

UNIVERSIDAD DE SONORA

División de Ingeniería
Departamento de Ingeniería Industrial

SOPORTE TECNICO A LOS PROCESOS DEL MAINFRAME

Trabajo Profesional

TUDO · LO · ILUMINAN

Que para obtener el título de
INGENIERO EN SISTEMAS DE INFORMACIÓN

Presenta

GABRIEL VELDERRAIN MITANI

Director

DR. JOSÉ LUIS OCHOA HERNÁNDEZ

Hermosillo, Sonora.

Septiembre 2022

Universidad de Sonora

Repositorio Institucional UNISON



**"El saber de mis hijos
hará mi grandeza"**



Excepto si se señala otra cosa, la licencia del ítem se describe como openAccess



EL SABER DE MIS HIJOS
HARÁ MI GRANDEZA

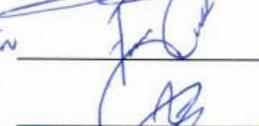
UNIVERSIDAD DE SONORA
DIVISION DE INGENIERIA
DEPARTAMENTO DE INGENIERIA
INDUSTRIAL

Hermosillo, Sonora., a 23 de Agosto del 2022

Dr. René Francisco Navarro Hernández
Coordinador del Programa de Ing. en Sistemas de Información
Presente.-

Por este conducto, hago de su conocimiento que estoy de acuerdo que se realice el examen profesional del alumno GABRIEL VELDERRAIN MITANI con expediente No. 209214200 el cuál será el día _____ en la plataforma TEAMS a las _____.

RELACION DE JURADOS:

	NOMBRE	FIRMA
PRESIDENTE:	<u>Dr. José Luis Ochoa Hernández</u>	
SECRETARIO:	<u>DR. FEDERICO MIGUEL CIRETTI GALÁN</u>	
VOCAL:	<u>Dr. Gerardo Sanchez Schmitz</u>	
SUPLENTE:	<u>Dr. René F. Navarro H.</u>	<u>René F. Navarro</u>

ATENTAMENTE

MIEMBROS DEL JURADO

Formato FT4.

1. RESUMEN

En este documento se exponen la práctica y experiencia laboral a partir de las tareas que ejecuta un ingeniero en sistemas de información en la empresa Tata Consultancy Services en un puesto en el que se le brinda soporte técnico a los procesos ejecutados por un mainframe, se describen algunas de las situaciones más comunes que se presentan diariamente en una jornada laboral normal en un entorno de tecnologías de la información, se presentan detalles de los procedimientos que se llevan a cabo, así como las herramientas utilizadas en el proyecto y se hacen recomendaciones para las empresas involucradas para la mejora de sus procesos.

2. DEDICATORIAS

Este trabajo está dedicado a:

A mis padres Prospero Velderrain Breach(†17 de junio de 2013) y Ana Maria Mitani Ricci, por sus consejos y palabras de aliento.

A mi novia Elide Marlene Gómez Andrade, por llegar a mi vida y motivarme a obtener mi título universitario.

Los amo.

Y por supuesto a mi querida Universidad de Sonora junto con todos los maestros con los que tuve el placer de tomar clases por haber compartido sus conocimientos a lo largo de mi carrera.

3. AGRADECIMIENTOS

Es para mí un honor utilizar este espacio para expresarles mis agradecimientos a las personas e instituciones que me han facilitado las cosas para poder realizar este trabajo y por fin después de varios años poder buscar mi título profesional.

Debo agradecer de manera especial al Profesor José Luis Ochoa Hernández por asesorarme durante todos mis estudios universitarios y años después de haberlos concluido para guiarme durante el proceso de la realización de este trabajo, su capacidad para guiar mis ideas como estudiante y hoy como profesional ha sido de gran aporte. Le agradezco también el haberme facilitado durante mis estudios soluciones a problemas que por ser alumno irregular se me presentaban semestre tras semestre. Muchas gracias, Profesor.

Quiero extender un sincero agradecimiento a la Universidad de Sonora por haberme permitido formarme en ella y todos los profesores que tuve el gusto de haber sido su alumno, gracias por todo el conocimiento que me brindaron, por su paciencia y orientación.

También quiero agradecer a Tata Consultancy Services, trabajar en esta empresa representa un gran paso en mi desarrollo profesional y no puedo más que expresar mi agradecimiento por la confianza que me han brindado para desarrollarme profesional y laboralmente.

Y, por supuesto, el agradecimiento más profundo es para mis padres, que a pesar de la distancia siempre han estado a mi lado, dándome ánimo para continuar, y aunque mi padre ya no está a mi lado, todo el tiempo que pudimos compartir me anima a seguir adelante, les agradezco infinitamente porque sin ustedes simplemente, no estaría aquí.

4. CONTENIDO

RESUMEN	ii
DEDICATORIAS	iii
AGRADECIMIENTOS	iv
CONTENIDO	v
LISTA DE FIGURAS	vii
1 CAPITULO 1 INTRODUCCIÓN	1
1.1 PLANTEAMIENTO DEL PROBLEMA	2
1.2 OBJETIVO GENERAL	3
1.3 OBJETIVOS ESPECÍFICOS	4
1.4 ALCANCES DEL PROBLEMA Y DELIMITACIONES	5
1.5 JUSTIFICACIÓN	5
2 CAPITULO 2. DESCRIPCIÓN DEL CONTEXTO	7
2.1 DESCRIPCIÓN DE LA NORMATIVIDAD O REGLAS DE OPERACIÓN DE LA EMPRESA	7
2.2 ENTORNO DONDE SE UBICA LA EMPRESA	8
3 CAPITULO 3. FUNDAMENTO TEÓRICO DE LAS HERRAMIENTAS Y CONOCIMIENTOS APLICADOS	9
3.1 HERRAMIENTAS UTILIZADAS EN LA SOLUCIÓN	12
3.1.1 Microsoft Windows 10	12
3.1.2 z/OS	12
3.1.3 Control-M	13
3.1.4 Servicenow (SNOW)	13
3.1.5 Zoom	14
3.1.6 Slack	14
3.1.7 Suite Microsoft Office	15
4 CAPITULO 4. PROPUESTAS DE SOLUCIÓN AL PROBLEMA O PROBLEMAS PRESENTADOS	16
4.1 EJEMPLO DE CANCELACIÓN #1:	16
4.2 EJEMPLO DE CANCELACIÓN #2:	22
4.3 EJEMPLO DE CANCELACIÓN #3:	28
4.4 EJEMPLO DE RETRASO #1:	32
4.5 Ejemplo de retraso #2:	35
4.6 Ejemplo de retraso #3:	37
5 CONCLUSIONES Y RECOMENDACIONES	42

5.1	CONCLUSIONES SOBRE EL PROCESO	42
5.2	CONCLUSIONES PERSONALES	42
5.3	RECOMENDACIONES	43

5. LISTA DE FIGURAS

Figura 1.1 Logo de TCS.	1
Figura 1.2 Logo de IBM.	3
Figura 3.1 IBM z15 mainframe.	10
Figura 3.2 JCL.	11
Figura 3.3 Logo de Windows 10.	12
Figura 3.4 Logo de z/OS.	13
Figura 3.5 Logo de Control-M.	13
Figura 3.6 Logo de Servicenow.	14
Figura 3.7 Logo de Zoom.	14
Figura 3.8. Logo de Slack.	14
Figura 3.9. Logo de la Suite Microsoft Office.	15
Figura 4.1. Alerta de fallo en Control M – Ejemplo de cancelación #1.	18
Figura 4.2. Registro de errores z/OS – Ejemplo de cancelación #1.	18
Figura 4.3. Data Set Utility – Ejemplo de cancelación #1.	20
Figura 4.4. Administrador de archivos Z/OS – Ejemplo de cancelación #1.	20
Figura 4.5. Documentación en SNOW - Ejemplo de cancelación #1.	21
Figura 4.6. Alerta de fallo en Control M – Ejemplo de cancelación #2.	24
Figura 4.7. Registro de errores z/OS – Ejemplo de cancelación #2.	24
Figura 4.8. SuperC - Ejemplo de cancelación de cancelación #2.	25
Figura 4.9. SuperC resultados – Ejemplo de cancelación #2.	26
Figura 4.10. Documentación en SNOW - Ejemplo de cancelación #2.	27
Figura 4.11. Control M – Ejemplo de cancelación #3.	29
Figura 4.12. Registro de errores z/OS - Ejemplo de cancelación #3.	30
Figura 4.13. Administrador de archivos z/OS – Ejemplo de cancelación #3.	30
Figura 4.14. Documentación en SNOW – Ejemplo de cancelación #3.	31
Figura 4.15. Control-M – Ejemplo de retraso #1.	32
Figura 4.16. Diagrama de flujo en Control-M - Ejemplo de retraso #1.	33
Figura 4.17. Diagrama de flujo en Control-M con jobs completados- Ejemplo de retraso # 1	33
Figura 4.18. Control-M – Ejemplo de retraso #2.	35
Figura 4.19. Diagrama de flujo en Control-M con información de espera- Ejemplo de retraso # 2.	36
Figura 4.20. Control-M – Ejemplo de retraso #3.	37
Figura 4.21. Artículo (KB) en SNOW - Ejemplo de retraso #3	38
Figura 4.22. Búsqueda de archivos en Z/OS - ejemplo de retraso #3.	39
Figura 4.23. Data sets encontradas - Ejemplo de retraso #3.	39
Figura 4.24. Petición para correr job en SNOW - Ejemplo de retraso #3.	40
Figura 4.25. Documentación en SNOW - Ejemplo de retraso #3.	41

1 CAPITULO PRIMERO. INTRODUCCIÓN

TCS es una multinacional india de consultoría y servicios de tecnología de la información (TI) con sede en Mumbai. Es la segunda empresa india más grande por capitalización de mercado y se encuentra entre las marcas de servicios de TI más valiosas del mundo. Es parte del Grupo Tata y opera en 149 ubicaciones en 46 países, siendo en Guadalajara donde desempeño mis actividades, aunque el lugar donde me desenvuelvo y llevo a cabo el proceso productivo es en mi departamento, ya que desde que empezó la pandemia, la empresa nos facilitó laptops para continuar con nuestras labores desde casa, practicando el distanciamiento social sin dejar de estar conectados [13].

Dentro de Tata Consultancy Services (TCS) el trabajo que he desempeñado en los últimos años es dentro del área de soporte técnico, donde he tenido la oportunidad de realizar el análisis de diferentes situaciones en la que se presentan fallos en los procesos batch ejecutados por la computadora principal también conocida como mainframe, llevando a cabo adecuadamente los controles establecidos para mitigar los diferentes riesgos a los que se encuentra expuesto el cliente de la empresa para la que trabajo.



Figura 1.1 Logo de TCS.

Como parte de los servicios que ofrece TCS y en específico, el área asignada de trabajo, es brindar servicio a empresas para el desarrollo, soporte y mantenimiento de tecnologías de información. Las funciones que se llevan a cabo implican trabajar en un entorno informático de clase empresarial, incluidas las series de sistemas IBM Z, realizar análisis del rendimiento del sistema, buscar problemas y/o solucionarlos a medida que surgen y algunas otras más que se explicaran en las siguientes partes del documento.

1.1 PLANTEAMIENTO DEL PROBLEMA

¿Qué es un proceso batch (batch job)?

Se conoce como batch a los procesos que se pueden ejecutar sin la interacción del usuario final, o que se pueden programar para ejecutarse según se necesite, también se denominan Jobs [2]. El procesamiento por batch es para aquellos programas de uso frecuente que se pueden ejecutar con una interacción humana mínima [6].

Por ejemplo, cada día muchas personas pagan la mensualidad de su tarjeta de crédito, cada una tiene un día determinado para hacer el pago, no se puede tener trabajadores mirando cada día a ver que clientes hay que hacerles el cargo de su tarjeta de crédito; lo que tendrá el banco, será un proceso batch que se ejecuta todos los días a cierta hora, comprueba los clientes que tienen que pagar el crédito ese día y les carga esa cantidad en su cuenta. A parte, esto desencadenará otros procesos batch diferentes, por ejemplo, otro job mandará un correo a cada cliente con el recibo del cargo, otro recalculará las cuotas del crédito que les quedan por pagar, otro recogerá los intereses pagados para enviárselo a Hacienda, etc. El término batch job [6] se originó en los días en que las tarjetas perforadas contenían las instrucciones que debía seguir una computadora cuando ejecutaba uno o más programas. Múltiples mazos de tarjetas que representan múltiples jobs a menudo se apilan uno encima de otro en la tolva de un lector de tarjetas y se ejecutan en lotes.

Después de que el usuario envía el job al sistema, la situación ideal es que no haya más interacción humana en el proceso hasta que se completa.

Los batch jobs se programan usando JCL (Job Control Language) en español Lenguaje de control de trabajo, que es el lenguaje de programación clave de z/OS que es el sistema operativo de los mainframe de IBM, estos últimos juegan un papel central en las operaciones diarias de la mayoría de las corporaciones más grandes del mundo, se llevan usando décadas en la industria probablemente por ser la

opción más eficiente y menos costosa de procesar datos, además de que muchas empresas simplemente prefieren no migrar a otras tecnologías por los costos que implicaría. [2]

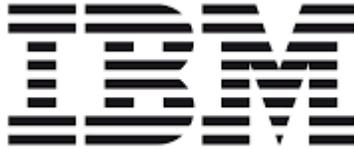


Figura 1.2 Logo de IBM.

Un job define los recursos requeridos por los procesos. Se usa para invocar programas o rutinas que se van a ejecutar, acceder a los ficheros que se van a utilizar, dar prioridades, entre otras cosas.

Los programas que se llaman desde dentro de un JCL, pueden estar hechos con diferentes lenguajes de programación, los más comunes son COBOL, Java y REXX, pero también encontramos C, PL/1 y Ensamblador. z/OS tiene compiladores para poder entender estos programas y ejecutarlos.

En este trabajo se expondrá de manera breve y concreta la trayectoria profesional en el ambiente de tecnologías de la información dentro del área de soporte técnico de los procesos ejecutados en el mainframe de un banco, se muestran ejemplos de situaciones que se presentan habitualmente, así como conceptos necesarios para entender las actividades realizadas.

1.2 OBJETIVO GENERAL

Debido a que 92 de los 100 principales bancos del mundo usan la tecnología mainframe [1] y prácticamente el 100% de las reservas de vuelos y operaciones con tarjetas de débito o crédito son procesadas por algún mainframe debido a la rapidez, seguridad y fiabilidad. También se estima que el 80% de la información corporativa/empresarial (que es lo que mueve nuestra economía), reside en máquinas mainframe, se sabe que estas computadoras de alto rendimiento trabajan

24/7 por décadas ejecutando batch jobs, por lo que siempre se debe de monitorear que estos se lleven a cabo correctamente.

En ese sentido se quiere asegurar que las cargas de los batch jobs que ejecuta el mainframe se completen, sin errores ni demoras. En caso de que se presenten fallos o retrasos en los procesos, se debe analizar, resolver y documentar cualquier problema que se presente, en caso de no contar con las herramientas requeridas para resolverlo o si simplemente no hay información suficiente sobre el error, se debe escalar la situación con un desarrollador.

1.3 OBJETIVOS ESPECÍFICOS

- **Monitorear:** Verificar la correcta ejecución de los batch jobs mediante la identificación de fallos y retrasos a través de software especializado.
- **Analizar:** En caso de alguna falla como el que un proceso no pueda completar o arrancar, debo analizar el registro de errores, encontrar el motivo por el cual el job no pudo completar/arrancar.
- **Solucionar:** Después de analizar se requiere encontrar la solución para que el job pueda completar satisfactoriamente, Informar / Reportar: en caso de no solucionar el error después del análisis y no encontrar documentación suficiente para proceder con la reparación del proceso, se debe contactar al desarrollador (nivel 3) para juntos encontrar la solución al problema y evitar que se vuelva a repetir.
- **Documentar:** Al confirmar que el proceso completó satisfactoriamente, se debe documentar todo trabajo hecho al job en un software especializado con la creación de un incidente, esto para llevar control y un historial de las veces que un proceso falla.
- **Administrar archivos:** Parte del trabajo consiste en editar, renombrar, copiar, eliminar y/o cambiar de directorio data sets para su disponibilidad para los jobs.

- **Coordinar equipos:** Para la resolución de una falla en la que se necesitan recursos adicionales para su arreglo, a veces se requiere comunicarse con otros equipos para y trabajar juntos en encontrar la mejor solución. Para la comunicación interna, se utilizan diferentes aplicaciones de las cuales se destacan Outlook [23], Zoom [19] y Slack [21] por ser las más utilizadas.

1.4 ALCANCES DEL PROBLEMA Y DELIMITACIONES

Este trabajo abarca las fases de mantenimiento de los procesos ejecutados por un mainframe, usando como objeto de estudio situaciones imaginarias que describirán y mostrarán las principales labores desempeñadas en el puesto de trabajo como ingeniero en sistemas en la empresa TCS, ya que por motivos de seguridad no es posible usar información extraída de escenarios reales, se realizará una descripción de los procesos, se identificará la información necesaria para analizarlos y elaborar una propuesta para mejorarlos.

1.5 JUSTIFICACIÓN

Los jobs que se ejecutan son críticos para el correcto funcionamiento del servicio de una empresa de servicios financieros, inversiones y aseguramiento con más de 13 millones de miembros, el correcto flujo de todos los procesos es fundamental para que los usuarios tengan acceso a sus bienes y servicios, permitiendo a la empresa ofrecer la disponibilidad de sus servicios todos los días del año. Debe asegurarse el correcto procesamiento de los jobs, para ello se apoya en un software de monitoreo llamado Control-M que permite saber cuándo un proceso no se completó satisfactoriamente o si no se ha ejecutado, las alertas que se reciben están programadas para aparecer cuando un job falla o si después de determinada hora y/o día no ha corrido, lo que permite analizar la situación para saber por qué el proceso no terminó o por qué se está retrasando. Dependiendo del caso, es necesario encontrar la mejor solución para que se complete correctamente o hacerlo correr para no retrasar cualquier otro proceso que espere por él.

La mayoría de jobs cuentan con condiciones que los hacen correr al cumplirse, si un job B es sucesor de un job A, su condición será que el job A complete satisfactoriamente, de lo contrario el job B no se ejecutara, por lo que el análisis no se limita a solo los jobs que disparan las alertas, sino también a los que les precede y los que prosiguen para descartar posibles errores que pudieron presentarse en procesos anteriores y evitar retrasos en procesos que están por ejecutarse en espera de los que no han ejecutado correctamente o a su debido tiempo. Los jobs que no tienen condiciones se ejecutan manualmente y por lo regular, son el primero de un conjunto de procesos que deben de correr después de haber revisado manualmente ciertos criterios que, por motivos del cliente, no se confían a la automatización.

2 CAPITULO SEGUNDO. DESCRIPCIÓN DEL CONTEXTO

En la modalidad presencial, TCS brinda el mobiliario necesario para llevar a cabo las actividades laborales, tales como escritorios, sillas, botes de basura, mesas de trabajo, sillones, entre otros, todo pensado para optimizar el espacio sin dejar mucho de lado la ergonomía.

Es casi imposible pensar que el trabajo en una empresa de TI pueda llevarse a cabo sin equipos de cómputo, software e internet, ya que son los recursos más valiosos para la producción y comunicación, por lo que la empresa provee con todo lo necesario para llevar a cabo las labores.

Las instalaciones cuentan con los servicios básicos como el agua, la luz y el antes mencionado servicio de internet.

2.1 DESCRIPCIÓN DE LA NORMATIVIDAD O REGLAS DE OPERACIÓN DE LA EMPRESA

En la modalidad normal / presencial el reglamento interno de TCS, como en cualquier otra empresa con rubro en las TI incluye todas disposiciones relacionadas con el uso del inmobiliario, los horarios, medidas de seguridad, entre otras, destacando fuertemente las reglas de confidencialidad, ya que se tiene acceso a herramientas que manejan datos sensibles que deben de protegerse de manera adecuada.

En la modalidad virtual por motivos de pandemia se firmó un contrato de confidencialidad para tratar la información fuera de la empresa, entre las nuevas medidas de seguridad destaca el no conectarse a redes de wi-fi públicas, no usar la laptop en lugares públicos o donde se encuentre gente desconocida así como tampoco hacer o recibir llamadas de trabajo, en pocas palabras lo que nos piden es no trabajar en Starbucks o lugares muy concurridos donde pueda estar en peligro

la información que manejamos, lo que personalmente no representa ningún problema ya que es mucho más sencillo trabajar desde casa.

2.2 ENTORNO DONDE SE UBICA LA EMPRESA

TCS es una empresa comprometida con el desarrollo y la diversidad, no solo por tendencia, sino también por expansión, en Guadalajara trabajan principalmente mexicanos e indios, pero se puede encontrar gente que viene de todo el continente americano, la empresa introduce empleados de otras nacionalidades para que exista una mayor diversidad cultural, y a su vez, un mejor manejo de los recursos ya que dependiendo del país, se generan menos costos con la reubicación de empleados con experiencia para sustituir bajas, que contratar a nuevo personal, por ejemplo, si un empleado que ya trabaja para TCS en México es experto en un área de la cual hay un puesto disponible en USA, en la mayoría de los casos, es más rentable reubicar al empleado de México que contratar a alguien con experiencia en USA.

A pesar de que el estado de Jalisco se encuentra en semáforo naranja debido a la pandemia, TCS no promueve la vacunación entre los colaboradores, probablemente por ser una empresa tan diversa en culturas, costumbres y religiones, se planea regresar a las oficinas antes de septiembre del 2022, por lo que muchos empleados se han pronunciado en contra de volver a las oficinas, ya que las instalaciones se encuentran bastante retiradas de la ciudad y el tráfico, se convierte en un problema que puede prolongar las horas necesarias para el trabajo ya que en un día normal, se puede hacer más de una hora para llegar a las oficinas y más de una hora para regresar a casa. TCS ha mencionado que está analizando la posibilidad de crear un sistema híbrido en el que no se tenga que ir a la oficina todos los días, pero al parecer será implementado hasta 2023.

3 CAPITULO TERCERO. FUNDAMENTO TEÓRICO DE LAS HERRAMIENTAS Y CONOCIMIENTOS APLICADOS

Para desarrollar bien este trabajo, es necesario conocer ciertos conceptos, a continuación, se mencionan los siguientes:

¿Qué es un mainframe?

En esencia, los mainframes son computadoras de alto rendimiento con grandes cantidades de memoria y procesadores que procesan miles de millones de cálculos y transacciones simples en tiempo real [1]. El mainframe es fundamental para las bases de datos comerciales, los servidores de transacciones y las aplicaciones que requieren alta resistencia, seguridad y agilidad. z/OS es el sistema operativo de los mainframes de IBM [4].

Estos mainframes ofrecen un altísimo nivel de seguridad debido a las configuraciones y hardware de criptografía que usa. Está pensado para trabajar en todo tipo de condiciones y garantizar su servicio.

Cuando empezaron a usarse los primeros mainframes no había ordenadores como los tenemos hoy en día. Por lo que la forma de trabajar con ellos era también bastante diferente.

Los códigos y configuraciones del sistema se almacenaban en tarjetas perforadas, que tenían que ser introducidas y leídas por la máquina para coger la información, y después de ser tratada, otras tarjetas perforadas podían ser grabadas con la información resultante.

Hoy en día, la forma de interactuar con la máquina es con el código y comandos que se introducen a través del terminal del ordenador. Estos códigos y la información serán leídas y grabadas en diferentes tipos de ficheros.

Físicamente un mainframe es como una nevera de grande, antiguamente podían ocupar una habitación entera. Hoy en día, ocupa un 75% de espacio menos que si se usa los equiparables servidores en rack, para conseguir el mismo rendimiento, además utilizan un 40% menos de energía.



Figura 3.1 IBM z15 mainframe.

¿Qué es COBOL?

COBOL (COmmon Business-Oriented Language) es el lenguaje de programación orientado a negocios que mueve prácticamente la economía mundial al ser utilizado en bancos e instituciones financieras[7]. Pese a que muchas personas creen que el este lenguaje está en desuso, la realidad es que casi todos los sistemas que requieren gran capacidad de procesamiento por lotes (batch), tanto las entidades bancarias como otras grandes empresas con sistemas mainframes utilizan COBOL. Esto permite garantizar la compatibilidad de los sistemas antiguos con los más modernos, así como tener la seguridad de que el lenguaje es perfectamente estable y probado. Los batch jobs (entre otras cosas) son los encargados de hacer los llamados a los programas escritos en COBOL.

¿De qué se compone un batch job?

JCL (acrónimo de Job Control Language), en español Lenguaje de Control de Trabajos, es un lenguaje informático en el que se escriben los jobs dándole instrucciones de ejecución al mainframe mediante los llamados “pasos” o “sentencias” que son declaraciones u órdenes con las que se indica al sistema operativo qué tareas debe realizar y que programa ejecutar, en qué secuencia han de hacerlo y en qué periféricos están ubicados los ficheros de datos (de entrada y/o de salida) que requieren dichas tareas.

En el contexto de JCL, un job es un conjunto coherente de instrucciones para realizar un proceso particular [28] en la mayoría de los casos ejecutando un programa COBOL. En pocas palabras, todos los programas COBOL de la empresa usan JCL para su ejecución a través de la sentencia EXEC.

En la ilustración 4 podemos ver un job escrito en JCL, que tiene como función, borrar el data set MTHUSR.TEST.PSFILE.

```
EDIT          MTHUSR.IEFBR14.JCLLIB (DELETE) - 01.00          Columns 00001 00072
Command ==>                                         Scroll ==> CSR
=COLS>  ---+---1---+---2---+---3---+---4---+---5---+---6---+---7---
***** Top of Data *****
000001 //MTHUSR JOB (123), 'MTHUSR', CLASS=A, MSGCLASS=A, MSGLEVEL=(1,1),
000002 //          NOTIFY=&SYSUID
000003 //*****
000004 //* DELETE A DATA SET USING IEFBR14 UTILITY
000005 //*****
000006 //STEP20 EXEC PGM=IEFBR14
000007 //SYSPRINT DD SYSOUT=*
000008 //SYSOUT DD SYSOUT=*
000009 //SYSDUMP DD SYSOUT=*
000010 //DD1 DD DSN=MTHUSR.TEST.PSFILE,
000011 //          DISP=(OLD,DELETE,DELETE)
000012 //DD2 DD DSN=MTHUSR.TEST.PDSFILE,
000013 //          DISP=(OLD,DELETE,DELETE)
000014 //*
***** Bottom of Data *****
```

Figura 3.2 JCL.

¿Qué es un data set?

El término data set se refiere a un archivo que contiene uno o más registros [3]. El registro es la unidad básica de información utilizada por un programa que se ejecuta en el sistema operativo del mainframe, estos pueden contener información, como registros médicos, registros de seguros, información personal, etc. Se puede comparar con una hoja de Word, una hoja de Excel o con una base de datos dependiendo de la función para que será utilizada por el programa ejecutado en el sistema. Los data sets también se utilizan para almacenar información que necesitan las aplicaciones o el propio z/OS.

3.1 HERRAMIENTAS UTILIZADAS EN LA SOLUCIÓN

Para realizar el trabajo se hace uso de diferentes tipos de software, incluyendo entre los principales, pero no limitado a:

3.1.1 Microsoft Windows 10

Microsoft Windows 10: Es un sistema operativo desarrollado por Microsoft [18] como parte de la familia de sistemas operativos Windows NT, integra las mejores características de seguridad, protección de la identidad e información [17]. También es compatible con todas las herramientas necesarias para un desempeño óptimo como las máquinas virtuales de z/OS que son indispensables en el área donde se realizan las labores.



Figura 3.3 Logo de Windows 10.

3.1.2 z/OS

z/OS: Es el sistema operativo de los mainframe de IBM [14], está diseñado para ofrecer un entorno estable, seguro y continuamente disponible para las aplicaciones que se ejecutan en el mainframe, está enfocado (pero no limitado) para trabajar con JCL y COBOL que han estado presentes en la industria en 6 diferentes décadas[4],

cuenta con las herramientas necesarias para la administración de archivos (data sets []), nuestro cliente siempre ha trabajado con estas herramientas desde sus inicios y hasta el momento, les sigue funcionando sin mayores problemas.

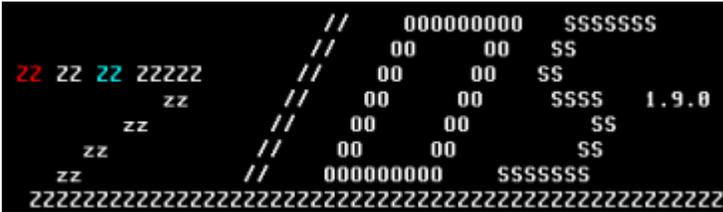


Figura 3.4 Logo de z/OS.

3.1.3 Control-M

Control-M: Es un software de automatización de cargas de trabajo que ofrece funcionalidades para crear, definir, programar, administrar y monitorear flujos de trabajo de producción (batch jobs)[16], este software es el que el cliente especifica que debe de usarse para el monitoreo de los Jobs.



Figura 3.5 Logo de Control-M.

3.1.4 Servicenow (SNOW)

Servicenow (SNOW): Es una herramienta que procesa y cataloga los fallos de los procesos, creando automáticamente un incidente de los que fallan en donde se debe de documentar el trabajo realizado para arreglar el fallo, el cliente especifica que este es el software que debe usarse para la documentación de los Jobs [15] .



Figura 3.6 Logo de Servicenow.

3.1.5 Zoom

Zoom: Es una de las principales aplicaciones de software de videoconferencia. Permite interactuar virtualmente con compañeros de trabajo en reuniones virtuales[19], previamente Skype [20], era la principal herramienta para la comunicación entre los compañeros de trabajo, después se migró a Zoom probablemente para reducir costos.



Figura 3.7 Logo de Zoom.

3.1.6 Slack

Slack: Es una aplicación de mensajería para empresas, está diseñado para equipos y lugares de trabajo que se pueden usar en múltiples dispositivos y plataformas, cuenta con funciones sólidas que le permiten no solo chatear uno a uno con asociados, sino también en grupos, además de permitir cargar y compartir archivos[21], probablemente el cliente pida el uso de esta aplicación por el motivo de que en IBM también la usan y facilita la comunicación entre las empresas y sus empleados.



Figura 3.8. Logo de Slack.

3.1.7 Suite Microsoft Office

Suite Microsoft Office: Es una suite ofimática de aplicaciones para computadoras de escritorio, servidores y servicio, la aplicación que más utilizamos de esta suite sin duda es Outlook [22, 23], que entre otras cosas incluye un cliente de correo electrónico y un calendario, indispensables para la comunicación entre colegas dentro de la empresa, probablemente se utilice esta Suite por ser tan completa, en nuestra área usamos muy poco o nada, las demás aplicaciones incluidas pero sin duda, en otros entornos, la totalidad de la suite es indispensable, por lo que su uso es un requisito del cliente.



Figura 3.9. Logo de la Suite Microsoft Office.

4 CAPITULO CUARTO. PROPUESTAS DE SOLUCIÓN AL PROBLEMA O PROBLEMAS PRESENTADOS

Una de las fases más importantes del ciclo de vida de desarrollo del software es el mantenimiento, para lo que hay que eliminar los defectos detectados durante su vida útil (mantenimiento correctivo).

Cada vez que se agregan o retiran procesos, se corre el riesgo de que algo no funcione correctamente, por lo que el monitoreo automatizado para la detección de errores es muy importante, desafortunadamente, no se puede crear un software que corrija los errores de un job que ha fallado, por lo que la intervención humana es necesaria, el documentar los errores y soluciones es de gran importancia para que en caso de que se vuelva a presentar un error, poder corregirlo lo más fácil y rápido posible.

Al realizar el monitoreo de jobs se presentan principalmente 2 casos:

1. En donde se recibe una alerta de que un job falló, por lo que necesita ser revisado para encontrar la razón y solución de su fallo, a este tipo de alertas se les conoce como cancelaciones.
2. Cuando se recibe una alerta de un job que no se ha ejecutado, por lo que se necesita revisar que es lo que lo detiene, a este tipo de alertas se les conoce como retrasos.

A continuación, se presentan 3 ejemplos de cancelaciones y 3 ejemplos de retrasos, su revisión, solución y documentación.

4.1 EJEMPLO DE CANCELACIÓN #1:

Se supone que la empresa “ABC” recibe de la empresa “XYZ” un data set con información muy importante cada semana con el nombre “XYZ.DATOS.IMPORANTES.V10”, este archivo contiene información encriptada

que es muy importante para la empresa, al ser recibido es respaldado por el job “JOBABC01” que su código JCL se presenta a continuación:

```
//JOBABC01 JOB ( ), ' ' CLASS=A, MSGCLASS=Q, MSGLEVEL=(1,1),
// REGION=0M,NOTIFY=&SYSUID
//*****
*****
//* Crear respaldo de data set de XYZ *
//*****
*****
//STEP1 EXEC PGM=IEBGENER
//SYSPRINT DD SYSOUT=*
//SYSUT1 DD DSN=XYZ.DATOS.IMPORANTES.V10,DISP=SHR
//SYSUT2 DD DSN=XYZ.DATOS.RESPALDO.G(+1),
//          DISP=(NEW,CATLG,DELETE),
//          UNIT=3390,
//          SPACE=(TRK,(10,5),RLSE),
//          DCB=(RECFM=FB, LRECL=50,BLKSIZE=0)
//SYSIN DD DUMMY
```

Código 1. JOBABC01 - ejemplo de cancelación #1.

Por alguna razón, el job “JOBABC01” falla, por lo cual se recibe una alerta en Control-M como se muestra en la ilustración 4.1.

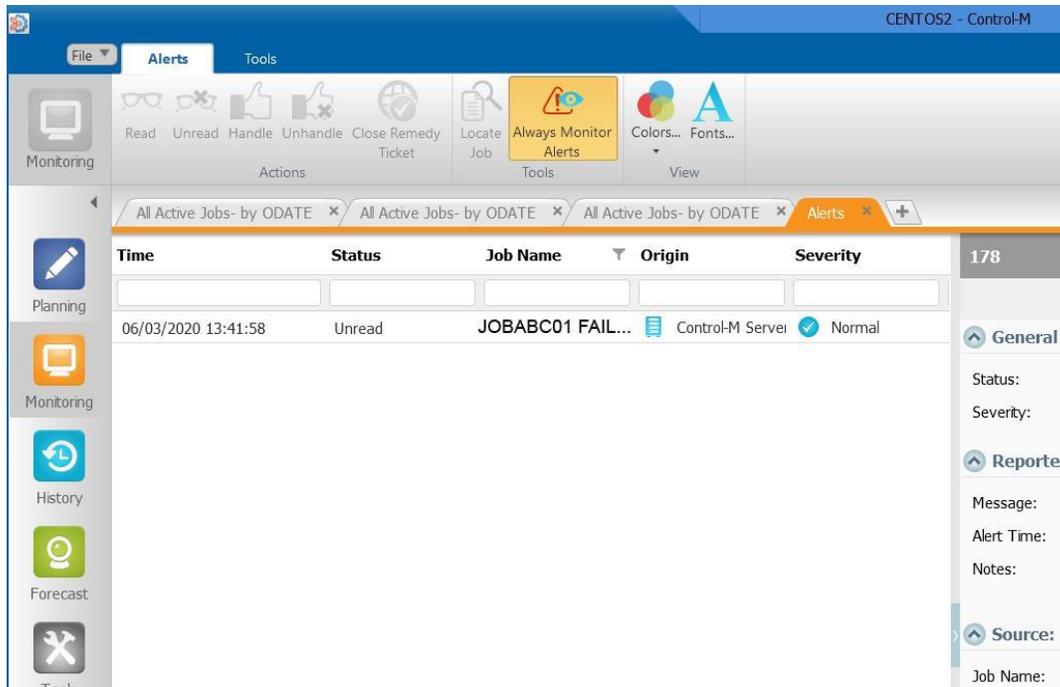


Figura 4.1. Alerta de fallo en Control M – Ejemplo de cancelación #1.

En este caso, al ser la primera vez que falla, probablemente no existirá documentación al respecto, por lo cual, se debe de investigar que provocó el error, arreglarlo y documentarlo, para esto, debemos de entrar a z/OS y ver el registro de errores del job. Se vera algo parecido a lo mostrado en la ilustración 4.2.

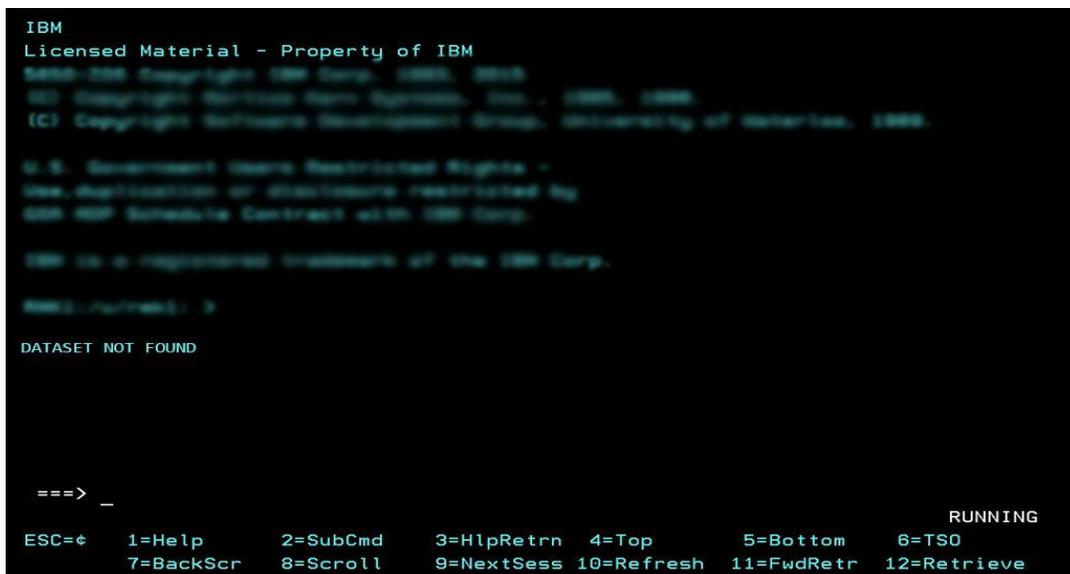


Figura 4.2. Registro de errores z/OS – Ejemplo de cancelación #1.

Como se puede ver, el motivo por el cual el job falla, es que no encontró el data set que necesita para trabajar, al encontrar un error, el proceso se detiene, evitando que cualquier otro job sucesor se ejecute.

De momento se sabe la razón por la cual “JOBABC01” falló, pero el registro no muestra el nombre del archivo que el JCL necesita para una ejecución exitosa, por lo que a través de z/OS se entra a la librería desde donde corren los jobs y se busca JOBABC01 para ver su código, lo que sería mostrado, es lo presentado en el código 1. JOBABC01 - ejemplo de cancelación #1.

La línea de código `SYSUT1 DD DSN=XYZ.DATOS.IMPORANTES.V10` es donde el job toma el data set para copiarlo con el nombre que se muestra en la línea `SYSUT2 DD DSN=XYZ.DATOS.RESPALDO.G(+1)` (el +1 indica una nueva generación con ese nombre)

Con esto se sabe que el job “JOBABC01” trabaja con el data set “XYZ.DATOS.IMPORANTES.V10” que no fue encontrado, en este caso, lo mejor es buscar cual fue el último archivo enviado por “XYZ” a “ABC”, esto puede realizarse a través de z/OS con la aplicación Data Set Utility [5], lo mejor sería buscar XYZ.DATOS* para mostrar todos los archivos que empiecen así su nombre como se muestra en la ilustración 4.3.

En este ejemplo, estamos esperando por el data set “XYZ.DATOS.IMPORANTES.V10” pero tenemos uno llamado “XYZ.DATOS.IMPORANTES.V00”, si se confirma que el archivo ha sido creado hoy, que es cuando “JOBABC01” debe ejecutarse, se puede asumir que XYZ mandó el data set con un nombre que el job no esperaba, por lo que se debe de contactar al proveedor del archivo para confirmar que el archivo es el correcto, pero que solo debe de renombrarse, para efectos prácticos de este ejemplo, se va a suponer que se manda un correo electrónico a XYZ preguntando si el data set es correcto y se obtiene respuesta que todo está bien dentro del archivo, pero que por un error, se mandó con un nombre diferente y se dan instrucciones de proceder, en este caso, solo sería necesario renombrar el data set a través de z/OS de “XYZ.DATOS.IMPORANTES.V10” a “XYZ.DATOS.IMPORANTES.V00” y reiniciar “JOBABC01” para que vuelva a ejecutarse, pero esta vez buscara un data set que sí existe, si no se encuentran errores adicionales, el job completara satisfactoriamente y deberá documentarse el proceso que se llevó a cabo para arreglarlo en la plataforma SNOW [15] como se puede ver en la ilustración 4.5.

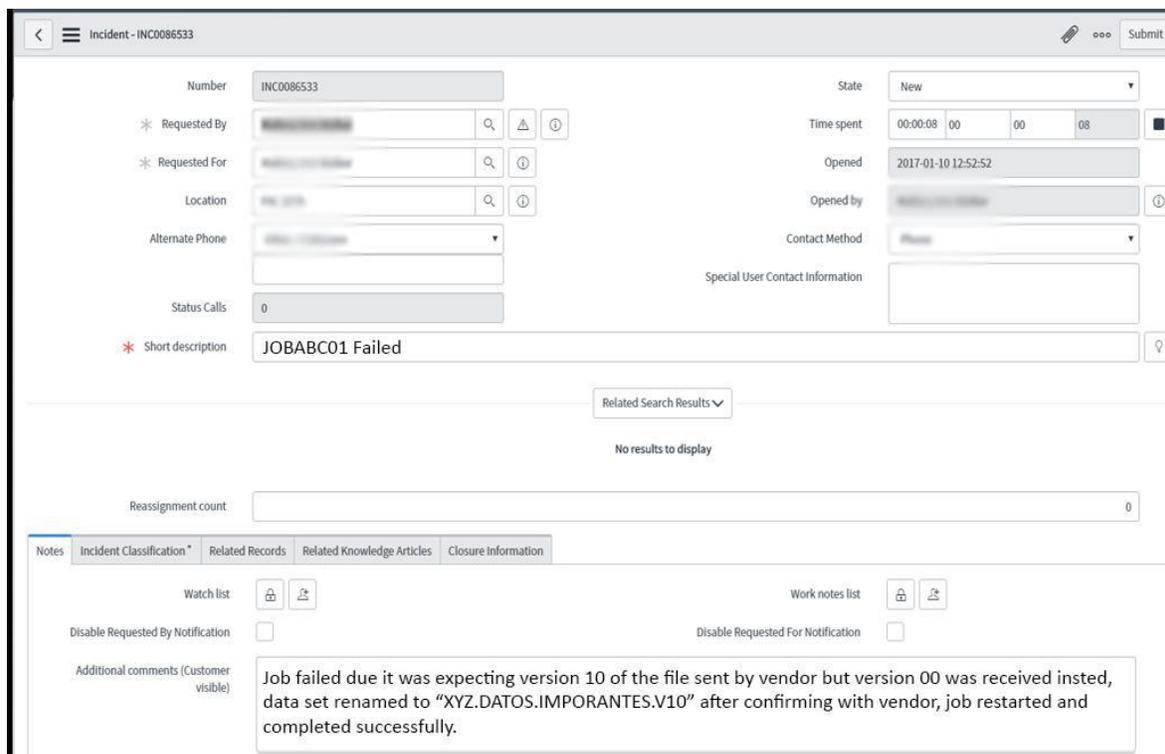


Figura 4.5. Documentación en SNOW - Ejemplo de cancelación #1.

Traducción: El job falló debido a que esperaba la versión 10 del archivo enviado por el proveedor, pero se recibió la versión 00, el data set se renombró a "XYZ.DATOS.IMPORANTES.V10" después de confirmar con el proveedor, el trabajo se reinició y completó con éxito.

Es muy importante que toda la comunicación y documentación se lleve a cabo en inglés, ya que es el idioma común entre los miembros de la empresa y proveedores.

4.2 EJEMPLO DE CANCELACIÓN #2:

En este caso, se verá el ejemplo de una cancelación en la que es necesario la administración de archivos, que es también una tarea fundamental de las labores. Se supone que la empresa "ABC" recibe de la empresa "XYZ" un data set cada día, con fecha de cuando el archivo es creado por lo que cada día, por lo menos la fecha debe de ser diferente, para esto, vamos a usar un job al que llamaremos JCOMPARA que haga una comparación de los archivos XYZ.DATA.SET.V*, si en los dos data sets, el día actual y el anterior son iguales el job fallara, si son diferentes completara satisfactoriamente.

El código se presenta a continuación:

```
//JCOMPARA JOB (),' ',CLASS=A,MSGCLASS=Q,MSGLEVEL=(1,1),
// REGION=0M,NOTIFY=&SYSUID,RESTART=STEP1
//*****
//* Comparar dos data sets *
//*****
//STEP1 EXEC PGM=IEFBR14
//SYSPRINT DD SYSOUT=*
//SYSUT1 CURRFILE=XYZ.DATA.SET.V(0),
//SYSUT2 PREVFILE=XYZ.DATA.SET.V(-1),
//SYSIN DD *
    COMPARE TYPORG=PS
/*
//
```

Código 2. JCOMPARA - ejemplo de cancelación #2

Suponiendo que el contenido del data set XYZ.DATA.SET.V01 es el siguiente:

XYZCLIENTESNUEVOS19/04/202221:00:00
JUAN PEREZ 123456789 987654321 HMOSON

Y el contenido del data set XYZ.DATA.SET.V02 es:

XYZCLIENTESNUEVOS20/04/202221:00:00

Se tiene en ambos data sets un encabezado que da a entender que este archivo contiene la información de clientes nuevos de XYZ, en V01 hay 2 renglones, uno con la fecha y hora de creación del archivo y el segundo renglón con lo que parece ser la información de un cliente nuevo.

En V02 se ve que solo tiene un renglón con encabezado, por lo que se puede asumir que no hay clientes nuevos para este día.

El job hará la comparación de los data sets y encontrará diferencias en el número de renglones y en las fechas, por lo que terminará satisfactoriamente.

Al día siguiente 21/04/2022 XYZ tampoco tiene clientes nuevos, por lo que en XYZ deciden hacer una copia del data set [3] XYZ.DATA.SET.V02 y enviarla como XYZ.DATA.SET.V03, al ser una copia contiene exactamente la misma información, el job[6] fallará al hacer la comparación y no encontrar diferencias.

Una vez que el falle, se recibirá una alerta a través de Control-M como se muestra en la ilustración 4.6.

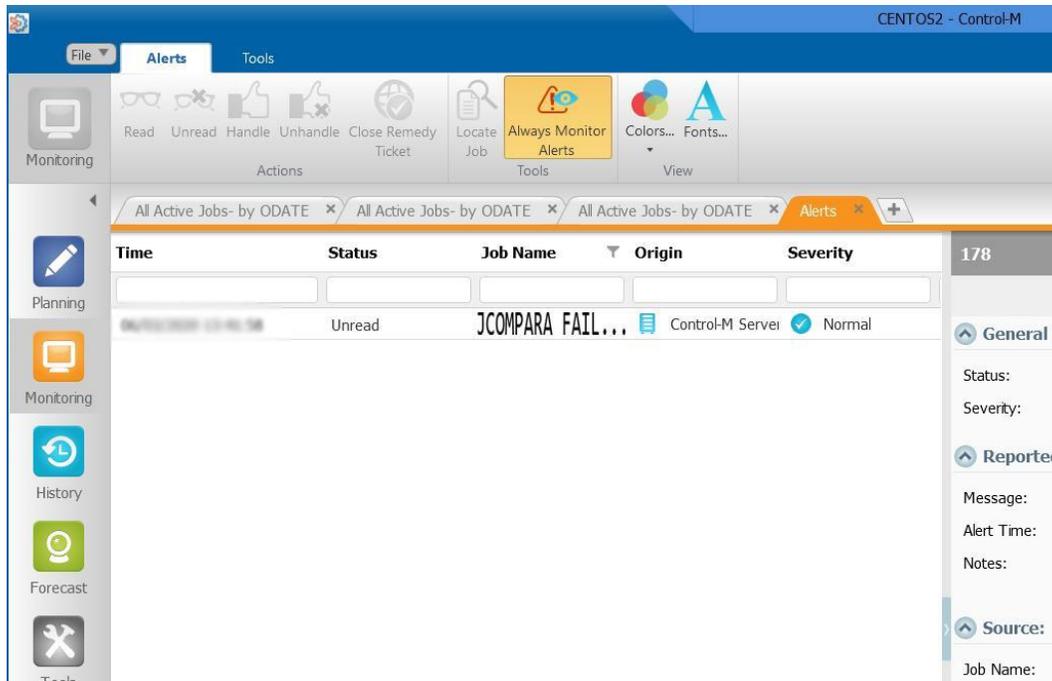


Figura 4.6. Alerta de fallo en Control M – Ejemplo de cancelación #2.

En z/OS podremos ver el registro de error es y ver un mensaje que dice “DUPLICATE FILE” como se muestra en la ilustración 4.7.

```

IBM
Licensed Material - Property of IBM
Copyright International Business Machines Corporation. 1989, 1990, 1991, 1992, 1993, 1994, 1995, 1996, 1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005, 2006, 2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022, 2023, 2024, 2025.
All rights reserved. Copying, distributing, or otherwise using this information without the prior written permission of International Business Machines Corporation is prohibited.
IBM is a registered trademark of the IBM Corp.

=====
DUPLICATE FILE
UNEXPECTED DATE IN HEADER

====> _
RUNNING
ESC=␣ 1=Help 2=SubCmd 3=HlpRetrn 4=Top 5=Bottom 6=TSO
7=BackScr 8=Scroll 9=NextSess 10=Refresh 11=FwdRetr 12=Retrieve

```

Figura 4.7. Registro de errores z/OS – Ejemplo de cancelación #2.

Se debe hacer una comparación manual de los últimos 2 data sets recibidos para confirmar que sean iguales, para eso se puede utilizar una herramienta de z/OS llamada SuperC que nos permite comparar 2 data sets, la aplicación se muestra en la ilustración 4.8.

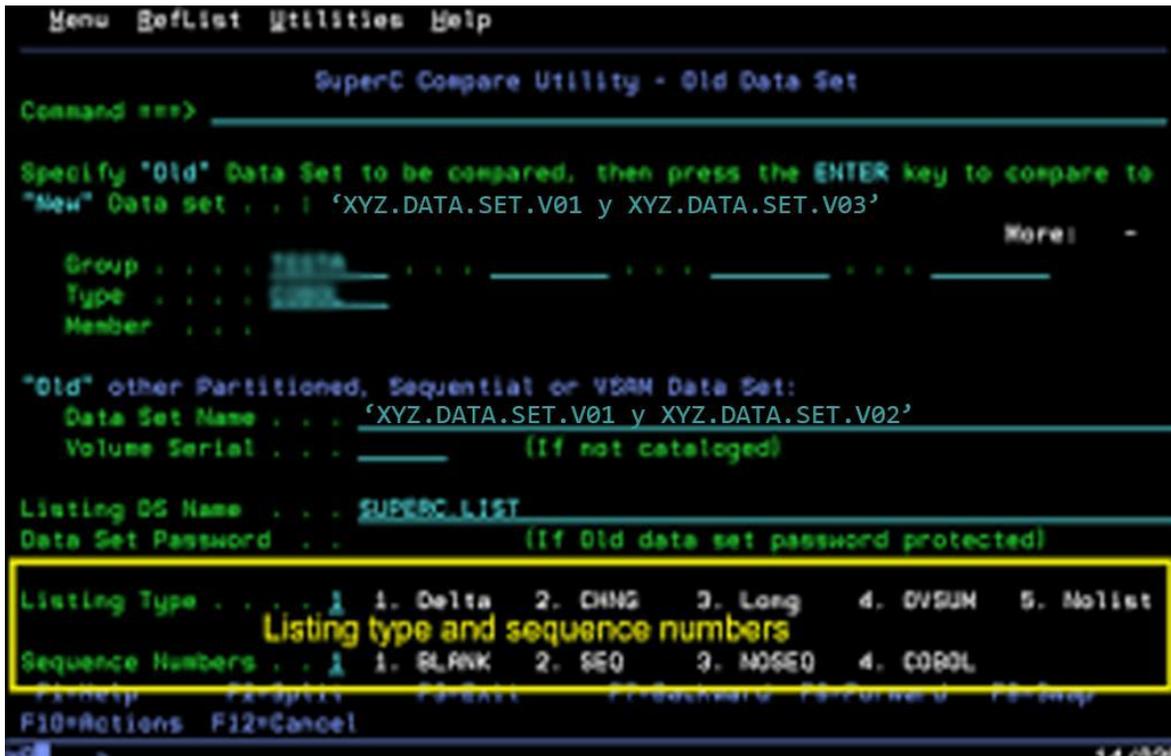


Figura 4.8. SuperC - Ejemplo de cancelación de cancelación #2.

SuperC pedirá los nombres de los archivos a comparar refiriéndose a ellos como "NEW" y "OLD" el último archivo recibido se insertará en NEW y el anterior en OLD lo que dará como resultado una pantalla como se muestra en la ilustración 4.9.

```
Command ===>
000014
000015      1 NUMBER OF LINE MATCHES
000016      0 REFORMATTED LINES
000017      0 NEW FILE LINE INSERTIONS
000018      0 OLD FILE LINE DELETIONS
000019      1 NEW FILE LINES PROCESSED
000020      1 OLD FILE LINES PROCESSED
000021
000022 LISTING-TYPE = DELTA      COMPARE-COLUMNS
000023 PROCESS OPTIONS USED: SEQ(DEFAULT)
```

Figura 4.9. SuperC resultados – Ejemplo de cancelación #2.

Como se puede ver, hay una línea procesada en el archivo nuevo (NEW) al igual que en el anterior (OLD), en NUMBER OF LINES MATCHES se muestra el número de líneas que coinciden entre los dos archivos, se puede ver que hay una, lo que quiere decir que la totalidad del contenido de los data sets coinciden al tener la misma cantidad de líneas procesadas y líneas que coinciden.

Después de hacer el análisis se le debe de comunicar al proveedor XYZ que el último archivo enviado contiene la misma información que la de ayer, incluyendo la fecha, si XYZ confirma que el data set es correcto, manualmente se editará el contenido en z/OS después de crear un respaldo, se cambiará la fecha del encabezado 20/04/202221:00:00 por 21/04/202221:00:00 y se reiniciará JCOMPARA que completará satisfactoriamente al no encontrarse con archivos duplicados, por lo que se puede proceder a documentar lo que pasó en SNOW como se muestra en la ilustración 4.10

Incident - INC0086533 Submit

Number	INC0086533	State	New
* Requested By	[Redacted]	Time spent	00:00:08 00 00 08
* Requested For	[Redacted]	Opened	2017-01-10 12:52:52
Location	[Redacted]	Opened by	[Redacted]
Alternate Phone	[Redacted]	Contact Method	Phone
Status Calls	0	Special User Contact Information	[Redacted]
* Short description	JCOMPARA Failed		

Related Search Results No results to display

Reassignment count:

Notes | Incident Classification* | Related Records | Related Knowledge Articles | Closure Information

Watch list | Work notes list

Disable Requested By Notification | Disable Requested For Notification

Additional comments (Customer visible)

Job failed due to a duplicate file, today's information is the same as yesterday, file only contained header with yesterday's date, manually edited header's file with today's date as per vendor, job restarted and completed successfully.

Figura 4.10. Documentación en SNOW - Ejemplo de cancelación #2.

Traducción: El job falló debido a un archivo duplicado, la información de hoy es la misma que la de ayer, el archivo solo contenía el encabezado con la fecha de ayer, se editó manualmente el encabezado con la fecha de hoy por instrucciones del proveedor, el job fue reiniciado y completado con éxito.

4.3 EJEMPLO DE CANCELACIÓN #3:

Ahora se presenta un job con el nombre JBORRADO que borra un archivo, su código es el siguiente:

```
//JBORRADO JOB
(123),'IBMMAINFRAME[1]R',CLASS=C,MSGCLASS=S,MSGLEVEL=(1,1),
//  NOTIFY=&SYSUID
//*****
//* ESTE JOB BORRA ARCHIVO.PRUEBA.BORRAR *
//*****
//STEP001 EXEC PGM=IEFBR14
//SYSPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//SYSDUMP DD SYSOUT=*
//DD01 DD DSN=ARCHIVO.PRUEBA.BORRAR,
// DISP=(OLD,DELETE, )
//*
```

Código 3. JBORRADO - ejemplo de cancelación #3

En este ejemplo este job trabaja con la utilidad IEFBR14 de z/OS que sirve para crear data sets vacíos o para borrarlos, en este ejemplo se puede identificar que borrara el archivo **ARCHIVO.PRUEBA.BORRAR** ya que el código contiene un comentario que nos ayuda a entender el funcionamiento del job, además de que contiene una línea que contiene **DISP=(OLD,DELETE,)** que le indica al JCL que un archivo existente (OLD) debe de ser borrado en caso de que el job termine bien, los 3 parámetros de **DISP** de izquierda a derecha indican: **OLD** es un archivo existente, **DELETE** borrar el archivo si el job termina sin errores y un espacio en blanco que indica que el archivo se conserva si el job tuviera algún error.

Se supondrá que alguien por alguna razón, tiene el data set **ARCHIVO.PRUEBA.BORRAR** abierto a la hora de ejecutar el job, al estar en uso, el proceso no puede llevarse a cabo ya que un archivo no puede ser usado por 2 procesos a la vez, lo que provocaría el fallo del job y el archivo que debía ser borrado quedará disponible en el sistema.

En este caso, se recibe una alerta en Control-M para hacer saber que el job falló como se muestra en la ilustración 4.11.

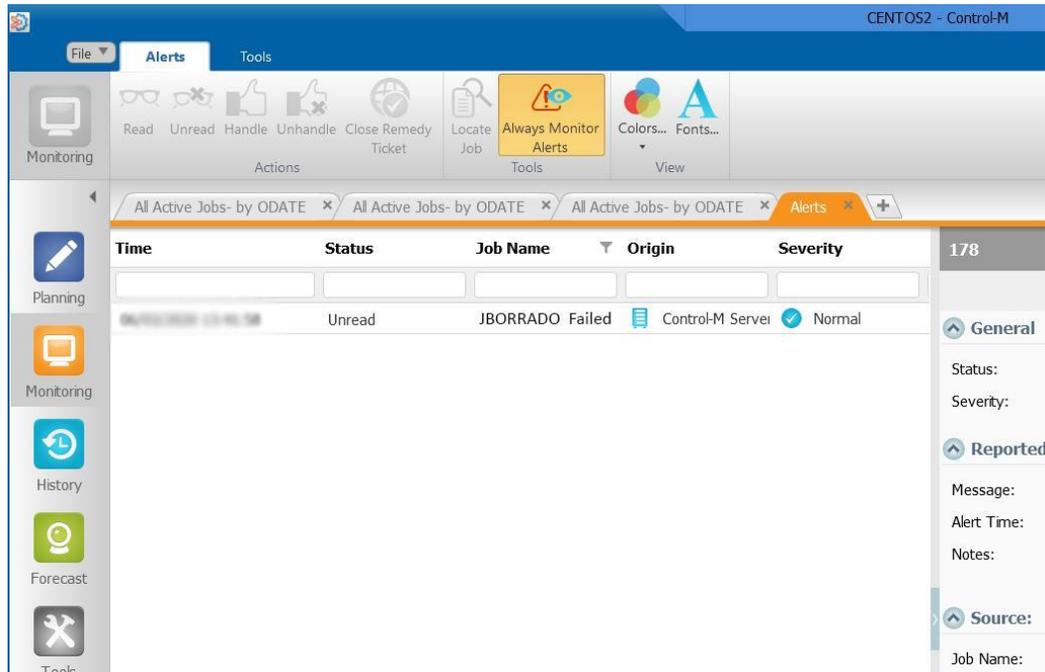


Figura 4.11. Control M – Ejemplo de cancelación #3.

nos encontraríamos en el registro de errores un mensaje como: “**DATA SET ALREADY IN USE**” algo parecido a lo que se muestra a en la ilustración 4.12.

```

IBM
Licensed Material - Property of IBM
5655-206 Copyright IBM Corp. 1989, 2003
    (C) Copyright Software Development Group, University of Waterloo, 1989.

U.S. Government Works Restricted Rights -
See Application or Distribution restricted by
DDP RFP Schedule Contract with IBM Corp.

IBM is a registered trademark of the IBM Corp.

=====
DATA SET ALREADY IN USE

====> _
                                                    RUNNING
ESC=␣  1=Help    2=SubCmd   3=HlpRetrn  4=Top      5=Bottom   6=TSO
        7=BackScr  8=Scroll   9=NextSess 10=Refresh 11=FwdRetr 12=Retrieve

```

Figura 4.12. Registro de errores z/OS - Ejemplo de cancelación #3.

Al analizar el JCL se puede ver que su trabajo es borrar el archivo por lo que puede ser borrado manualmente haciendo uso del administrador de archivos de z/OS, se insertara una 'D' de DELETE (borrar) a la izquierda del archivo como se muestra en la ilustración 4.13 y presionando enter el data set quedara eliminado.

```

  Menu  Options  View  Utilities  Compilers
-----
DSLIST - Data Sets Matching ARCHIVO.PRUEBA*
Command ==> _____

Command - Enter "/" to select action
-----
D      ARCHIVO.PRUEBA.BORRAR
      ARCHIVO.PRUEBA.NOBORRAR
      ARCHIVO.PRUEBA.ABC
      ARCHIVO.PRUEBA.123
***** End of Data Set

```

Figura 4.13. Administrador de archivos z/OS – Ejemplo de cancelación #3.

Una vez borrado el archivo se puede proceder a la terminación forzosa del job, esto hará que en Control-M aparezca como si hubiera terminado satisfactoriamente y así si hay jobs esperando por JBORRADO puedan ejecutarse, después de esto se puede proceder a documentar el fallo en SNOW como se muestra en el ejemplo de la ilustración 4.14.

Number: INC0086533

State: New

Requested By: [Redacted]

Requested For: [Redacted]

Location: [Redacted]

Alternate Phone: [Redacted]

Status Calls: 0

Time spent: 00:00:08

Opened: 2017-01-10 12:52:52

Opened by: [Redacted]

Contact Method: Phone

Special User Contact Information: [Redacted]

Short description: JBORRADO Failed

Reassignment count: 0

Notes: Incident Classification * | Related Records | Related Knowledge Articles | Closure Information

Additional comments (Customer visible): Job failed due data set was already in use when trying to delete it, file was deleted manually and job was forced completed.

Figura 4.14. Documentación en SNOW – Ejemplo de cancelación #3.

Traducción: El job falló debido a que el data set ya estaba en uso cuando se intentó eliminarlo, el archivo se eliminó manualmente y se forzó la terminación del job.

4.4 EJEMPLO DE RETRASO #1:

Cuando se recibe una alerta por el retraso de un job, por lo regular no es necesario revisar el JCL, toda la información que se necesita se puede encontrar en Control-M la mayoría de las veces. Para este ejemplo, se supone que existe un job llamado JSUCBORRA que es sucesor de JBORRADO visto en el ejemplo de cancelación #3, Para este ejemplo aún no se resuelve el problema del job cancelado, la alerta se recibe minutos después del fallo de JBORRADO ya que JSUCBORRA no puede ejecutarse si su predecesor se encuentra en estado fallido, la ventana de alertas en Control-M se ve como se muestra en la ilustración 4.15.

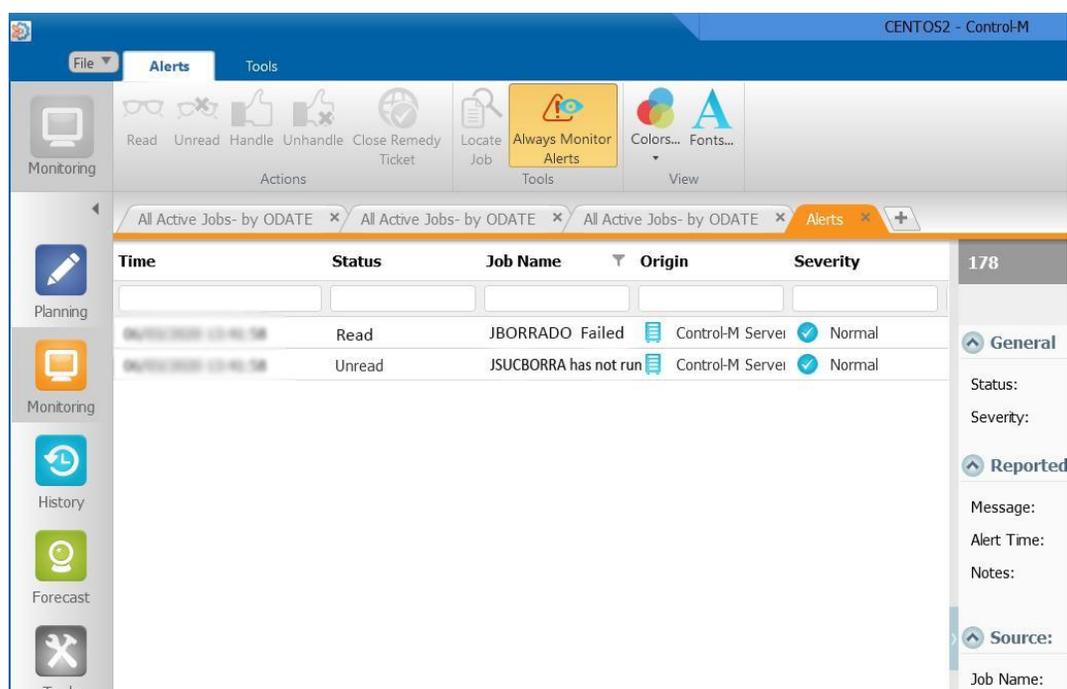


Figura 4.15. Control-M – Ejemplo de retraso #1.

Para analizar la situación del job, se puede ver el diagrama de flujo situándose en el proceso retrasado como se muestra la ilustración 4.16.



Figura 4.16. Diagrama de flujo en Control-M - Ejemplo de retraso #1.

Se puede apreciar que JSUCBORRA depende de que JBORRADO complete satisfactoriamente y no se ejecutará hasta que este último termine sin errores.

Para este tipo de alertas, se debe trabajar en el job que falló, por lo que una vez que se arregle JBORRADO como se muestra en el Ejemplo de cancelación #3 el job JSUCBORRA se ejecutará, completará y el diagrama de flujo quedará como se muestra en la ilustración 4.17.

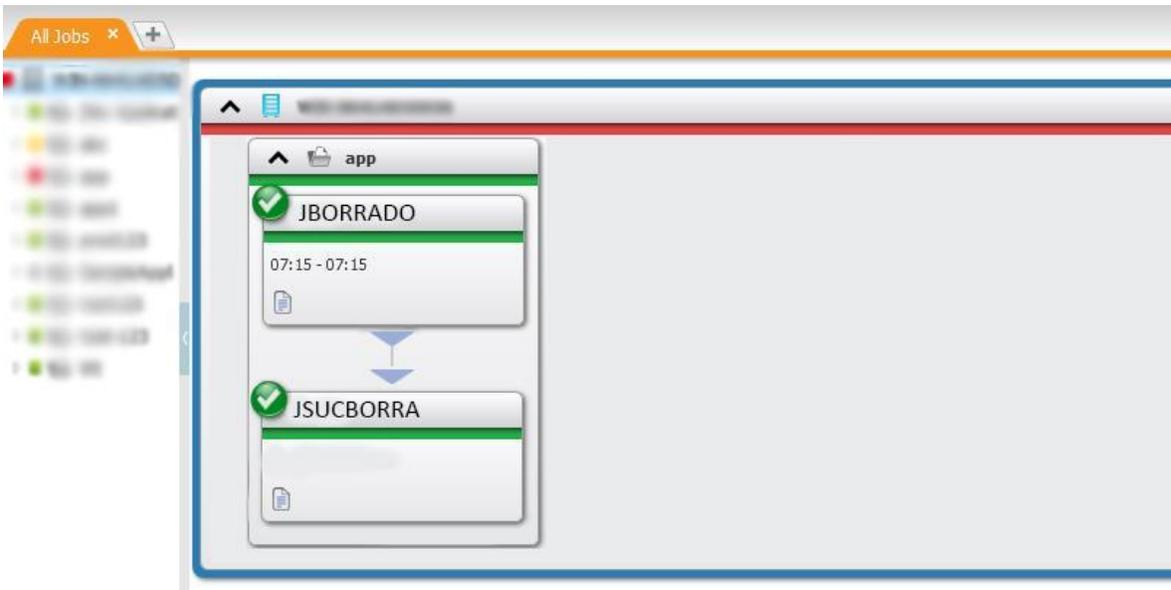


Figura 4.17. Diagrama de flujo en Control-M con jobs completados- Ejemplo de retraso # 1

Al no haber trabajado directamente en el job JSUCBORRA no será necesario documentarlo en SNOW, con la documentación hecha sobre el fallo de JBORRADO es suficiente.

4.5 Ejemplo de retraso #2:

Muchas de las alertas de retrasos son avisos que los desarrolladores programan para procesos que son importantes, si un job debe de correr a determinada hora se puede agregar una alerta una hora antes para estar atentos a que corra a su debido tiempo. En esta suposición se tiene un job llamado JOB1234A programado para correr a las 5:00 PM y los desarrolladores han programado una alerta a las 4:00 PM, a esta hora Control-M muestra la alerta como en la ilustración 4.18.

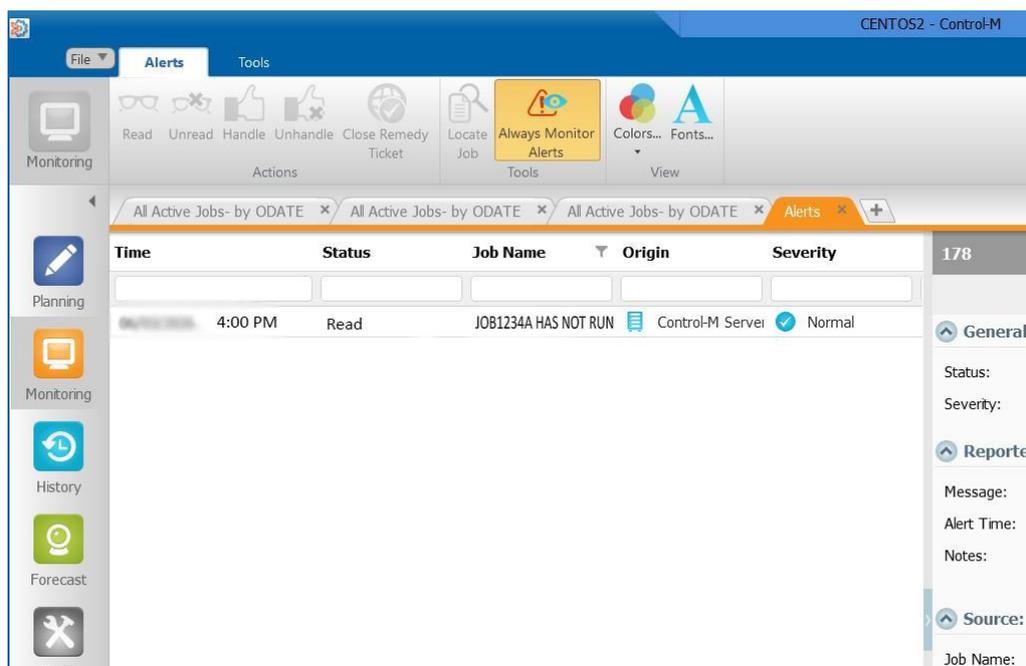


Figura 4.18. Control-M – Ejemplo de retraso #2.

Para saber qué está esperando el job para correr, se puede ir al diagrama de flujo, donde se muestra una pestaña con el título Waiting Info (Información de espera) donde se ve qué es lo que impide al batch correr como se puede ver en la ilustración 4.19.

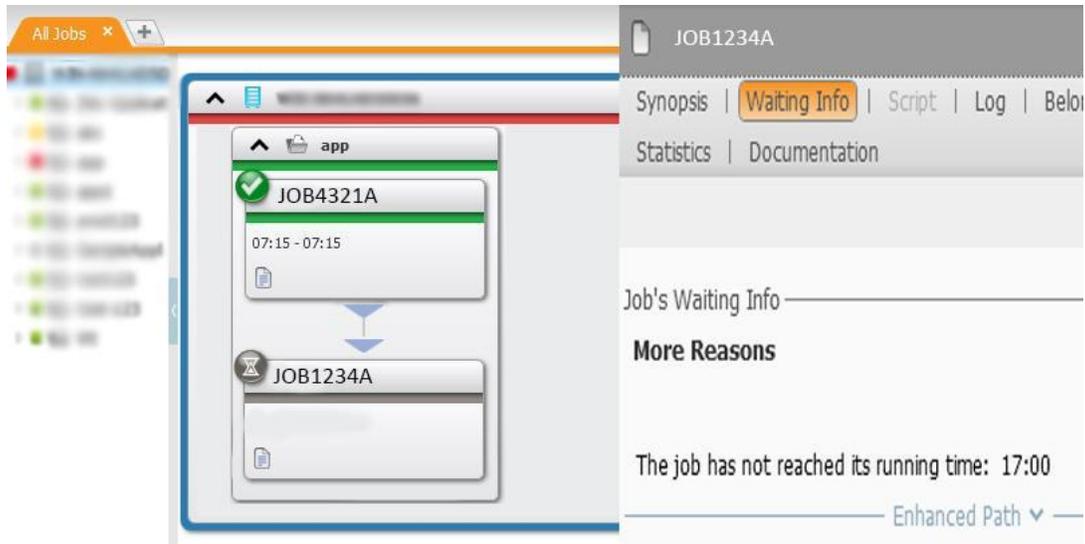


Figura 4.19. Diagrama de flujo en Control-M con información de espera- Ejemplo de retraso # 2.

Como se puede apreciar, JOB1234A tiene como predecesor a JOB4321A y tiene una condición de tiempo para ejecutarse a las 5:00 PM, al confirmar que su predecesor completó bien, se puede asegurar que JOB1234A se ejecutará sin problemas cuando se cumpla su condición de tiempo, en este caso solo queda esperar a que corra y quitar la alerta de Control-M, no es necesario documentar el proceso en SNOW ya que no hubo trabajo realizado sobre el proceso batch.

4.6 Ejemplo de retraso #3:

Hay algunos jobs que deben de correr al recibir un data set pero por decisión de los desarrolladores no cuentan con condiciones automáticas que lo hagan correr, por lo que es necesario revisar manualmente que se haya recibido el archivo y después ejecutar el batch job. Para este ejemplo se tiene JOB1FILE que cuenta con documentación hecha por los desarrolladores donde se establece que para ejecutarlo primero se debe de confirmar que exista el archivo XYZ.ARCHIVO.MANUAL. Al recibir la alerta Control-M la mostrará como se observa en la ilustración 4.20.

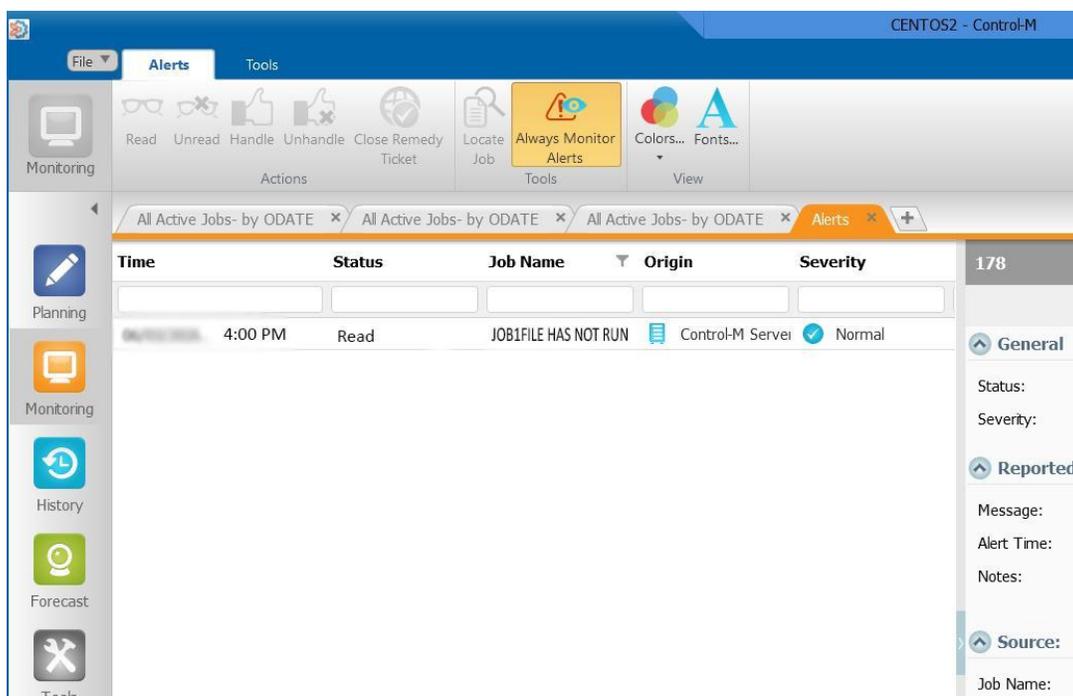


Figura 4.20. Control-M – Ejemplo de retraso #3.

Para esto se siguen las instrucciones de los desarrolladores que deben estar en un artículo de SNOW que también se les conoce como KB por los primeros 2 dígitos de sus identificadores, se busca el nombre JOB1FILE y dará como resultado un documento parecido a lo que se muestra en la ilustración 4.21.

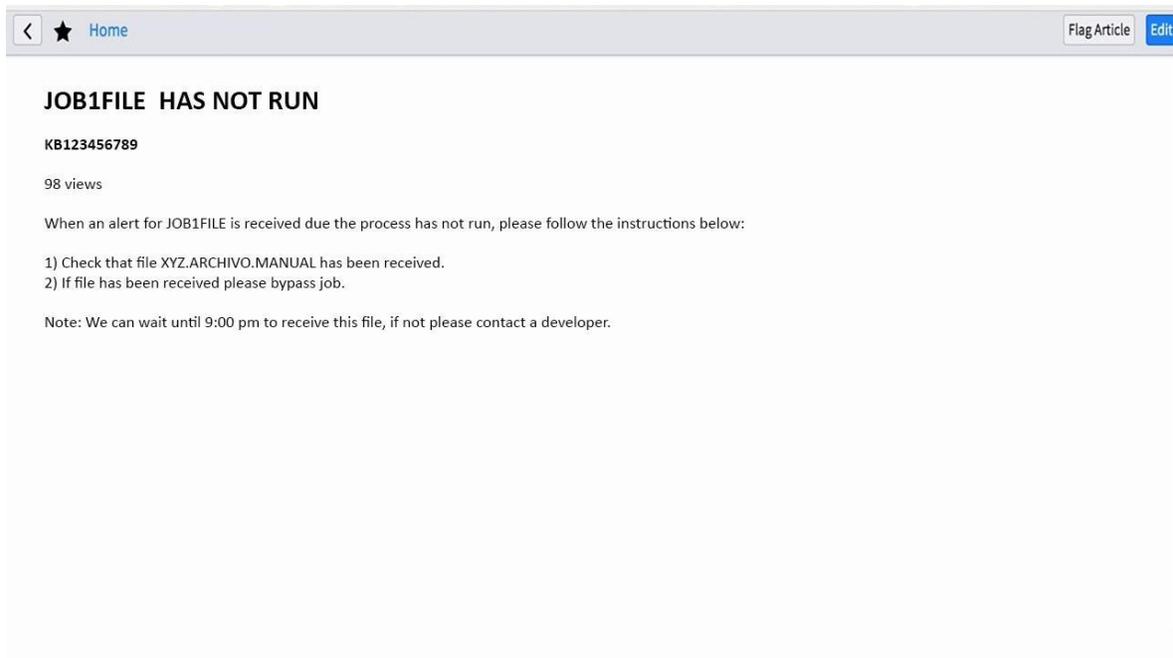


Figura 4.21. Artículo (KB) en SNOW - Ejemplo de retraso #3

***Traducción:** Cuando se recibe una alerta para JOB1FILE porque el proceso no se ha ejecutado, siga las instrucciones a continuación:*

- 1) Verificar que se haya recibido el archivo XYZ.ARCHIVO.MANUAL.*
- 2) Si se ha recibido el archivo, ejecute el job.*

Nota: Podemos esperar hasta las 9:00 p. m. para recibir este archivo; de lo contrario, comuníquese con un desarrollador.

Para cumplir con el paso 1 de las instrucciones se debe usar el administrador de archivos de z/OS para verificar que el archivo existe, haciendo una búsqueda como se muestra en la ilustración 4.22.

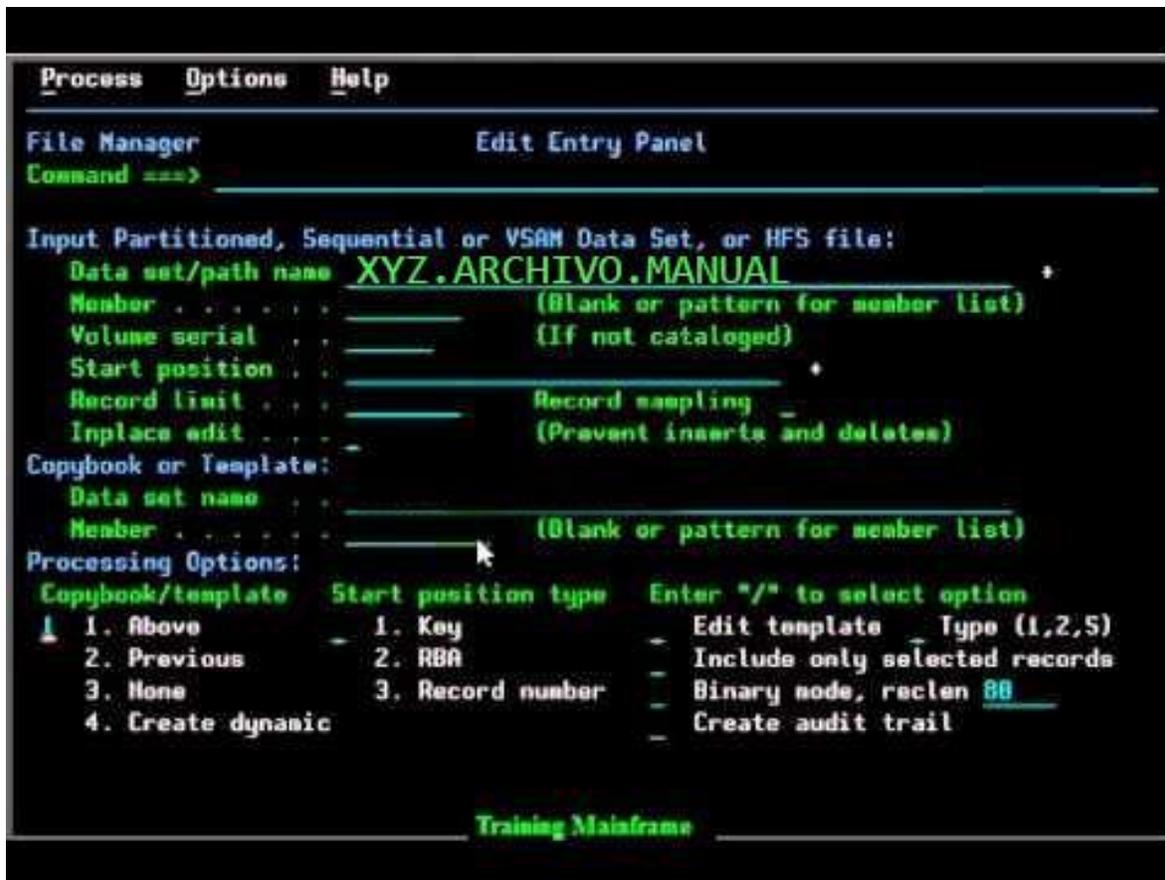


Figura 4.22. Búsqueda de archivos en Z/OS - ejemplo de retraso #3.
 Si el archivo existe lo muestra como en la ilustración 4.23.



Figura 4.23. Data sets encontradas - Ejemplo de retraso #3.

Ya que se confirma que el data set sí se encuentra en el sistema se procede a ejecutar el job mediante una petición por SNOW como se muestra en la ilustración 4.24.

The screenshot shows a web-based form for creating a new incident in the SNOW system. The form is titled 'New' and is part of a workflow that includes 'Assess', 'Authorize', 'Scheduled', 'Implement', 'Review', 'Closed', and 'Canceled' stages. The incident is currently in the 'New' stage. The form fields are as follows:

- Number:** A text input field with a blue highlight.
- Type:** A dropdown menu set to 'Emergency'.
- Requested by:** A text input field with a search icon and a help icon.
- State:** A dropdown menu set to 'New'.
- Category:** A dropdown menu set to 'Service Request'.
- Conflict status:** A dropdown menu set to 'Not Run'.
- Configuration item:** A text input field with a search icon, a help icon, and a delete icon.
- Conflict last run:** A text input field.
- Priority:** A dropdown menu set to '4 - Low'.
- Assignment group:** A text input field with a search icon and a help icon.
- Risk:** A dropdown menu set to 'Moderate'.
- Assigned to:** A text input field with a search icon.
- Impact:** A dropdown menu set to '3 - Low'.
- Short description:** A text input field containing 'Bypass job JOB1FILE'.
- Description:** A text area containing 'Please bypass job JOB1FILE with today's order date'.

At the bottom of the form, there are tabs for 'Planning', 'Timeline', 'Details', 'Notes', and 'Attachments'.

Figura 4.24. Petición para correr job en SNOW - Ejemplo de retraso #3.

Traducción: Por favor ejecuta job JOB1FILE con la fecha de orden de hoy.

El incluir la fecha de orden es importante para poder diferenciar al job en caso de que existan instancias del mismo con fechas diferentes, también se puede agregar el identificador único de cada job. Una vez enviada la petición, el equipo encargado de los horarios de los procesos ejecutará JOB1FILE que terminará satisfactoriamente al estar presente el archivo XYZ.ARCHIVO.MANUAL.

Con esto se puede documentar el trabajo realizado en SNOW como se aprecia en la ilustración 4.25.

The screenshot shows a ServiceNow incident record for 'Incident - INC0086533'. The incident number is INC0086533. The state is 'New'. The time spent is 00:00:08. The incident was opened on 2017-01-10 12:52:52. The short description is 'JOB1FILE LATE SHOUT'. The reassignment count is 0. The incident classification is 'Incident Classification *'. The related records, related knowledge articles, and closure information are also visible. The additional comments section contains the text: 'As per KB job bypassed after confirming file XYZ.ARCHIVO.MANUAL was available.'

Figura 4.25. Documentación en SNOW - Ejemplo de retraso #3.

Traducción: Según KB el job se ejecutó después de confirmar que el archivo XYZ.ARCHIVO.MANUAL estaba disponible.

Después de ver estos ejemplos, hay que tener claro que no todos los jobs tienen la misma relevancia, hay unos que manejan datos críticos y otros no tanto, por lo que al tener múltiples alertas se debe tener a consideración cuáles son los procesos prioritarios, cuando un job importante se puede ver afectado por alguna falla, se debe de contactar a un desarrollador para que este al tanto de que tal vez se pueda necesitar de su ayuda, incluso existen jobs que se espera que corran satisfactoriamente siempre, en caso de que presenten fallas inmediatamente se debe abrir una junta en Zoom con desarrolladores y personal de las áreas afectadas para trabajar inmediatamente en el error y solucionarlo lo más rápido posible para reducir al mínimo el impacto que pueda provocar.

5 CAPITULO QUINTO. CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES SOBRE EL PROCESO

De acuerdo con lo observado en los resultados obtenidos, se puede decir que los objetivos planteados inicialmente en este trabajo se cumplen todos los días. El objetivo general consiste en mantener los procesos batch funcionando fluidamente y mantener los fallos al mínimo para mantener a los usuarios y clientes de la empresa con servicio lo más cerca posible al 100% del tiempo, este objetivo se lleva a cabo diario como se muestra en el capítulo 4 de este documento.

Mediante las técnicas y herramientas proporcionadas por la empresa, es posible monitorear, analizar y solucionar los procesos que se encuentren con fallos de una manera satisfactoria.

Valorando la eficacia y efectividad, considero que el proceso que se sigue para solucionar los problemas que se presentan en los procesos batch es el óptimo, tomando en cuenta que todo siempre se puede mejorar, al momento de redactar este trabajo, el cliente, TCS y sus empleados, están satisfechos con el proceso y su calidad, ya que en la mayoría de casos se minimiza o anula el impacto negativo que pueden llegar a crear los fallos y/o retrasos de los procesos batch, por lo que el cliente le puede brindar el mejor servicio posible a sus usuarios. Se puede concluir que los resultados que obtenemos día a día son buenos, sin embargo, los desarrolladores deben de poner mayor atención en reducir las alertas creando programas que no fallen.

5.2 CONCLUSIONES PERSONALES

Hace más de 10 años empezó mi vida laboral, he tenido trabajos en los que he aprendido muchas cosas, habilidades y conocimientos técnicos, sobre relaciones humanas y como trabajar bajo presión y en este trabajo en TCS es donde más he puesto en práctica todo lo aprendido en experiencias anteriores y donde considero que he reforzado mis conocimientos y más he ampliado mis conocimientos, al

contratarme para este puesto, creí que me darían un puesto de desarrollador web o java, ya que eran las habilidades donde tenía más experiencia, pero durante la entrevista me ofrecieron un puesto donde tendría que aprender todo desde cero, ya que la tecnología que usaría ya no se enseña en las escuelas, cualquier información en internet sobre las principales herramientas que adoptaría es escasa, por lo que dude un momento, pero vi una oportunidad de obtener nuevas habilidades en un lugar donde estaban dispuestos a pagarme por hacer un trabajo en el cual no tenía experiencia, eso fue hace más de 2 años y sigo aprendiendo cosas nuevas todos los días.

Considero que mi trabajo contribuye al mantenimiento y disponibilidad de servicios que usan millones de personas todos los días, y me ha ayudado a comprobar que el área de las TI es inmenso, ya que mi actividad profesional es una de las que tenía total desconocimiento de su existencia antes de ser contratado.

Mi proceso de profesionalización no ha concluido ya que sigo aprendiendo todos los días, de hecho, está lejos de terminar, ya que de momento solo le doy mantenimiento a programas escritos en JCL que a su vez hacen llamados a programas escritos en COBOL, actualmente me encuentro tomando un curso de desarrollo para este lenguaje que me ayuda a tener un mejor entendimiento de los procesos aplicados en mi trabajo y tal vez en un futuro también desarrollar.

5.3 RECOMENDACIONES

El software que maneja el cliente es eficiente en su mayoría, pero hay unas excepciones que vale la pena resaltar, Control-M es uno de los programas más inestables que he usado dentro de mi experiencia laboral, en una jornada normal se debe de reiniciar al menos 3 veces, por lo menos 1 vez se congela y por lo menos 2 veces deja de mostrar jobs, afortunadamente con solo reiniciar la aplicación vuelve a funcionar normalmente, pero esto toma alrededor de 7 minutos, el cliente está consciente de las fallas que presenta esta aplicación y ha brindado una alternativa de la misma aplicación pero en una versión web, que es más rápida pero muy limitada la información que puede brindar sobre los procesos, muchas veces se

tiene que trabajar con la aplicación de escritorio y web al mismo tiempo para obtener los mejores resultados, personalmente no tengo experiencia con otras alternativas a Control-M pero definitivamente le recomendaría al cliente buscar otro proveedor para el software de monitoreo.

Por otra parte, tenemos los programas dedicados a la comunicación de los equipos, el correo electrónico de Outlook, hace aproximadamente 1 año estaba limitado a 1GB por cuenta, por lo que no se podía tener correos con más de 10 días de antigüedad, era realmente molesto iniciar la jornada laboral eliminando correos para poder recibir nuevos, el cliente aumentó la capacidad y ahora se tiene un periodo de 3 meses, sin importar el tamaño de los archivos adjuntos y no se tienen que borrar manualmente ya que los que excedan 3 meses son borrados automáticamente, quitando la molestia de limpiar las bandejas cada ciertos días, y aunque hubo una gran mejora en este aspecto, 3 meses siguen pareciéndome poco, se tienen jobs que se ejecutan 1 vez al año, por lo que si se manda un correo sobre alguno de estos procesos y la misma situación se vuelve a presentar es de gran ayuda contar con la información manejada en esos correos electrónicos y no solo lo documentado en SNOW, por lo que recomendaría al cliente aumentar la capacidad de Outlook a por lo menos 18 meses. Estas son las 2 principales recomendaciones que se tienen para el cliente, sin embargo, hay cosas que mejorar en TCS.

En esta empresa se fomenta mucho el aprendizaje, incluso cuenta con su propia plataforma de cursos que cualquier empleado puede usar, esta cuenta con temas diversos orientados principalmente a las TI y es obligatorio su uso, lo que considero un desperdicio de recursos ya que la plataforma no cuenta con los mejores cursos ni todos los temas de interés para todos los empleados, mi recomendación para TCS es fomentar el aprendizaje a través de cualquier plataforma y no obligar a usar la suya, que debería de ser solo una opción más.

También, tenemos una mejorable evaluación de desempeño laboral, TCS no cuenta con aspectos definidos a evaluar, cada año la empresa califica a sus empleados con lo que se conoce como bandas, dependiendo del criterio de diferentes departamentos se le otorga una banda con calificación A, B o C al empleado, donde

A es la mejor posible y C la peor posible, esta calificación se hace llegar por medio de una plataforma de la empresa sin ningún tipo de detalles, ni explicación de cómo se llegó a esa banda, lo que es muy perjudicial para los empleados que obtienen una B o una C, ¿Qué pasa si ellos quieren subir su calificación para el siguiente año? Al no haber retroalimentación, los empleados que consideren que merecían una mejor banda o que cumplieron con todas sus metas durante el año, probablemente solo les quede esperar que para la próxima no sean tan duros con la evaluación. Por lo que recomiendo a TCS crear o adoptar una guía de desempeño laboral con un proceso sistemático de estimación cuantitativa y cualitativa del grado de eficacia de sus empleados, donde con argumentos se demuestre si las personas cumplen o no con el trabajo en el tiempo establecido, de la forma en cómo el cliente y TCS lo solicitaron y donde se pruebe si se alcanzan o no los resultados deseados, ya que su actual método parece ser aleatorio.

6. REFERENCIAS BIBLIOGRÁFICAS Y VIRTUALES

[1] Mainframe - <https://www.ibm.com/es-es/topics/mainframe> consultada en junio de 2022.

[2] JCL - <https://www.mainframes.com/JCL.html> consultada en junio de 2022.

[3] data sets - <https://www.ibm.com/docs/en/zos-basic-skills?topic=zc-zos-storage-constructs-file-systems-data-sets-more> consultada en junio de 2022.

[4] Z/OS - <https://es.wikipedia.org/wiki/Z/OS> consultada en junio de 2022.

[5] Data Set Utility - <https://www.ibmmainframer.com/tso-ispf/dataset-utility-panel/> consultada en junio de 2022.

[6] Batch Jobs - <https://www.ibm.com/docs/en/zos-basic-skills?topic=jobs-what-is-batch-processing> consultada en junio de 2022.

[7] COBOL - <https://es.wikipedia.org/wiki/COBOL> consultada en junio de 2022.

[8] Java - [https://es.wikipedia.org/wiki/Java_\(lenguaje_de_programaci%C3%B3n\)](https://es.wikipedia.org/wiki/Java_(lenguaje_de_programaci%C3%B3n)) consultada en junio de 2022.

[9] REXX - <https://es.wikipedia.org/wiki/REXX> consultada en junio de 2022.

[10] C - [https://es.wikipedia.org/wiki/C_\(lenguaje_de_programaci%C3%B3n\)](https://es.wikipedia.org/wiki/C_(lenguaje_de_programaci%C3%B3n)) consultada en junio de 2022.

[11] PL/1 - <https://es.wikipedia.org/wiki/PL/1> consultada en junio de 2022.

[12] Ensamblador - https://es.wikipedia.org/wiki/Lenguaje_ensamblador consultada en junio de 2022.

[13] TCS - https://en.wikipedia.org/wiki/Tata_Consultancy_Services consultada en junio de 2022.

[14] IBM - <https://es.wikipedia.org/wiki/IBM> consultada en junio de 2022.

[15] SNOW - <https://www.servicenow.es/products/itsm.html> consultada en junio de 2022.

[16] Control-M - <https://www.bmcsoftware.es/it-solutions/control-m.html> consultada en junio de 2022.

[17] Windows 10 - https://es.wikipedia.org/wiki/Windows_10 consultada en junio de 2022.

[18] Microsoft - <https://es.wikipedia.org/wiki/Microsoft> consultada en junio de 2022.

[19] Zoom - <https://www.pocket-lint.com/es-es/aplicaciones/noticias/151426-que-es-zoom-y-como-funciona-mas-consejos-y-trucos> consultada en junio de 2022.

[20] Skype - <https://support.skype.com/es/faq/fa6/que-es-skype> consultada en junio de 2022.

[21] Slack - <https://slack.com/intl/es-mx/help/articles/115004071768-%C2%BFQu%C3%A9-es-Slack> consultada en junio de 2022.

[22] Suite Office - https://es.wikipedia.org/wiki/Microsoft_Office consultada en junio de 2022.

[23] Outlook - https://es.wikipedia.org/wiki/Microsoft_Outlook consultada en junio de 2022.

[24] SuperC - <https://ruifeio.com/2017/07/11/using-superc-to-search-for-text-strings/> consultada en junio de 2022.

[25] IEFBR14 - <https://www.mainframestechhelp.com/utilities/iefbr14/delete-datasets.htm> consultada en junio de 2022.

[26] Administrador de archivos de Z/OS - <http://www.universocobol.com/2014/11/> consultada en junio de 2022.

[27] HTML5 - <https://es.wikipedia.org/wiki/HTML5> consultada en junio de 2022

[28] https://es.wikipedia.org/wiki/Job_Control_Language consultada en junio de 2022.