uso posterior.

<sup>14</sup> Archivo en el que se almacenan todos los eventos ocurridos en un sistema informático.



Como se mencionó anteriormente, hay procesos de carga, los cuales están programados para ser ejecutados automáticamente; siempre se deben revisar los *logs* que generan para saber si las cargas fueron exitosas, así como la información cargada, en caso de que no sea así, hay que examinar el archivo que fue utilizado y verificar que no tenga datos inválidos, si esto sucede, hay que quitarlos e intentar nuevamente la carga, se tiene que hacer esto hasta que se hayan cargado registros válidos y los datos incorrectos se deben enviar a los dueños de la aplicación de donde proviene para que la revisen y los reenvíen corregidos, esto mismo se debe hacer para cada aplicación que envíe archivos.

Para los registros que se generan y se mandan a Conecta (aplicación que trabaja en otra plataforma<sup>15</sup>), por medio de archivos planos, si no fueron recibidos, se revisan los *logs* para buscar algún error, y si fueron generados correctamente, entonces se vuelven a enviar, en caso contrario se tiene que volver a ejecutar el proceso con el que se crearon, para detectar algún error si es que vuelve a fallar. Después que carguen la información, si llegan a tener algún problema nos dirán que registros son los que no fueron cargados y entonces deben corregirse, actualizándolos en la base de datos.

Para los casos antes mencionados, vamos a tener *Shell's* y bloques anónimos, con los que se va a estar trabajando; hasta que haya sido entendido su funcionamiento, el análisis que se haga debe ayudar a entender como trabajan y a solucionar muchos de los problemas que se puedan presentar como parte del soporte que se tiene que dar, aquí es donde se tiene que poner la mayor atención posible, ya que se concentran las tareas más importantes del soporte.

La parte del *Frontend* es más sencilla, ya que sólo se tendremos que asegurarnos que esté trabajando correctamente, y en caso de que haya algún problema se deberá reportar al equipo de soporte que está encargado de las dificultades que se presentan. Se utiliza cuando necesitemos ver la información de alguna compañía, revendedor o contacto y en caso de que precisemos modificar algo muy específico que se pueda hacer con esta herramienta, de lo

---

<sup>15</sup> Es el basamento, ya sea de hardware o software, sobre el cual un programa puede ejecutarse.

contrario nos ayudaremos con las utilerías para hacer consultas a la base de datos.

*Dataware House*: esta aplicación es: “una colección de información corporativa derivada directamente de los sistemas operacionales y de algunos orígenes de datos externos. Su propósito específico es soportar la toma de decisiones de un negocio.”<sup>16</sup>

Como en toda empresa, se tiene la necesidad de llevar un mejor control de las ventas que se realizan, para esto se utiliza el *Dataware House*, en el cuál se concentra información acerca de clientes, productos, órdenes, contratos y embarques. Esta información llega a la base de datos por medio de fuentes externas, esto es, de otras aplicaciones, las cuales mandan archivos planos con la información que se requiere cargar, después de ser procesada, se explota para tener una mejor idea de lo que está pasando y en que aspectos se puede mejorar. Estos datos son los más esenciales para las personas que dirigen el negocio por lo tanto, se le tiene que dar un formato amigable y entendible para el usuario, y de esta manera tome la mejor decisión.

Las tareas que se realizan en el soporte, de manera general, son: revisar que la información se recibió y fue procesada sin problemas, ejecutar los procesos para hacer esta actividad, revisar que los programas automáticos se hayan ejecutado y dar mantenimiento a la base de datos, apoyar a los usuarios que están encargados de la aplicación en los problemas que lleguen a tener y estar al pendiente del ambiente en el que está trabajando la aplicación.

Por lo tanto el *backup*, será capacitado por el titular de éste soporte y deberán revisar los puntos antes mencionados y de ser necesario abordar temas técnicos que le hagan falta para que pueda realizar las actividades lo mejor posible. Cuando tenga que cubrirlo, pueda realizar los casos que se mencionaron anteriormente y durante ese tiempo estará al frente y cumplirá con todas las tareas necesarias para el óptimo funcionamiento de la aplicación. En cuanto a las responsabilidades, serán casi las mismas, aunque el tiempo de respuesta no será el mismo, pero el servicio se debe cubrir sin problemas.

---

<sup>16</sup> Michael J. Corey, Michael Abbey. Oracle *Data Warehousing*. p.1

También es importante señalar que el *backup* sólo estará apoyando cuando se le necesite, esto es, cuando haya carga de trabajo o exista algún proyecto en el que pueda ayudar, sus tareas están en función a las solicitudes que se le hagan y para lo cual fue capacitado, o de acuerdo a sus nociones pueda cubrirlas sin problema.

Lo más importante para el recurso, es el aprendizaje que puede tener al estar trabajando con distintas personas y en puestos diferentes, esto le ayudará a su crecimiento profesional, pero dependerá del entusiasmo por cumplir con sus actividades y el provecho que obtenga de sus jefes.

## **2.2 Soporte a *Profiler***

La aplicación almacena la información de los socios de negocio de la empresa, esencialmente compañías, sitios y contactos, la cual puede ser consultada, modificada o agregada a través de una aplicación de Internet, también se pueden hacer actualizaciones o cargas masivas, a petición de los usuarios a los que se les da soporte, por medio de interfaces. En cuanto a los datos, de manera general es lo que se tiene que administrar, por lo que se tiene que vigilar la consistencia de éstos.

Dentro de las tareas técnicas a realizar está la de administrar la base de datos, esto es, darle mantenimiento a las tablas, índices<sup>17</sup>, *tablespaces*<sup>18</sup> y todos los objetos que la conforman. También se gestionan los directorios en el servidor donde está trabajando la aplicación, en este caso, se encuentra montada en un sistema operativo *Unix* y la base de datos es Oracle<sup>19</sup>, donde los más importantes son:

- a. *Bin.* - En éste se guardan todos los programas ejecutables.
- b. *SQL.* - Aquí se guardan los procesos de Oracle (Bloques anónimos).

---

<sup>17</sup> Estructura interna que el sistema usa para encontrar uno o más registros en una tabla de forma rápida.

<sup>18</sup> Objeto lógico que consiste en uno o más archivos de base de datos situados en el disco duro, éste almacena tablas u otros objetos.

<sup>19</sup> Sistema de administración de base de datos.

- c. *Log*.- Almacena los archivos que se generan para detectar errores o el éxito de los procesos que se ejecutan.
- d. *Dropbox*.- Guarda los archivos de entrada para las cargas.
- e. *Outbox*.- Guarda los archivos de salida que generan los procesos.

Estos son los directorios más importantes y con los que se estará trabajando para la parte de *Backend*.

Finalmente, se estará dando soporte a la aplicación *Web* la cuál si llega a fallar se reportará a otro equipo que está encargado de mantenerla funcionando. El soporte está enfocado principalmente a la parte de *Backend* que es donde se tiene la mayor responsabilidad.

### 2.2.1 Funciones y Actividades

Después de la capacitación, el *backup* ya puede cubrir al titular, por lo tanto ahora veremos las tareas que se realizan dentro del soporte y la manera de solucionar los problemas que se van presentando.

De las principales tareas que se realizan en el ambiente de producción, está la carga diaria de compañías, para esto los usuarios mandan un archivo con la información que requieren, y se deben realizar las siguientes actividades:

- a. Revisar que el formato del archivo sea el correcto y que no contenga errores, si llega a tenerlos se regresa al usuario para que lo corrija hasta que la información que trae sea consistente y entonces se podrá hacer la carga.
- b. Se guarda el archivo con separación por comas y se pone en el directorio *dropbox*.
- c. Se ejecuta el *Shell* de carga diaria de *Profiler*.
- d. Cuando haya terminado se revisa el *Log* que se genera para verificar que no hubo errores durante la carga, éste va mostrar cuantos registros fueron leídos con éxito, cuantos fueron rechazados y cuantos fueron actualizados por cada entidad que es modificada, en caso contrario se revisan los errores que arrojó, si fueron datos, se buscan los rechazados y se omiten del archivo. A continuación se describen los posibles errores que pudieron ocurrir:

- Violación de *constraints*<sup>20</sup>: ocurren al tratar de duplicar un valor, al almacenar información que excede el tamaño que tiene definido el campo en la tabla, el tipo de dato que se está insertando no corresponde al determinado para el campo.
- Crecimiento de *extents*<sup>21</sup>: sucede cuando una tabla ya no puede crecer más, y puede ser por dos razones: la cantidad que fue definida llegó a su límite, así que se tiene que modificar este atributo para corregir el problema; y la otra causa es porque el *tablespace* donde está, ya no tiene espacio, por lo tanto se tiene que aumentar el tamaño de éste para corregir el problema.
- La tabla está en modo ocupado: esto significa que está siendo bloqueada por otro usuario (que esté haciendo transacciones) y hasta que libere la tabla no se podrán hacer modificaciones en ella.
- Problema de conexión a la base de datos: éste puede ser ocasionado porque la contraseña del usuario de la base de datos caducó, entonces hay que cambiarla y en el *Shell* de carga se debe actualizar por la nueva.
- Usuario bloqueado en la base de datos, entonces se debe desbloquear para lograr el acceso.
- Base de datos en modo restringido: no se podrá acceder ya que esta en mantenimiento por lo que se tiene que esperar a que notifiquen que ya se puede hacer.

Solucionado el problema, se vuelve a ejecutar el *Shell* de carga.

- e. Finalmente se debe validar que el número de registros cargados correspondan con los que indica el *Log*, esto se hace a través de un campo de la tabla que guarda la fecha en que fueron creados; si fueron actualizados, se usa otro campo que sirve para saber cuando paso por última vez ya sea la compañía, el sitio o el contacto, si las cifras corresponden entonces se notifica al usuario para que dé su visto bueno de la solicitud.

El proceso antes descrito es el único que hace cargas de datos, los demás servirán para enviar información a otras aplicaciones y para esto se crearon

---

<sup>20</sup> Restricciones de integridad que hacen cumplir reglas de las actividades en el nivel de la base de datos, cuando se ejecutan instrucciones de ejecución, inserción y borrado.

<sup>21</sup> El almacenamiento de objetos es asignado en un número de bloques que deben ser contiguos en los archivos de la base de datos, los cuales son conocidos como *extents*.

interfaces, para Conecta<sup>22</sup>, hay una que le envía la información para ser cargada en sus tablas y ellos la puedan utilizar en su aplicación. El programa que lo hace, está automatizado y sólo se ejecuta una vez al día y por la madrugada, en ocasiones piden ejecuciones manuales porque corrigieron algún problema, para esto, se generan archivos planos que en el servidor de *Profiler* son guardados en el directorio *outbox* y son enviados a conecta por *ftp*<sup>23</sup> y dejados en el directorio *dropbox*, está es la funcionalidad que tiene el proceso pero para asegurarnos que todo se hizo con éxito se deben revisar los siguientes puntos:

- a. En las mañanas se debe revisar el *Log* de la ejecución para validar que todo se haya hecho exitosamente, si hubo algún error se notifica a los dueños de la aplicación o tal vez ellos avisen antes. Los errores que se pueden encontrar son:
  - En la base de datos, no se pueda lograr una conexión, se debe analizar cual fue la razón y corregir el problema.
  - En cuanto a problemas en el servidor: puede ser que el *filesystem*<sup>24</sup> donde se guardan los archivos se quedó sin espacio, por lo tanto, se deben depurar los archivos más viejos de los directorios de *logs*, *dropbox* y *outbox*, si es necesario se respaldan,
  - Falla al hacer la conexión por *Ftp*: esto puede ocurrir porque la contraseña del usuario, con el que se conecta, haya caducado, entonces se les debe notificar, para que nos proporcionen la contraseña correcta.
  - Sin espacio en el servidor de conecta: al hacer la conexión por *Ftp*, no hay espacio en el *filesystem* donde se dejan los archivos, por lo tanto se les tiene que avisar para que liberen un poco para enviar los archivos (en este caso se hará manual).
- b. Solucionado el problema, según sea el caso, se tendrá que volver a ejecutar la interfaz para que puedan hacer su carga.

---

<sup>22</sup> Aplicación para el acceso exclusivo de los distribuidores, revendedores y socios.

<sup>23</sup> Protocolo de transferencia de archivos.

<sup>24</sup> Un sistema de archivos que consta de tipos de datos abstractos que son necesarios para el almacenamiento, organización jerárquica, manipulación, navegación, acceso y consulta de datos.

Otro proceso es la interfaz con el *Dataware House*, la cual se ejecuta dos veces por día de manera automática, ésta envía información de los revendedores y para validar que se haya ejecutado exitosamente se debe revisar lo siguiente:

- a. Verificar el *Log* para saber si hubo errores durante la ejecución, si se presentaron problemas entonces se deben revisar los siguientes casos:
  - Problemas con la conexión a la base, se revisa el error y se corrige el problema.
  - Falta de espacio en el *filesystem*: se libera espacio.
  - Problema con la conexión por *Ftp*: se busca la causa y se debe corregir.
  - Contraseña invalida para usuario de *Ftp*: se les debe notificar para que envíen la contraseña correcta.
- b. Hecho lo anterior, se vuelve a ejecutar la interfaz para enviarles los archivos que están esperando.

Existe una interfaz con una aplicación llamada *Patsy*, a la cual se le envía información de las compañías, ésta se ejecuta una vez al día, y para verificar que no hubo problemas se debe hacer lo siguiente:

- a. Revisar el *Log* que genera la interfaz para saber si hubo errores durante la ejecución, si ocurrieron, pudieron ser originados por las siguientes causas:
  - Problemas con la conexión a la base: se busca el error y se corrige el problema.
  - Falta de espacio en el *filesystem*: se tiene que liberar espacio.
  - Conexión por *Ftp* fallida: se busca el problema y se corrige.
  - Contraseña del usuario de *Ftp* inválida: se les debe avisar para que envíen la contraseña correcta.
- b. Si el problema fue corregido se vuelve a ejecutar la interfaz para enviarles los archivos que no les llegaron.

Estas son las interfaces más importantes que se utilizan en la aplicación y que tienen que estar vigilando para que no haya errores, ya que si la información que es generada no les llega puede causar muchos problemas, por eso se debe poner la mayor atención posible. Hay un proceso más, que su función es la de encontrar los registros duplicados en la base de datos, básicamente lo que hace

es inhabilitar todos los registros replicados y guardar la información en una tabla llamada duplicados, como se puede ver este proceso ayuda a que la información sea lo más consistente posible, y no se manden datos inservibles a las demás aplicaciones.

Para las actividades diarias, existen tareas de mantenimiento las cuales ayudan a que la plataforma funcione correctamente y previenen problemas que afecten el funcionamiento de la aplicación. Es muy importante mantener la base de datos funcionando, lo más importante es revisar que los *tablespaces* no tengan problemas de espacio, si esto sucede se debe solicitar al equipo de soporte de la base de datos que aumenten el espacio de éstos. En caso de que alguno llegará a su límite de capacidad, ya no podrán crecer las tablas y esto puede provocar que haya fallas en el proceso de carga.

Diariamente se debe hacer un respaldo de la base de datos, éste servirá en caso de que se haya afectado información importante y necesiten quitar esos cambios, para esto existe un proceso que lo genera, se ejecuta por las mañanas, y se debe revisar si el respaldo se hizo correctamente, si hubo algún problema se busca lo que provocó la falla y se soluciona, después se vuelve a ejecutar el proceso. Cuando solicitan que se haga una recuperación de uno de estos respaldos se tiene que hacer lo siguiente:

- a. Tomar el último respaldo que se haya hecho.
- b. Desde el sistema operativo en la línea de comandos se ejecuta la instrucción: `imp usr/pwd@SID file=respaldo.dmp log= respaldo.log`, donde *usr* es el usuario para acceder a la base, *pwd* es su contraseña, en la instrucción *file* se indica el nombre que le fue asignado al respaldo y en *Log* se indica como se llamará el archivo donde se lleva el registro del comando que se ejecutó.
- c. Si todo fue exitoso, se revisa que la información se haya cargado correctamente. Este comando falla cuando no hay espacio en los *tablespaces* o no logra hacer una conexión a la base datos.

Cada mes se realiza una depuración de la información, esto es, se eliminan las compañías que han estado inactivas durante más de seis meses, y se debe



verificar que se eliminen sus sitios asociados, así como sus contactos, de esta manera no habrá registros inservibles en la base de datos.

Las tareas que se realizan en el ambiente de pruebas son similares, la diferencia radica que en fue diseñado para hacer los desarrollos solicitados (por usuarios) en los procesos de carga y extracción de datos, después se llevan a cabo las revisiones necesarias para poder ser puestos en producción los procesos, a continuación veremos cuáles son esas tareas y como se llevan a cabo las pruebas:

- a. Se atienden solicitudes de cambios en la funcionalidad de los procesos, se analiza el impacto y se notifica al usuario los cambios que se requieren hacer y el tiempo que toma realizarlos.
- b. Si está en proceso algún proyecto sobre la aplicación se da soporte, se auxilia en dudas que puedan tener con respecto a la funcionalidad de los programas en el sistema, los directorios en el servidor y problemas de conexión a la base de datos.
- c. Se participa en los proyectos desarrollando programas en *SQL* o *Shell*.
- d. Cuando algún proyecto lo requiere, se pone la información que existe en la base de datos de producción, esto es, se toma algún respaldo y se carga en el ambiente de pruebas.
- e. Para todo proyecto se revisa la funcionalidad de los procesos desarrollados o modificados antes de ser puestos en producción, y se hacen las pruebas necesarias para asegurarse que no haya problemas en el ambiente de producción.
- f. Se da mantenimiento a la base de datos, se verifican los *tablespaces*, tablas e índices para evitar retrasos en los proyectos que se estén llevando a cabo.
- g. Se revisa que los *filesystem* del servidor no tengan problemas de espacio, si esto ocurre, se debe liberar espacio.
- h. Se hacen cargas de datos con los procesos indicados, para esto se revisa que los archivos estén correctos y se haya cargado correctamente la información, por lo que se valida que en las tablas correspondientes los datos se muestren correctamente.

Básicamente éstas son las actividades que se llevan a cabo en cada ambiente de la aplicación, como se dijo anteriormente, el *backup* cubrirá esas tareas, y hay que recalcar que el titular siempre dará su apoyo para resolver todas las dudas, y la labor de éste será entender lo que se le explique, seguirse capacitando en cuestiones técnicas si es necesario y aprender todo lo posible.

## **2.3 Soporte al *Dataware House***

La aplicación concentra la información de ventas, inventarios, órdenes y embarques de los productos que vende la empresa a distribuidores de América Latina, ésta llega de varias aplicaciones en archivos planos y es cargada por un proceso que a la vez ejecuta otros cuatro más, lo cuales validan la que fue enviada, ya que está en la base de datos se verifica que hayan sido almacenados correctamente, por lo tanto su labor es administrarla y cuidar su integridad, ésta es la tarea más importante dentro del soporte y se debe realizar eficazmente.

Las tareas técnicas que se llevan a cabo son la de revisar que la base de datos esté trabajando, verificando que los objetos almacenados no tenga problemas de consistencia; que el servidor no tenga problemas de espacio para los archivos que son enviados; es importante mencionar que ésta aplicación está trabajando en un servidor *Unix* y con base de datos Oracle, para esto el *backup* después de ser capacitado deberá ser capaz de cubrir las actividades necesarias para mantener funcionando la aplicación.

Existen 2 ambientes funcionando para ésta aplicación: el de producción y el de pruebas, ambos contienen la misma información sólo que el segundo es usado para llevar a cabo los desarrollos que se llegan a solicitar para mejorarla.

### **2.3.1 Funciones y Actividades**

De las principales tareas que se realizan en el *Dataware House* son las cargas diarias que llegan de otros sistemas para alimentar la aplicación, la información es de ordenes, embarques, ventas e inventarios, para esto existe un proceso que hace la carga de esta información el cual ejecuta un proceso por cada tipo

de dato, por lo que se debe procurar que sean cargados correctamente, para lo cual se deben seguir los siguientes procedimientos:

- a. Revisión del proceso de carga diario, éste se ejecuta automáticamente por las mañanas todos los días, es el más importante ya que pone la información en la base de datos y para asegurar que todo se haya hecho correctamente se debe hacer lo siguiente:
  - Revisar el *Log* que genera el proceso, si no hubo problemas entonces el proceso fue exitoso, en caso contrario se deben revisar los errores que está arrojando, los cuales pudieron ser ocasionados por: Archivos que son enviados por otros sistemas y no llegaron al servidor correctamente por lo que se les debe solicitar que los envíen nuevamente; el *filesystem* del servidor se quedo sin espacio entonces se debe liberar lo suficiente y notificar que pueden volver a enviarlos; alguno de ellos tuvo problemas con la información, esto es, hubo registros con datos incorrectos y provocó el problema durante la actualización, por lo tanto no fueron cargados en la tabla correspondiente, entonces se manda un reporte al equipo de soporte de la aplicación que los envió para que los devuelvan corregidos.
- b. Una vez hecho lo anterior se procede a ejecutar nuevamente el proceso de carga, pero si sólo falló en alguna parte de esta entonces se ejecuta esa sección, y no el programa completo.
- c. Ya que la información esté en la base de datos se debe revisar que sea consistente, para esto se ejecutan consultas para contar el total de registros almacenados contra el total de datos enviados y estas cifras deben ser iguales, en caso contrario se debe revisar lo que hizo, se buscan los datos que no fueron cargados en los archivos recibidos y se vuelve hacer la carga, para encontrar el problema y buscar una solución.

Otras tareas consisten en la administración de la base datos para lo cual se deben realizar las siguientes actividades:

- a. Verificar que los *tablespaces* no tengan problemas de espacio, si esto llega a pasar se debe solicitar que aumenten su tamaño.
- b. Validar la consistencia de los objetos:

- Tablas, si requieren hacer un cambio en el esquema de la base de datos, se debe analizar si no impacta en los objetos que ya existen, para evitar problemas con los datos.
  - Agregar un nuevo *trigger*, una función<sup>25</sup>, un paquete<sup>26</sup> o procedimiento almacenado<sup>27</sup>, previamente se debieron hacer pruebas para validar su funcionalidad.
  - Índices, cuando es necesario tener un mejor acceso a los registros se agregan a los campos para lograr que éste sea más rápido.
  - Actualización de estadísticas de las tablas y los índices para evitar que las tablas tengan acceso lento.
- c. Se debe revisar el respaldo de la base de datos que se hace diario, en caso de que haya fallado se tiene que volver a ejecutar el proceso que lo crea, esto es importante por si se requiere en algún momento recuperar información.
- d. En caso de que se necesite recuperar alguna tabla se toma el respaldo que se realiza y se ejecuta la siguiente instrucción:
- ```
imp usr/pwd@SID file=respaldo.dmp log= respaldo.log tables=tabla
```
- Donde *usr* es el usuario para acceder a la base de datos, *pwd* es la contraseña de éste, *SID* es el nombre de la base de datos, *file* es el respaldo que se hizo, *Log* es el archivo donde se va a guardar el resultado de la ejecución y *tables* son las tablas que se requieren recuperar.

También se deben resolver algunos problemas y realizar las tareas, en el servidor donde está alojada la aplicación, que se muestran a continuación:

- a. Verificar el espacio disponible todos los días para poder recibir correctamente todos los archivos que llegan, en caso de que haya poco se deben borrar los archivos más viejos.
- b. Se deben respaldar los archivos que son recibidos.

---

<sup>25</sup> Conjunto de instrucciones en *PL/SQL*, que pueden ser llamados usando el nombre con que se le haya creado y retornan un valor. Las funciones retornan un valor al ambiente desde donde fueron llamadas.

<sup>26</sup> Son una forma que Oracle proporciona para encapsular estamentos y funciones *PL/SQL* en una entidad.

<sup>27</sup> Conjunto de instrucciones en *PL/SQL*, que pueden ser llamado usando el nombre que se le haya asignado. Los procedimientos pueden regresar o no, un valor.

- c. Cuando cambia la contraseña del usuario con el que accesan las aplicaciones se les debe notificar para que lo cambien en sus procesos.

Éstas son las tareas que debe realizar el *backup* durante la ausencia del titular, éste le puede solicitar información sobre algún proceso o actividad, pero se debe capacitar constantemente para que no tenga problemas, también es importante que el responsable capacite al recurso en caso de que surjan nuevas tareas.

Finalmente, para cualquiera de las 2 aplicaciones que va a llevar, siempre debe tener disponibilidad y capacidad suficiente, ya que puede haber ocasiones en que tenga que cubrir a ambos, aunque no es lo ideal puede llegar a pasar, para eso debe estar bien preparado y conocerlas lo mejor posible para no tener problemas. Tal vez el *backup* no tiene la suficiente experiencia, pero la va adquirir conforme vaya entendiendo mejor los procesos y las tareas que se realizan y puede abrirle las puertas para llevar alguna aplicación, para esto debe trabajar arduamente, y conseguir la confianza de los usuarios para que logre ese objetivo.

## Capítulo 3

### Soporte a la plataforma de *Broadvision*

#### 3.1 Antecedentes

Para la misma cuenta, requieren de un recurso que tenga conocimientos de administración de bases de datos, administración del sistema operativo *Unix* y si es posible, administración de *Broadvision*<sup>28</sup>. El recurso estará como *backup* del titular de la plataforma, por lo cual será capacitado para cubrir las actividades que se realizan cuando éste llegue a faltar, además, por petición del líder, por parte de la empresa, solicita que sea enviado a tomar cursos de administración de bases de datos Oracle, aunque antes será entrenado, por lo que debe organizarse ya que también esta apoyando en otras aplicaciones y debe tener tiempo para estas actividades y las de la nueva asignación.

La plataforma soporta las siguientes aplicaciones: *Conecta*, *PSIS*, *Reseller Web*, *Price List*, y *HPStore*, todas tienen diferentes funciones, de las cuales no hay que preocuparse, lo importante es verificar que el ambiente donde están trabajando lo haga correctamente, está es la principal función de la plataforma, proveer un entorno propicio para su buen funcionamiento, la cual está montada en un sistema operativo *Unix*, una base de datos Oracle y *Broadvision* que aplica las reglas de negocio necesarias en la manipulación de la información y como servidor de Internet se utiliza *iPlanet* de *Netscape*<sup>29</sup>.

Existen tres ambientes a los que se tiene que dar soporte, los cuales son:

- ❖ Desarrollo: este ambiente está conformado por un solo servidor el cual funciona como *Backend* y como *Frontend*, y todos los desarrolladores tienen permisos para realizar las actividades que requieran para llevar a cabo el soporte de sus aplicaciones o proyectos, la única responsabilidad que se

---

<sup>28</sup> Es una plataforma para administrar y desarrollar aplicaciones e-business que mantienen las relaciones entre el negocio y sus clientes.

<sup>29</sup> Solución de servidor de Internet de alta escalabilidad y flexible para entornos seguros SSL y no seguros.

tiene es de apoyarlos cuando tengan problemas ya sea con *Broadvision*, con la base de datos o con el servidor (ver *figura 2*).

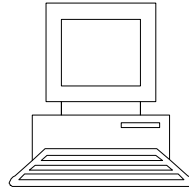


Figura 2. Ambiente de desarrollo, conformado por un solo servidor.

- ❖ Pruebas: está conformado por tres servidores, uno para *Backend* y dos para *Frontend* (a éstos se les conocerá como *Conecta* y *Clientes*), este ambiente es muy semejante a producción, la información será prácticamente la misma, al igual que los procesos que existen, los usuarios no tendrán muchos privilegios, lo que significa que hay más control sobre el entorno, lo único que podrán hacer son consultas. Se usa principalmente para hacer pruebas de todos los componentes que conforman las aplicaciones que están trabajando en la plataforma antes de ser movidos a producción, en el *Frontend* se revisarán los *logs* de *Broadvision* y para el *Backend* el desempeño. En la base de datos se administrará el espacio y se crearán nuevos objetos, para *Broadvision* e *iPlanet* se llevará su administración (ver *figura 3*).

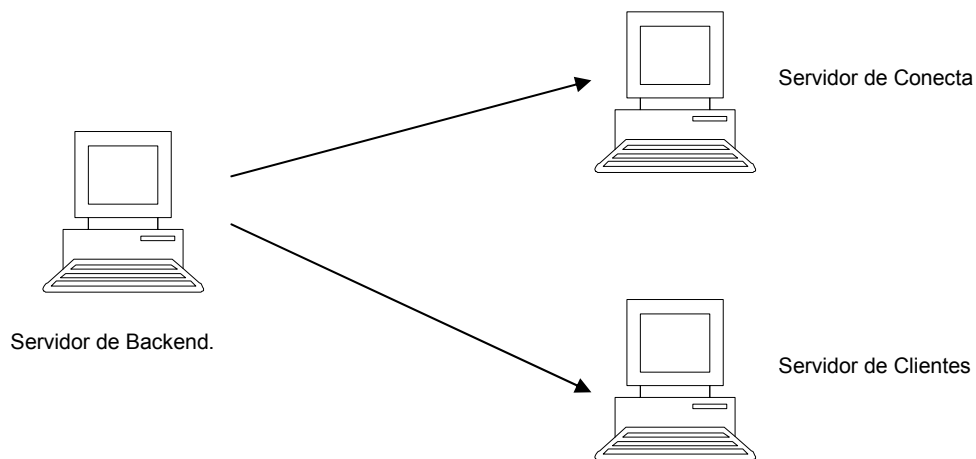


Figura 3. Ambiente de Pruebas

- ❖ Producción: para este ambiente existen cuatro servidores, dos de *Backend*, en donde existe un cluster el cual reside en uno de ellos y el otro funciona como *failover*<sup>30</sup>, y dos de *Frontend* donde uno es Conecta y el otro Clientes, éste es el ambiente productivo y se debe mantener funcionando para que las aplicaciones trabajen correctamente, esto es, la base de datos siempre debe estar disponible, los servidores no deben tener mucha carga de procesos, el servidor de Internet no puede dejar de funcionar y *Broadvision* siempre debe estar habilitado para la comunicación entre éste y el de *Backend* (ver figura 4).

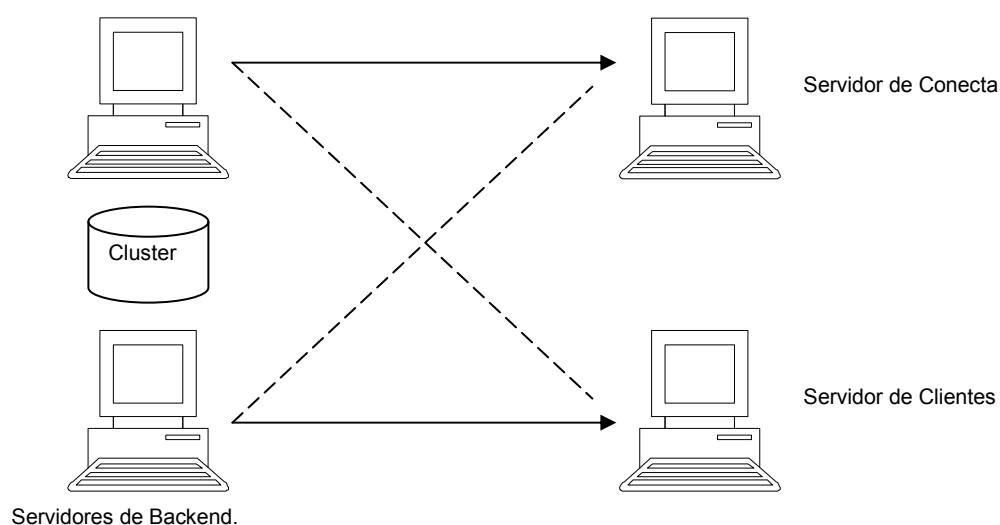


Figura 4. Ambiente de Producción.

Durante la capacitación el recurso conocerá a detalle las actividades que se llevan a cabo para que la plataforma funcione correctamente, verá las tareas administrativas en la base de datos, los servidores, *Broadvision* y del Servidor de Internet (*iPlanet*), el soporte para los responsables y los equipos de desarrollo en cada una de las aplicaciones. Cuando haya terminado su entrenamiento y a petición del usuario, tomará un curso de Administración de Bases de datos Oracle 8i, como requisito para que pueda suplir al titular.

Los cursos que tomará son de arquitectura y administración de base de datos y respaldo y recuperación, ambos para Oracle 8i, en el primero se conocerá como funciona todo el ambiente de Oracle, los demonios que lo hacen trabajar, y

---

<sup>30</sup> Es la capacidad de cambiar automáticamente a un servidor redundante o en espera.



cada una de las partes que lo conforman, se verán las principales tareas administrativas que se debe llevar a cabo para que siempre funcione y no tenga problemas, se crearán instancias nuevas. En el segundo curso se verán los tipos de respaldos que se pueden realizar y como recuperar la información en caso de algún problema.

Con lo aprendido en los cursos el recurso será capaz de realizar las actividades de administración para la base de datos en los tres ambientes, y cubrir con las expectativas del usuario con respecto al servicio que esta recibiendo. Por lo tanto, al tener cubiertos los conocimientos necesarios, ya puede fungir como *backup* del titular cuando éste llegue a faltar.

Las veces que tenga que cubrir al titular, deberá atender los requerimientos de cada equipo de soporte de las aplicaciones, tendrá que administrarse para poder atenderlos cuando necesiten ayuda, ya sea para hacer pruebas, ejecutar algún proceso, otorgar permisos en archivos o directorios, reinicio de servicios de *Broadvision*, compilación de procesos en Oracle y *Pro\*C*<sup>31</sup>, así como dar mantenimiento a los servidores, a la base de datos y *Broadvision*.

Se llevo a cabo un proyecto de estandarización de páginas para las aplicaciones que residen en la plataforma, para el cual fue requerido el apoyo del recurso donde sus actividades fueron revisar los *logs* de las páginas que se estaban modificando, por la cercanía de la fecha de entrega surgió la necesidad de trabajar fuera de horas de oficina, inclusive fines de semana para lograr la entrega del proyecto. Cada página que se revisa no debe arrojar errores, si esto pasa la tienen que volver a revisar, inclusive se tuvo que validar si el error existía desde antes, cuando ya no había problemas entonces se aprobaba. Si realizaban algún cambio en cualquiera que ya había sido aprobada, se tenía que volver a revisar. Al final del proyecto con las aprobaciones, se pusieron en el ambiente de producción y se revisaron por los desarrolladores, para verificar que estuvieran funcionando correctamente, también se movieron imágenes que eran utilizadas por éstas. A pesar de haber sido un proyecto agotador se logró

---

<sup>31</sup> Es la combinación entre el poder de un lenguaje de alto nivel como lo es C/C++ y la capacidad de manipulación en bases de datos de SQL.

el objetivo y se entregó el proyecto con menos retraso del que ya se tenía, y esto llevó al recurso a ser tomado en cuenta para otros proyectos.

El equipo de Lista de Precios solicitó soporte para después del horario de trabajo, por lo que contrato el servicio y con la aprobación del líder de la plataforma, se llevo a cabo, el cual consistía en ejecutar procesos de carga que duraban más de una hora cada uno, se realizaban respaldos de las tablas involucradas en los procesos, y si algo llegaba a fallar se recuperaba la información a las tablas originales; se les apoyaba si requerían hacer cambios en los procesos y se ponían en producción para ser ejecutados. También se tenía que trabajar fines de semana para dar este servicio, el cuál le otorgó el reconocimiento de los usuarios y del responsable de la plataforma.

Tiempo después por cambios dentro de la organización, la persona que está como titular del soporte, fue requerida para otra aplicación y gracias al desempeño que había tenido el *backup* en proyectos y en los servicios extraordinarios en los cuales fue requerido, le dieron la oportunidad para ser el titular del soporte de la plataforma y con esto iniciar una nueva etapa, con más responsabilidades y nuevas metas.

### **3.2 Funciones y Actividades**

Para los tres ambientes que se tienen, se debe vigilar que siempre estén en óptimas condiciones para lo cual se deben realizar tareas de mantenimiento que ayuden a que no fallen, también se deben realizar actividades de soporte para ayudar a los usuarios cuando requieran hacer pruebas, necesiten permisos para archivos, ejecutar procesos, compilar programas, etc. Esto es en general el servicio que se da dentro de la plataforma, desde que el recurso inicio como *backup* hasta que tomó la titularidad del soporte, ha realizado estas tareas y ha creado procesos para facilitar el trabajo, a continuación se verán con detalle las actividades que se realizan, para lograr el objetivo.

### 3.2.1 Tareas de gestión y de mantenimiento

Para que la plataforma funcione correctamente se deben llevar a cabo actividades que permitan corregir posibles problemas en su entorno y para cada ambiente se tiene que revisar las partes que la conforman, el ambiente más importante es producción, en el cual recae toda la carga de trabajo y donde el usuario final va a estar más atento a las fallas que ocurran; después está el de pruebas, el cual también es importante ya que en éste los equipos de soporte y desarrollo de cada aplicación estarán validando el funcionamiento de los componentes que requieren poner en producción, y es lo más semejante a éste ultimo; finalmente está el ambiente donde desarrollan, en el cual las tareas son menores, pero se debe cuidar que los programadores no lo afecten y por lo que se deben estar detectando problemas y en su momento solicitarles apoyo cuando sea necesario. A continuación se mencionan las tareas que se realizan en cada ambiente:

Base de datos: ésta es la fuente de información para todas las aplicaciones, por lo tanto el mantenimiento es frecuente y las actividades que se deben realizar son:

- a. Administración de los *tablespaces*. Se generaron varios *tablespaces*, principalmente para tablas e índices. Si se termina el espacio en alguno de ellos la instancia falla, esto es, la base se colapsa y todo el sistema deja de funcionar, por esta razón se debe revisar al menos una vez por semana para que no pase esto, si alguno llega a tener problemas se debe analizar cada objeto que contiene e identificar cuales pueden ayudar a liberar espacio. En el caso de las tablas, se siguen los siguientes pasos:
  - Identificar las que ocupan más espacio.
  - Borrar los registros viejos (con la previa autorización de su dueño).
  - Moverlas a un *tablespace* que tenga espacio suficiente (se debe hacer una por una).
  - Regresarlas al *tablespace* original.
  - Reconstruir los índices, ya que estos pierden su sincronización y ya no funcionan.

Con esto se libera espacio en donde se alojan las tablas y los índices, al reconstruir estos últimos se reduce su tamaño, pero de ello hablaremos más adelante. Para comprender mejor, al definir una tabla se le asignan parámetros de almacenamiento, los principales son: el tamaño que tendrá al crearse (inicial), es decir, reserva la proporción indicada y aunque se encuentre vacía la está ocupando, en “siguiente” se indica la extensión que va solicitar cuando necesite crecer, estos dos valores darán la pauta al crecimiento que tendrá. Cada una de ellas es definida de manera diferente, lo cual depende de los datos que estará almacenando, si alguna llena uno de los *tablespaces*, se produce el problema que se mencionó, por esta razón se tienen que hacer las actividades señaladas anteriormente para liberar área dentro de éste. La parte complicada en todo esto, es que se deben hacer fuera de horas de oficina o si el problema es muy grave, se debe solicitar el permiso a los dueños de las aplicaciones porque se puede bloquear y si algún proceso trata de almacenar información no lo podrá hacer, por la urgencia se debe hacer en el momento y dependiendo de su tamaño, es el tiempo que se puede tomar en realizarla. Para los índices, si alguno llega a tener problemas, basta con borrar registros en los objetos asociados y reconstruirlos, con lo que se logra el objetivo y se resuelve el problema. Para cualquiera de los dos casos, antes de borrar los datos se debe pedir la autorización a los dueños de las entidades, cuando se tenga su aprobación, se elimina la información más antigua o la que nos indiquen que puede ser borrada, pero antes se debe respaldar por si llega a ocurrir algún percance, hecho esto se realizan las actividades.

- b. Administración de tablas e índices. Frecuentemente se les debe dar mantenimiento, es decir, se deben actualizar sus estadísticas, lo que ayuda a que el acceso de estos objetos no sea lento. Para lo cual se hace un análisis y existen dos métodos: el de estimación y el de cómputo, el primero se hace con base en un porcentaje de registros y el segundo se basa en la recopilación de estadísticas, esta actividad va a depender de la necesidad, en caso de que algún objeto esté causando problemas se debe utilizar el método de estimación, el de cómputo se debe usar cuando no hay mucha carga de trabajo, para evitar bloqueos de objetos, ya que este último al ser ejecutado no permite que nadie más pueda realizar transacciones sobre el, hasta que termine de calcular las estadísticas. Esto ayuda mucho en los

tiempos de acceso a las consultas que se hacen a la base de datos y en la ejecución de procesos. Como parte de la administración, cuando se requiere crear una nueva tabla y sus índices, se deben calcular sus parámetros de almacenamiento, es decir, el tamaño inicial del objeto, que se calcula en base a la cantidad de registros que se almacenarán en la primer carga y la extensión que va a tener cuando solicite más espacio, se deduce de acuerdo al conjunto de datos que se van a estar almacenando después de la primer carga, también se indica en que *tablespace* va a ser guardado, dependiendo del tipo de objeto.

- c. Conexiones a la base de datos. Se tiene definido un número de usuarios que pueden estar conectados a la base, si se ocupan todos los accesos permitidos, entonces nadie más podrá acceder, este problema se presenta cuando alguna aplicación abre una conexión y al terminar su tarea no la cierra, entonces se van acumulando las conexiones hasta llegar al máximo, para resolver este problema, en el instante se deben eliminar las conexiones que tienen más tiempo ó tienen estado de inactivas. El impacto que puede tener es en los procesos de carga que existen, si alguno llega a fallar puede impactar a las aplicaciones. Para solucionar este problema se realizó un programa que monitorea las conexiones abiertas en la base de datos y elimina todas las que tengan dos días de antigüedad, de esta manera se tiene un mejor control de los accesos y se asegura la accesibilidad.
- d. Administración de Usuarios y permisos. Para tener acceso a la base de datos se necesita un usuario para realizar operaciones sobre la instancia, el cual es creado de acuerdo a las necesidades del equipo que lo solicite. Dependiendo del objeto es el permiso que se va a otorgar, los más comunes son:
  - Tablas: lectura, escritura, actualización y borrado.
  - Funciones, procedimientos almacenados y paquetes: ejecución y *debug*<sup>32</sup>.
  - Usuarios: el permiso más importante es el de conexión, aunque existen otros como el de *quota*, creación de tablas, otorgar permisos, etc., otro que es muy importante es de DBA, se usa para administrar los objetos, crear usuarios, compilar programas, por lo regular es el dueño de todas

---

<sup>32</sup> Proceso de remoción de errores en el software o en los datos.

las tablas que existen. Como gestor podrá definir que es lo que podrán hacer los equipos de desarrollo dentro de la base de datos.

Como parte del soporte se debe tener el control completo de la base de datos, y los usuarios sólo podrán hacer consultas y en casos especiales, se les concederán permisos de ejecución, actualización, borrado e inserción.

- e. Administración de Procedimientos. Para cada aplicación existen programas que ayudan a darle consistencia a la información que es almacenada en las tablas, pero los dueños de las aplicaciones y de estos objetos no pueden modificarlos, sólo el administrador puede hacerlo, ya sea por algún cambio o para crear un nuevo programa. La única tarea que se realiza es compilar procedimientos almacenados, funciones, paquetes y disparadores, si en alguna ocasión una tabla es modificada en su estructura, los objetos se descompilan, por lo tanto es necesario volver a compilarlo, para que no se vea afectado su funcionamiento.
- f. Respaldo y recuperación. Es necesario respaldar de la base de datos al menos cada tercer día, por lo que se realizó un programa que es ejecutado automáticamente durante la madrugada, se necesita por si llegará a ocurrir algún problema con la información de las tablas o se pierda, y puedan recuperarse, aunque implique pérdida de información dependiendo del último que se hizo correctamente. El proceso que se ejecuta lo realiza con una utilería de Oracle llamada *export*, ésta respalda la información mientras la instancia esta funcionando, y la herramienta para hacer la recuperación se llama *import*. Cuando se va a ejecutar una actualización, inserción o borrado, y van a afectarse muchos registros se procede a respaldar las tablas involucradas, por si algo llegará a pasar y se necesiten deshacer los cambios, se tomará uno anterior para recuperar la información. En cuyo caso se puede hacer creando una nueva tabla con el mismo esquema que la original o utilizando las utilerías antes mencionadas.

Servidores: conforman la parte más importante de la plataforma, ya que en ellos se aloja la base de datos, *Broadvision*, *iPlanet* y todas las utilerías necesarias para el funcionamiento de las aplicaciones, las tareas necesarias para que los servidores funcionen adecuadamente son las siguientes:

- a. Administración de espacio. Se deben verificar los *filesystems* de los usuarios que se utilizan en cada servidor, regularmente se borran archivos viejos que son generados por los procesos de las aplicaciones, se tiene una política de mantener máximo dos meses los archivos que se han guardado, después de ese tiempo se empiezan a eliminar, está actividad se realiza cada mes. También se tienen detectados otros que se pueden llenar fácilmente por la cantidad de información que se guarda, estos se revisan semanalmente para evitar un problema mayor. Si alguno de ellos se quedará sin espacio, causaría serios problemas en el ambiente, porque ya no recibirían archivos de entrada de sistemas externos y mucho menos podrían generar los archivos de salida, lo cual provocaría retrasos que perjudicarían el funcionamiento de la plataforma y de los sistemas externos.
- b. Administración de Procesos. Cada aplicación tiene programas trabajando en la plataforma, algunos son para almacenar información en la base de datos, otros extraen información que es enviada a otros sistemas, los cuales pueden llegar a consumir muchos recursos y pueden volver inestable el ambiente. Existe una lista de procesos que están programados para ser ejecutados en un día y hora específicos, los que han sido clasificados de acuerdo al consumo de recursos que requieren y su importancia, sólo cuando se necesita volver a ejecutar un proceso se hace manualmente a solicitud del dueño de la aplicación.
- c. Administración de componentes. En los servidores hay archivos binarios, planos, de *Log*, *SQL*, páginas de Internet, librerías, de configuración, etc., para estos se debe tener un control de permisos, lo que significa que no todos los usuarios pueden ejecutar, leer o escribir sobre ellos. Si llegan a necesitarlos, sólo se les otorga de lectura, a menos que sea en el ambiente de pruebas, se le podría otorgar de escritura o ejecución, aunque sería temporal. En algún caso especial se les llega a conceder de ejecución sobre procesos muy específicos y que de antemano se sabe que no causarían problemas en el ambiente de producción.
- d. Administración de usuarios. En los ambientes de producción y pruebas, los usuarios están programados para que cambien de contraseña cada 3 meses, cuando expira el de alguno de ellos, se procede a generar uno nuevo. Si el usuario es interno, se actualiza en un archivo de variables de ambiente para que los procesos no fallen, en caso de que sea restringido y

sólo lo utilicen los equipos de soporte de las aplicaciones, se les notifica que cambió para que usen la nueva contraseña. En el entorno de desarrollo pueden solicitar un nuevo usuario para algún proyecto nuevo, en este caso se debe pedir a otro equipo su creación, una vez que lo hayan hecho se les manda el nombre con el que fue creado y la contraseña que fue asignada; para los demás, cuando la contraseña expira se cambia y se notifica al equipo correspondiente, aunque puede ser que ya lo hayan hecho.

- e. Instalación de utilerías. Cuando se necesita instalar nuevas herramientas para un nuevo desarrollo, se solicita información sobre ésta, se debe validar que exista una versión compatible con el sistema operativo y también que sea aprobada por el personal de soporte de la empresa para evitar problemas de seguridad, si se cumple con eso se coloca la herramienta, primero en el ambiente de los desarrolladores, si funciona bien se procede a instalar en el entorno de pruebas, y hasta que se vaya a poner en producción la aplicación para el cuál lo solicitaron, se instala en este ambiente.

*Broadvision*: esta es la solución que maneja las reglas de negocio para las aplicaciones que radican en la plataforma. En los ambientes de producción y pruebas, las tareas consisten en reiniciar el ambiente completo, cuando han hecho algún cambio en el esquema de la base de datos, o por alguna librería nueva que va a ser utilizada por las páginas. Sólo en desarrollo, cuando se crea un usuario nuevo y necesita tener su propia instancia, se hace la configuración necesaria, esto es, instalar y configurar el núcleo de *Broadvision*, crear las instancias para la comunicación entre el *backend* y el *frontend*.

*iPlanet*: este es el servidor de Internet que se utiliza en la plataforma, existen otros servidores que establecen comunicación con éste, cuya funcionalidad es la autenticación de los usuarios que se conectan a las aplicaciones, entonces cuando están en *failover*, es necesario reiniciarlo para que establezca la comunicación con el otro servidor. Existen dos puertos configurados, el puerto 80 y el puerto seguro, ambos deben ser reiniciados cuando suceda lo antes mencionado.



### 3.2.2 Tareas de soporte

Cualquier problema que afecte las aplicaciones y tenga que ver con la plataforma deben ser atendidos por el responsable de ésta, y todo lo que se refiere a la funcionalidad lo deben resolver los equipos de soporte de cada aplicación, estos pueden ser ocasionados por algún error que no detectaron los desarrolladores en su momento por lo tanto no es responsabilidad del equipo, las obligaciones se limitan a problemas con la plataforma, para cada ambiente cambian mucho. Por lo tanto veamos por ambiente cuales son las actividades a realizar.

Ambiente de desarrollo: aquí los desarrolladores, tienen permiso para realizar cualquier actividad sin restricciones, por cada aplicación hay configurado un ambiente de *Broadvision*, para que puedan trabajar sin afectar a los demás equipos. Para la base de datos, tienen acceso al usuario administrador, por lo cual, las actividades se reducen considerablemente, y el apoyo que se les da consiste en lo siguiente:

- a. Soporte a la base datos. Cuando no pueden crear una tabla, índice o procedimiento, se revisa lo que están haciendo y se corrigen los errores que tienen; en otros casos si se queda atorada alguna sesión, se mata; también se pueden quedar bloqueadas y se procede a matar el proceso que la esta afectando. Muchas veces se les da consultoría cuando tienen algún problema que no pueden resolver.
- b. Soporte a *Broadvision*. El problema más común, es cuando reinician los servicios y no los pueden levantar, en este caso se revisa si algún proceso sigue funcionando estando abajo todo, se mata, y se vuelven a iniciar hasta que queda solucionado el problema.
- c. Soporte a *iPlanet*. Generalmente llegan a tener problemas con la instancia que existe, ya que todos utilizan la misma para trabajar, por lo que cuando llega a fallar se debe reiniciar para que funcione nuevamente, cuando no se resuelve el problema, se notifica a otro equipo de soporte para que reinicie todos los servicios y ellos solucionan el problema.
- d. Soporte a *Unix*. Los problemas más comunes son cuando tratan de compilar un programa y no lo pueden hacer, entonces se revisa la instrucción que

están ejecutando y si tiene algún problema se corrige, aunque muchas veces son problemas con el código fuente del programa y por esa razón no lo pueden hacer.

- e. Creación de usuarios en el sistema operativo. Cuando un equipo de desarrollo, inicia un nuevo proyecto, llega a solicitar la creación de usuarios, esto se solicita el equipo de soporte de *Unix*, en cuanto notifican que ya fue hecho, se hacen cambios al perfil del usuario, que en estos casos llegan a pedir la configuración de un nuevo ambiente de *Broadvision*, entonces se genera y configura para que se pueda conectar al servidor de *iPlanet* y que los servicios e *Interaction Manager*<sup>33</sup> estén funcionando correctamente, junto con ello solicitan que se pongan todas las páginas que existen en producción las cuales se toman de ese ambiente, y al final todo debe trabajar sin problemas, hasta entonces se les manda la información de nombre de usuario y contraseña para que puedan ingresar y verificar que está configurado su nuevo ambiente.

Ambiente de pruebas: aquí se revisan desde desarrollos nuevos hasta pequeños cambios realizados a cualquiera de los componentes del ambiente productivo. Las tareas a realizar son:

- a. Pruebas de páginas. En el servidor de Internet existen páginas dinámicas (JSP), las cuales utilizan funciones que son almacenadas en otros archivos (JS), éstas hacen consultas a la base de datos y utilizan elementos de *Broadvision* para poder mostrar y manipular la información que solicita el usuario final. Por otra parte *Broadvision* genera unos *logs* que muestran los errores que pueden tener, por lo tanto, cuando cualquier equipo haga un cambio en ellas, tiene que solicitar la aprobación técnica, para poder ser puestas en el ambiente de producción. Para lo cual se lleva un control de versiones de las páginas, y para solicitar la aprobación, deben mandar la versión del componente a revisar y se les notifica en que momento se pueden llevar a cabo. Durante las pruebas se revisan los *logs* mientras se navega, si no aparece errores entonces se aprueba, si llega a mostrar algún error deben corregir el problema, de otra manera no se les puede dar el visto

---

<sup>33</sup> Procesan las peticiones y crean las páginas de respuesta a través de procesos llamados páginas de generación dinámica (JSP).

bueno, cuando lo hayan hecho, deben volver a mandar la nueva versión del componente para repetirlas. Si resultaron correctas entonces se manda la aprobación y pueden solicitar que se mueva a producción ese componente. Las funciones que utilizan las páginas (JS) requieren del reinicio de los *Interaction Manager* ya que se cargan en la memoria del servidor para ser utilizadas, cuando se hace esto, se revisa que no haya errores en el *Log* y que estén funcionando bien todos los servicios, inclusive durante la navegación se pueden probar, con lo que se asegura que no haya problemas con estos componentes.

- b. Pruebas de componentes de *Backend*. Los procesos que se revisan son: *Shell's*, *SQL's*, *Pro\*C*, java, los cuales también están en el control de versiones y cuando requieren revisarlos para obtener la aprobación se verifica que no afecten el desempeño en el ambiente. Generalmente los *Shell's* llaman un *SQL* o un binario (*Pro\*C*), por lo tanto cuando se ejecuta se revisan ambos, se toma el tiempo que tarda y se identifica el proceso en el servidor para monitorear que no esté tomando mucha memoria o CPU, estos siempre deben estar optimizados, para evitar problemas de lentitud, en caso de que no cumplan con este requisito no se les da la aprobación hasta que consigan que consuman pocos recursos. Los java y los *Pro\*C* deben ser compilados antes de probarse, ambos generan un binario, el cual es ejecutado desde el *Shell* correspondiente, a partir de este paso se pueden detectar fallas ya que pueden compilarse erróneamente y por lo tanto no pueden ser revisados hasta que se genere el ejecutable correctamente. Los Java generalmente funcionan como demonios<sup>34</sup>, y son llamados al realizar alguna consulta desde una página o al llegar archivos a un directorio en específico, entonces se debe conseguir que esto ocurra para que puedan probarse dichos programas, se monitorea su consumo de memoria y CPU, si no causan problemas se aprueban, en caso contrario se deben corregir.
- c. Pruebas de componentes de base de datos. Estos pueden ser funciones, paquetes, procedimientos y *triggers*. Los primeros pueden ser llamados desde alguna página o desde algún procedimiento del *Backend*, cuando se ejecutan se debe monitorear su consumo de recursos y el tiempo que toman para terminar, si existen problemas deben corregirlos para que obtengan la

---

<sup>34</sup> Programa que ejecuta o lanza un servicio en segundo plano.

aprobación. Los *triggers* son ejecutados al realizar una transacción sobre la tabla asociada, si al realizarse ésta, no ocurre ningún error entonces se aprueba el componente, de lo contrario deben corregir el problema. También al solicitar las pruebas deben enviar la versión que corresponda a cada objeto.

- d. Pruebas de componentes de *Broadvision*. Para el manejo de la información, se utilizan contenedores, para crearlos se debe hacer lo siguiente: los desarrolladores envían dos archivos con su respectiva versión, uno conocido como *make* y el otro como *source*, se ejecuta un comando para compilarlos y como resultado se obtendrá un *SQL*, el cual tendrá instrucciones de inserción y borrado de registros a tablas de control propias, y también las sentencias para crear tablas y sus índices. El archivo generado, se debe complementar con los parámetros de almacenamiento para las tablas e índices, hecho esto, se ejecuta desde *sqlplus* todas las sentencias que contiene, con esto se crea el nuevo contenedor, pero no podrá ser visto hasta que no se hayan reiniciando completamente los servicios. Si se presenta algún problema durante la compilación o la ejecución, lo deben corregir; cuando se haya creado exitosamente se dará la aprobación. El caso más sencillo es cuando se realiza la modificación a algún contenedor, en este caso sólo contiene sentencias de borrado e inserción, sin embargo se debe reiniciar todo el ambiente para que se tomen estos cambios.
- e. Requerimientos de la base de datos:
- Solicitudes para acceso de lectura, escritura, actualización y borrado a las tablas: si son compartidas con otras aplicaciones les puede dar esos permisos, en caso contrario no se podrá realizar el requerimiento.
  - Permisos de ejecución para los paquetes, funciones y procedimientos: no se les pueden otorgar a todos, sólo en casos que realmente lo requieran y no impacten a objetos que sean compartidos.
  - Compilación de procedimientos: esta tarea es muy común cuando se están realizando cambios a procesos existentes o nuevos.
  - Creación de tablas: deben justificar la necesidad o si es para alguna nueva funcionalidad de la aplicación donde lo solicitan.
  - Creación de índices: generalmente van asociados a las tablas cuando son creadas, pero también se da el caso que soliciten crearlos sobre un

campo que no lo tiene, para mejorar los tiempos de respuesta de las tablas.

- RespalDOS de tablas: cuando van a realizar alguna carga masiva, solicitan que sean respaldadas las que van a ser afectadas, y en caso de que tengan algún problema piden que se recupere la información.
- f. Solicitudes para archivos. Generalmente solicitan permisos para lectura de los *logs* que genera *Broadvision*, para revisarlos mientras navegan en su aplicación, y en un caso urgente se otorgan de escritura a páginas que necesitan modificar para resolver algún problema. Los desarrolladores de todas las aplicaciones podrán hacer uso de ciertos usuarios para ver todos estos archivos, pero no tendrán suficientes privilegios, ya que han sido creados con el fin de tener un control total sobre los componentes.
- g. Solicitudes de *Broadvision*:
- Reinicio de Servicios: una tarea que se realiza frecuentemente es cuando han hecho cambios a archivos de funciones (JS) en el servidor de Internet y para que sean cargados requieren el reinicio de los *Interaction Manager*, o cuando han hecho algún cambio en el esquema se debe reiniciar todo el ambiente previa notificación a todos los usuarios, ya que los equipos de desarrollo pueden estar trabajando y si no son notificados se les puede retrasar en su trabajo, hasta que se recibe la aprobación de todos se lleva a cabo esta actividad.
  - Cargas de datos: muchas veces se requiere cargar información a tablas de *Broadvision*, para esto existe una utilería para realizarla, el procedimiento para hacerlo es el siguiente: se ejecuta una serie de sentencias para eliminar datos que pueden afectar la carga, después desde el sistema operativo con el usuario administrador, se ejecutan los comandos que toman los archivos donde está la información a ser cargada que al finalizar dejan un *Log*, se revisa, sin o con errores se prosigue con los demás archivos, de lo contrario no se puede continuar hasta que se haya resuelto el problema. Al final, cuando se hayan hecho las cargas, se debe ejecutar un proceso para reconstruye las vistas materializadas para que la información pueda ser vista desde la aplicación.

- *Command Center*: para la manipulación de los datos existe una herramienta llamada *DCC*, los requerimientos consisten en crear usuarios y otorgar permisos a contenedores.
  - Limpieza de *cache*: en el *Frontend* existe un proceso que refresca el *cache*<sup>35</sup> de *Broadvision*, esta solicitud la hacen cuando no pueden ver algún cambio que hicieron con la herramienta mencionada en el punto anterior.
- h. Ejecución de procesos. Esta tarea consiste en ejecutar *Shell's* cuando se están realizando pruebas, para mostrarle al usuario final el funcionamiento de su proceso.
- i. Problemas de desempeño. Muchas veces los procesos que están ejecutándose en el ambiente empiezan a tomar muchos recursos y lo vuelven muy lento, se busca el que esta causando problemas y se mata, si es muy repetitivo, se busca el padre, si es ejecutado automáticamente se quita hasta que se estabilice. Otra causa, puede ser que alguna tabla o índice necesite ser analizado, se realiza esta tarea y se vuelve a ejecutar el programa, se monitorea para saber si se resolvió. Pueden existir muchas causas, las mencionadas anteriormente son las más comunes, cuando el entorno es lento siempre se deben matar los eventos que están tomando recursos, hasta que se encuentre la fuente del error.
- j. Problemas de espacio en disco. Es difícil saber el tamaño que puede requerir un archivo generado por alguno de los procesos y muchas veces llenan los discos, cuando un usuario notifica que no se generaron sus archivos lo primero que se revisa es el espacio que tiene el disco, si esta lleno se libera borrando los objetos más viejos o en su defecto se respaldan los más grandes en otro disco antes de ser borrados. Con esto se soluciona el problema, cabe mencionar que a pesar de realizar tareas de mantenimiento para evitar este tipo de problemas, suelen ocurrir.
- k. Creación de variables de ambiente. Todos los *Shell's* que existen en el servidor, usan este tipo de variables, por lo tanto cuando requieren cambiar el valor a una de ellas, o crear nuevas, hacen su solicitud y se realiza esta tarea.

---

<sup>35</sup> Memoria que se sitúa entre el procesador y la memoria para acelerar los intercambios de datos.

- I. Problemas con *iPlanet*. Muchas veces la instancia del servidor de Internet falla, este problema lo detectan los usuarios cuando tratan de ingresar a las páginas y estas no son encontradas, para resolver este problema se deben reiniciar los servicios para el puerto 80 y el seguro. Puede pasar que solo el nodo seguro deje de funcionar y por lo tanto sólo se reinicia este último.

Ambiente de Producción: es el ambiente más importante, ya que en el residen todas las aplicaciones y por lo tanto los usuarios finales realizan muchas de las actividades de negocio dentro de la empresa, por eso es fundamental mantenerlo trabajando correctamente y cuando sea requerido apoyarlos, siempre va a ser prioridad sobre los demás. Las tareas que se llevan a cabo son las siguientes:

- a. Problemas de desempeño. Existe un demonio que mata los procesos de Oracle que toman muchos recursos, éstos sólo pueden estar funcionando durante 10 minutos ocupando un 100% de CPU, al llegar a este límite se eliminan y si es muy repetitivo, se les notifica a los dueños de éste, para que ya no lo ejecuten porque está causando problemas y se les pide que lo hagan después de horas de oficina, sólo en caso de que sean cargas muy importantes entonces se les permite que sigan realizando esta actividad, para lo que se apaga el programa que los monitorea, y cuando terminan se vuelve a habilitar. Generalmente el primer día de cada mes, hay dificultades de este tipo, inclusive llegan afectar a otras aplicaciones. Cuando es muy lento el ambiente, se tiene que matar todo lo que está tomando muchos recursos hasta que se normalice, también se deben realizar algunas tareas de mantenimiento a las tablas e índices, para buscar corregir el problema, aunque muchas veces no depende de esto, sino de la cantidad de información que se está cargando, y se tienen que poner de acuerdo los dueños de las aplicaciones para llevar a cabo sus actividades y evitar que se siga afectando a los demás.
- b. Problemas de espacio. Como ya se mencionó en el punto anterior, esos procesos no sólo llegan afectar al ambiente sino también al espacio en disco del servidor, si se generan o reciben archivos muy grandes, se puede terminar el espacio, y se deben buscar los de mayor tamaño para ser eliminados. Para algunas aplicaciones como la de *PSIS*, que crean mucha

información al mes, se llegó a un acuerdo para que su información se mantengan cuando mucho un mes, por lo que todos los que tengan más tiempo se eliminan.

- c. Ejecución de consultas. Cuando requieren cambiar la información a un usuario, o cualquier registro en alguna de las tablas de la base de datos, hacen esta solicitud, antes de ejecutarlos, se respaldan las tablas afectadas, por si llegará a haber algún problema y se necesite recuperar.
- d. Creación de usuarios de base de datos. Los dueños de las aplicaciones puede solicitar que se genere un nuevo usuario de base de datos, después de justificar su solicitud se procede a realizar la tarea. Éste es creado con permisos para conexión y de lectura a las tablas que haya solicitado.
- e. Permisos a objetos de la base de datos. Generalmente los usuarios, tienen la necesidad de consultar tablas a las que no tenían permiso, para esto el dueño de la aplicación hace la solicitud y se les otorga. En el caso de que requieran ejecutar procedimientos, actualizar, borrar o insertar, primero se verifica que la tabla o procedimiento no sea compartido y segundo deben justificar el requerimiento, si no cumplen con esos dos puntos, no se les concede el permiso, en caso contrario, se les otorga bajo reservas de que lleguen a abusar, y se les revocará.
- f. Ejecución de procesos. Suele pasar que algún proceso falle, si esto sucede, solicitan que se ejecute manualmente el programa para detectar la causa del problema, pudo ser por algo que este mal con los datos que esta tomando o que estaba consultando alguna tabla bloqueada en el momento en que se ejecutó, en estos casos piden que se monitoree el proceso durante su ejecución para detectar el problema y poder corregirlo. Estas solicitudes siempre son hechas por los dueños de las aplicaciones.
- g. Permisos de *Log*. En el *Frontend* hay *logs* de *Broadvision* que muestran los problemas que tienen las páginas, éstos se llegan a crear sin permisos y para poder verificar que error está ocurriendo, solicitan acceso de lectura y se les concede. Los procesos de *Backend* también generan *logs*, cuando necesitan revisar si terminaron correctamente solicitan acceso de lectura para poder hacerlo.
- h. Variables de ambiente. La mayoría de las aplicaciones externas, cambian sus contraseñas cada determinado tiempo, cuando esto sucede mandan la solicitud para modificar el valor en el archivo de variables de ambiente. Lo



mismo pasa cuando expira la de alguno de los usuarios de sistema, inclusive si se trata del usuario de base de datos.

- i. Procesos de *cron*. En *Unix* existe una herramienta para que los procesos se ejecuten automáticamente en determinada fecha y hora, ésta se conoce como *crontab*, todos los procesos que se ejecutan en el ambiente están programados con esta utilidad, cuando un usuario necesita cambiar la hora de ejecución, deshabilitar o programar un nuevo proceso, hacen su petición y se hacen los cambios solicitados.
- j. Solicitudes de *Broadvision*. Los requerimientos más comunes consisten en cargas de datos, administración de *DCC* y depuración de memoria temporal, a continuación se describe cada uno de ellos:
  - Carga de información: regularmente las cargas que realizan son de productos, para hacer esto se llevan a cabo los siguientes pasos: se ejecutan sentencias para eliminar datos preexistentes, se guardan los archivos que contienen la información en el servidor de *Backend*, después desde el sistema operativo con el usuario administrador se ejecutan los comandos para hacer la carga lo cuales dejan un *Log*; se verifica que no haya habido errores, de lo contrario no se podrá continuar hasta que haya sido resuelto el problema. Cuando se hayan terminado, se deben refrescar las vistas materializadas que contienen la información válida y pueda ser vista desde la aplicación.
  - *Broadvision* tiene una herramienta llamada *DCC*, con la cual se manipula la información contenida en la base de datos, los requerimientos consisten en crear usuarios y otorgar permisos a contenedores (siempre y cuando pertenezcan a la aplicación de quien los solicita).
  - Depuración de memoria temporal: en el *Frontend* existe un proceso que depura la memoria temporal, el cual está programado en el *cron* y se ejecuta en distintos horarios, cuando realizan algún cambio desde el *DCC* y necesitan verlo reflejado en la aplicación solicitan que se corra manualmente.
- k. Problemas con *iPlanet*. Es raro cuando falla el servidor de Internet, cuando esto sucede, todos los usuarios notifican el problema, se puede deber a algún error con la misma instancia, por lo que se tienen que reiniciar los servicios para solucionarlo, tanto para el puerto 80 como para el seguro. Pero la causa más común es cuando *HPP* (este es un servidor de

autenticación de usuarios) falla, se pierde la comunicación y por lo tanto la instancia deja de funcionar, en este caso se depende considerablemente de su tiempo de respuesta para resolverlo; cuando vuelve a trabajar, se inicia la instancia de *iPlanet* y todo vuelve a funcionar correctamente. Es importante mencionar que ambos servidores deben permanecer comunicados, de lo contrario la plataforma falla y sólo estará funcionando la parte de *Backend*.

I. Movimientos a producción. Cuando solucionan problemas de funcionalidad de las aplicaciones o van instalar proyectos nuevos, los equipos de cada una de ellas deben conseguir la aprobación técnica (Soporte de *Broadvision*) y del usuario final (Solicitante), cuando las tengan deben solicitar vía correo electrónico al responsable de la plataforma el requerimiento, anexando las aprobaciones y versiones correspondientes, para que puedan ser atendidos, si algo de esto llega a faltar se regresa su solicitud. Habiendo cumplido con los requisitos previos se realizan los cambios solicitados y se hace lo siguiente:

- Crear archivos BKE y LST: el primero es para los procesos de *Backend* (.BKE) y el segundo para los de *Frontend* (.LST), dentro de ellos se pone el directorio donde se va a alojar, el nombre del archivo y la versión, todo separado por un espacio, se pueden agregar todas las líneas que sean necesarios.
- Ingresar al ambiente de pruebas con el usuario para movimientos a producción, en raíz hay dos directorios importantes: MTP y *bin*, en el primero se ponen los archivos que se crearon con anterioridad, en el segundo se ejecuta el *Shell* que hace el movimiento, se revisa el *Log* que genera para saber si se hizo correctamente, si estuviera mal alguna ruta, nombre de archivo o la versión, no se moverá nada hasta que todas las líneas estén bien dentro de los archivos, cuando indique que todo se movió correctamente, se ingresa al servidor de *Backend*.
- Otorgar los permisos necesarios a los archivos: para el caso de los *Shell's* y binarios es muy importante que tengan permisos de ejecución de lo contrario no funcionarán y los *SQL* deberán tener permisos de lectura.
- Compilación de *Pro\*C*: en este caso se tienen que compilar en el ambiente de pruebas, ya que en producción no existen las utilerías

necesarias para hacerlo, después se mueve el binario por *Ftp* a producción.

- Compilación de objetos de Oracle: los procedimientos almacenados, paquetes, funciones y disparadores se compilan desde *sqlplus*.
- Compilación de Java: se compilan con un *Shell* que generan, donde incluyen las librerías necesarias, y al ser ejecutado se genera el binario.
- Cambios al esquema de *Broadvision*: se ingresa con el usuario administrador para compilar los archivos que generan el *SQL* que contiene tablas e índices, se les agrega los parámetros de almacenamiento, se ingresa a *sqlplus* y se ejecuta las sentencias generadas, cuando se hayan aplicado los cambios, se debe reiniciar todo el ambiente, sólo que es más complicado hacerlo porque dejaría de funcionar uno de los servidores, además de que toma más tiempo y esto afectaría a todas las aplicaciones, por lo que generalmente se hace después de horas de oficina, para evitar cualquier contra tiempo. Para los componentes de *Frontend*, ya que estén los archivos en los servidores de Internet, se ingresa a ellos y se verifican los permisos, se les debe otorgar de ejecución; cuando se trata de funciones (JS), para que sean tomadas requieren el reinicio de los *Interaction Manager*, antes de hacerlo se debe poner en *failover* el servidor que lo requiere y al finalizar se quita, también hay *CGI's*, estos cuando se mueven no necesitan ningún tipo de configuración.

Estas son las actividades de soporte que se realizan dentro de la plataforma, muchos de los problemas que suelen presentar no son predecibles, por lo que en ocasiones los usuarios son los que terminan reportándolos y se tienen que atender en el momento, sobre todo si es en producción, hay que hacer un análisis exhaustivo para determinar la causa y solucionarlo, en este tipo de casos no se tiene un tiempo determinado para dar solución, sino que la respuesta se debe dar lo más pronto posible, porque el ambiente no puede estar abajo, aquí se depende mucho de las habilidades y los conocimientos del recurso para resolverlos, siempre se trabaja en equipo para encontrar más rápido las soluciones.

La parte complicada del soporte se basa en dar solución a los problemas que llega a tener la plataforma, los cuales generalmente son causados, por la base de datos, *Broadvision*, *iPlanet* o el sistema operativo, pero existen otros equipos de soporte por parte de la empresa, que están monitoreando los servidores y cuando notan que existe alguna falla la reportan. Mandan la solución (instalación de un nuevo parche, problemas de espacio, problemas de hardware, etc.) y dan fechas para llevarlo a cabo, se les notifica a los dueños de las aplicaciones para que estén enterados del mantenimiento que se les estará dando a la plataforma.

Finalmente, siempre se debe tener buena disponibilidad después de horas de oficina y fines de semana, en caso de que algún acontecimiento llegará a pasar en la plataforma se debe estar al pendiente para mantenerla siempre en línea y así brindar un buen servicio a los usuarios.

## Reflexiones

Cuando inicie mi carrera profesional, llegue con ganas de sobresalir, buscando ser el mejor, sin embargo tropecé con mucha competencia, personas muy buenas con los que había rivalidades sanas, pero siempre hubo apoyo, empezamos a trabajar en equipo y formamos grupos de estudio, si alguien era bueno en matemáticas, nos ayudaba en esa materia, si otro era bueno para programar nos apoyábamos en él cuando teníamos dudas. Fue una buena experiencia que al final nos ayudo a todos salir adelante. También conocí maestros muy buenos, como el de cálculo diferencial, siempre nos motivaba para que entendiéramos y pensáramos como resolver los problemas, el de programación que siempre nos explicó de una manera muy simple y nos decía que no nos agacháramos cuando trabajáramos con personas de otras universidades, y así tuve varios maestros de los cuales aprendí bastante y gracias a ellos he logrado cumplir con muchos de mis objetivos.

En los últimos semestres de la carrera quería dedicarme a la seguridad informática, siempre me ha llamado la atención, pero en realidad nunca investigué a fondo el tema, puesto que sólo conocía algunas cosas que había leído acerca de los *Hackers* y *Crakers*, con el tiempo supe que había mucho más que aprender y finalmente lo deje a un lado para hacer otras cosas. Algo importante que debo recalcar sobre estudiar, es que, no sólo se trata de pasar las materias, sino de sacar provecho de lo que se imparte en clase, porque si no después te arrepientes y te gustaría volver a tomarlas.

Después de haber cursado todas las materias, en los cinco años que duró la carrera, mi objetivo era titularme en el menor tiempo posible y pensé en hacer una tesis sobre un traductor de lenguas romances, a pesar de haber hecho una buena investigación, por cuestiones laborales y personales no pude concluirlo, fue hasta hace poco que se abrieron nuevas opciones de titulación, que opte por hacer un informe sobre mi experiencia profesional.

Durante el lapso de tiempo, desde que termine la carrera hasta que inicie este informe, el primer año estuve buscando trabajo, pero no encontraba nada, en todos lados pedían experiencia, la cual aún no tenía, precisamente por eso lo

buscaba, luego un momento en que me pregunté si no había errado de profesión, entonces decidí buscar empleo como profesor de computación en una escuela, así fue como inicié mi vida profesional, impartiendo clases de paquetería principalmente, estuve cerca de un año haciendo esto, hasta que se presentó una oportunidad de empleo, un amigo me llamó para avisarme que estaban solicitando personal en Softtek<sup>®</sup>, como becario de Oracle, por lo que siguiendo su recomendación me llamaron a entrevista, tuve que presentar algunos exámenes psicométricos y de conocimientos, éste tipo de pruebas eran un dolor de cabeza, tuve que investigar en Internet como tenían que ser resueltos y encontré bastante información, así cuando llegue a presentarlos no tuve complicaciones, después de unos días me avisaron que había sido aceptado.

Los cursos que impartieron fueron sobre: *SQL y PL/SQL, Unix y programación Shell, Forms y Reports*. Al finalizar cada curso nos hacían una evaluación, para conocer nuestro avance y si alguien obtenía malas notas, le notificaban que si continuaba así, podía quedar fuera, y eso pasó con varios compañeros. Tal vez lo complicado fue el poco tiempo que nos dieron para aprender estas herramientas, y a pesar que algunos ya tenían nociones no lograron quedarse, porqué, además de las calificaciones, tomaron en cuenta el trabajo en equipo y la rapidez de aprender y solucionar problemas. Cuando nos dieron los resultados, empezaron a llamarnos, al final quedamos siete y fuimos aceptados para iniciar nuestra vida laboral en Softtek<sup>®</sup>. Al inicio nos comentaron que íbamos a estar a prueba por tres meses para conocer nuestro desempeño y si llegábamos a fallar nos despedían.

De esta manera nos fueron integrando en las cuentas donde se da servicio, fui el último en ser asignado, prácticamente lo que hacen decirte que necesitan que te presentes con algún cliente y da una explicación sencilla de lo que vas hacer y por cuanto tiempo, me tocó ir a una empresa de telefonía celular, me presente con la líder de un equipo de soporte, quien me comentó cuales eran las tareas que tenía que realizar, me asignaron un equipo y empecé por entender el requerimiento, hasta que concluí una base de datos que servía para capturar todos los requerimientos que eran solicitados y con base en la información capturada obtener métricas del servicio que se estaba dando,

cuando terminé ese desarrollo, regrese a las oficinas de Softtek<sup>®</sup> y de inmediato fui asignado a otra cuenta.

Para mi segunda asignación, me comentaron que necesitaban a alguien con mi perfil y me mandaron a una empresa que se dedica al desarrollo de nuevas tecnologías para equipos de cómputo, me presente con la gerente de la cuenta y me dijo cuales iban a ser mis actividades, después fui presentado con las personas que iba a trabajar. El puesto por el que llegué era de *backup*, tuve que capacitarme con tres personas, cada una llevaba una aplicación, mi labor consistía en cubrir sus ausencias por enfermedad, vacaciones, etc., ya que el servicio no podía desatenderse, así que tuve que aprender administración de bases de datos y *Unix* a nivel administrativo, no hubo tantas tareas de programación, sin embargo fue un reto que logre cumplir, no siempre recibes el apoyo de las personas con las que trabajas. Es cuando entiendes que te estás enfrentando al mundo y aprendes a sobrevivir, la parte más difícil, fue sobrellevar a quienes no me aceptaban como recurso porque no tenía el conocimiento suficiente, aunque sabían que estaba ahí para aprender y que estaba dispuesto a hacerlo, aun así, tuve que luchar el doble para mostrar mi capacidad y conseguir su respeto.

La plataforma de conecta fue una de las más complicadas, ya que solicitaron que tomará cursos de administración de base de datos de Oracle, para que pudiera fungir como *backup*, a pesar de que ya había tomado la capacitación, no podía cubrir al titular, por lo que me mandaron a cursos y los cuales aproveche bastante, porque llevaba una idea más clara de lo que era y saque provecho del instructor. Cuando regrese de los cursos, pude trabajar en la plataforma cuando me requerían, hubo periodos largos hasta de 1 mes, en los que estuve al frente, fue un poco complicado al principio, pero con el tiempo fui tomando confianza y experiencia. Participe en un proyecto, haciendo pruebas de páginas, tenía que monitorear que no arrojaran errores, y por la magnitud, debía estar muchas horas frente a la computadora mientras otra persona navegaba, inclusive trabajamos fines de semana, con lo cual me gane el reconocimiento del cliente.

Después, el dueño de la aplicación de lista de precios, necesitaba soporte después de las horas de oficina, y por la confianza que había ganado, permitían que me quedaría a apoyar, muchas veces era salir de madrugada y al otro día estar temprano en la oficina, inclusive los fines de semana tenía que trabajar, pero esto me ayudo, ya que, cuando al titular del soporte decidieron moverlo a otra aplicación, me dieron la oportunidad de estar al frente, lo cual fue un reto más. Al principio hubo mucha la presión, todos los usuarios esperaban el mismo tiempo de respuesta que les daba la otra persona, algunos eran comprensivos, pero la mayoría querían las cosas al instante. Al principio me falló la administración de los requerimientos, pero aprendí y empecé a llevarlos mejor, cuando dominé la plataforma todo fue más sencillo.

El primer proyecto grande que tuve que migrar a producción, fue con un hindú, esta persona hablaba demasiado rápido y me costo mucho trabajo entenderle, por lo cual tuve problemas con ese proyecto, se tuvieron que volver a realizar los pasos para instalarlo, esta vez estuvo el encargado de esa aplicación y se consiguió instalar con éxito el proyecto, fue una buena experiencia y como muchos dirían fue mi novatada. Conforme el tiempo paso fui adquiriendo experiencia y eso me ayudo a consolidarme en el soporte, logré el reconocimiento de los usuarios y del cliente, con lo cual supe que estaba haciendo las cosas bien.

Algo curioso que quiero mencionar es que nunca reconocieron mi progreso ni el esfuerzo en Softtek<sup>®</sup>, y no es el único lugar donde pasa esto, lo satisfactorio es saber que estoy dejando una buena imagen y esto me puede ayudar cuando me cambie de proyecto, o se acabe en el que estoy, siempre hay que buscar crecer profesionalmente y no quedarse estancados, a veces los cambios dan miedo, pero hay que saber enfrentarlos y siempre buscar nuevos retos.

Finalmente, puedo decir que he logrado mucho a lo largo de los años que llevo trabajando y que si antes no pude titularme no fue por falta de ganas, sino falta de tiempo, trabajar en sistemas consume mucho tiempo, pero lo importante es terminar esta etapa, para buscar nuevos objetivos. No me imaginaba trabajando en una empresa como está, mucho menos como DBA, mi idea era diferente,



puedo decir que me gusta mucho lo que hago, administrar no es tarea fácil y Oracle es un monstruo, y siempre tiene retos nuevos.

Espero les sirva mucho lo comentado en este trabajo y que ayudé a entender a quienes apenas van empezando, saber de que se trata esta carrera, y que no todo son Matemáticas o Electrónica, en lo personal me gustan mucho, pero habrá quienes prefieran programar.

## Bibliografía

- Advanced information systems, Inc. *La biblia de Oracle 8*  
Madrid, España, Ed. Anaya Multimedia, 1998, 918 pp.
- Urman, Scott. *Oracle 8i Programación Avanzada con PL/SQL*  
Madrid, España, Ed. McGraw-Hill, 2001, 778 pp.
- Pérez López, César. *Oracle 10g Administración y Análisis de Bases de Datos*  
Madrid, España, Ed. Alfaomega Ra-Ma, 2004, 671 pp.
- Corey, Michael J.; Abbey, Michael. *Oracle Data Warehousing*  
España, Ed. McGraw-Hill, 1997, 344 pp.
- BroadVision®, Inc  
*BroadVision One-To-One® Enterprise, Overview*

### Referencias electrónicas:

- [http://www.gwolf.org/soft/tut\\_shell/tut\\_shell.pdf](http://www.gwolf.org/soft/tut_shell/tut_shell.pdf), 2002. Gallardo, Rodrigo  
Sección: Shell de hoy.
- [http://www.camaraalcoy.net/Servicios\\_web/glosario/Glosario/L.htm](http://www.camaraalcoy.net/Servicios_web/glosario/Glosario/L.htm), 2006.  
Sección: *Log File*.
- [http://es.wikipedia.org/wiki/Sistema\\_de\\_ficheros](http://es.wikipedia.org/wiki/Sistema_de_ficheros), 2006.  
Sección: Sistema de archivos.
- <http://en.wikipedia.org/wiki/Failover>, 2006.  
Sección: *Failover*.

- <http://es.wikipedia.org/wiki/Cache>, 2006.

Sección: Memoria Caché.

- <http://es.wikipedia.org/wiki/Software>, 2006.

Sección: *Software*.

- [http://es.wikipedia.org/wiki/Base\\_de\\_datos](http://es.wikipedia.org/wiki/Base_de_datos), 2006

Sección: Base de datos.

- [http://es.wikipedia.org/wiki/Front-end\\_y\\_back-end](http://es.wikipedia.org/wiki/Front-end_y_back-end)

Sección: *Front-end* y *Back-end*.

- <http://www-db.stanford.edu/~ullman/fcdb/oracle/or-proc.html>, 2006.

Sección: *Overview Pro\*C*.

- [http://es.wikipedia.org/wiki/Demonio\\_%28inform%C3%A1tica%29](http://es.wikipedia.org/wiki/Demonio_%28inform%C3%A1tica%29), 2006.

Sección: Demonio (Informática).