

Caso de estudio en la industria privada. Migración de información a un nuevo Sistema de Adquisiciones en plataforma AS/400

VEYNA-LAMAS, Manuel*†, VELA-DÁVILA, José y VELÁZQUEZ-MACÍAS, Jesús.

Recibido Julio 4, 2016; Aceptado Septiembre 7, 2016

Resumen

La implantación de un nuevo sistema de cómputo implica varios retos, uno de los más comunes es la migración de información ya existente a la nueva plataforma. En los sistemas AS/400 un problema común es la sobrecarga de tareas y por lo tanto la disminución de desempeño del procesador, por tal motivo se debe tener cuidado de utilizar algoritmos optimizados al momento de crear aplicaciones que se ejecuten bajo este entorno, para no afectar a otros usuarios y el desempeño general del sistema. En el presente trabajo se describe un caso de estudio real llevado a cabo en una compañía cervecera en el estado de Zacatecas, México, que en ese momento utilizaba un sistema de cómputo multiusuario de la compañía IBM, llamado AS/400. El caso consistió en la solución dada a un problema de desempeño en la migración de información de un sistema antiguo a uno nuevo el cual correría bajo el mismo entorno. El problema fue que la solución original de migración de datos implicaba el llamado excesivo a procesos, los cuales eran repetitivos y lentos. Esto ocasionó que el procesador del sistema AS/400 se sobrecargara y disminuyera drásticamente su capacidad de procesamiento y desempeño, hasta llegar al punto de afectar a otras áreas de la compañía. Por lo tanto no era factible.

AS/400, RPG, *INLR, RETRN, migración, desempeño

Abstract

The implementation of a new computer system involves several challenges, one of the most common is the migration of existing data to the new platform. In the AS/400 systems a common problem is the overload of tasks and thus reducing processor performance, for that reason should be careful to use optimized algorithms when creating applications that run under this environment, so as not to affect other users and the overall system performance. This paper describes a real case study conducted in a beer maker company in the state of Zacatecas, Mexico, which at that time used a multiuser computer system by IBM, known as AS/400. The case involved the solution given to a performance problem in the migration of information from an old system to a new one which would run under the same platform. The problem was that the original data migration solution involved the so-called excessive processes, which were repetitive and slow. This overloaded the processor of the AS/400 and drastically decreased processing capacity and performance, to the point of affecting other areas of the company. Therefore it was not feasible.

AS/400, RPG, *INLR, RETRN, migration, performance

Citación: VEYNA-LAMAS, Manuel, VELA-DÁVILA, José y VELÁZQUEZ-MACÍAS, Jesús. Caso de estudio en la industria privada. Migración de información a un nuevo Sistema de Adquisiciones en plataforma AS/400. Revista de Sistemas Computacionales y TIC'S 2016, 2-5: 42-52

* Correspondencia al Autor (Correo Electrónico: mveyna@upz.edu.mx)

† Investigador contribuyendo como primer autor.

Introducción

El uso e implantación de sistemas de información implica diversas actividades, no solamente el desarrollo, sino también y muy frecuentemente la transferencia inicial de la información histórica que ya maneja el usuario, ya sea que esta provenga de plataformas ajenas o de algún sistema ya existente que será reemplazado. Esta actividad no siempre es tomada en cuenta en el ciclo de desarrollo cualquiera que sea la metodología utilizada, lo que ocasiona trabajo no planificado desencadenando retrasos en los planes previstos.

Además durante este tipo de procesos y otros casos como son las horas pico o cierres mensuales, la carga de trabajo del equipo de cómputo se incrementa de forma considerable por lo que es recurrente la lentitud de procesamiento durante estos periodos, pero la industria requiere que sus tiempos se reduzcan para ser más eficientes buscando con ello el aumento en su productividad.

El caso de estudio se realizó en una empresa ubicada en el estado de Zacatecas, líder en la elaboración, distribución y venta de cerveza en México, actualmente parte importante de otra empresa cervecera líder a nivel global. Al implantar sistemas comunes a todas las empresas filiales, sí se planificó la migración de la información histórica del sistema de adquisiciones antiguo al nuevo, pero se presentó un caso en el que era tal la cantidad de datos que ocasionaba gran carga de trabajo al procesador del equipo AS/400 de IBM, además de que al tratar de transferir el total de la información mencionada era tan lento que no se lograba en los tiempos destinados a ello.

El sistema AS/400 es una plataforma de cómputo de IBM tipo mainframe de gama media y alta para todo tipo de empresas y grados departamentos¹. Cuenta con la base de datos relacional DB2/400 integrada en el sistema operativo, además de sus características de alta disponibilidad y desempeño. Soporta varios lenguajes de programación, siendo en ese momento el más utilizado el RPG (Report Program Generator) en su versión III, y cuyas aplicaciones se conocen como tipo OPM (Original Program Model). La versión IV conocida como ILE-RPG (Integrated Language Environment – RPG) soporta la integración de varias aplicaciones como un módulo único, además de permitir el formato libre de las instrucciones, no siendo obligatorio el estricto formato posicional.

Contexto

El grupo cervecero posee ocho fábricas en el territorio nacional, así como otras compañías filiales para distribución y venta, además de empresas de transporte y otros servicios; fábricas de malta, envases y empaque para sus productos².

Para reunir y consolidar la información en el corporativo ubicado en la Ciudad de México, cada una de las filiales debía realizar el desarrollo y adaptación de su software dado su modelo de desarrollo in house³, propagando así el mismo trabajo hacia todas sus empresas. Derivado de esto surgió la iniciativa de crear sistemas de información homogéneos para todo el grupo, siendo los precursores los sistemas de “Contabilidad y Finanzas” y un sistema para el registro y control de las adquisiciones conocido como “Sistema Único de Compras”, a los cuales luego se sumaron otros más.

¹ <https://es.wikipedia.org/wiki/AS/400>

² www.gmodelo.mx

³ Cada empresa contaba con su departamento de desarrollo interno de software.

Al ser sistemas que serían estandarizados en todas las filiales, se tomó en cuenta que varios de ellos se encontraban enlazados a otros también antiguos, tanto como en el uso directo de archivos⁴ como en el llamado a programas externos con rutinas para la obtención de datos. Para solventar esto y permitir transportar este software en forma independiente sin depender de aplicaciones externas a él, éstas fueron creadas como aplicaciones denominadas “ligas o enlaces” en las bibliotecas⁵ locales de cada compañía, siendo esto un trabajo previo planificado antes de la puesta en marcha.

El primer sistema en ser implantado en las filiales de mayor importancia fue el Sistema de Contabilidad y Finanzas el cual no tuvo mayores incidencias para su puesta en marcha. Fue seguido por el Sistema Único de Compras. Para éste sistema se requería transferir la información histórica de todos los pedidos de materiales, materias primas, y cualquier otra compra realizada por cada compañía donde fuese implantado. Para la correcta operación fue necesario capacitar en cada empresa en el uso de este sistema al personal que en ellas labora y que tiene la atribución de solicitar y/o autorizar alguna compra para la operación de la compañía. Dado que se trataba de un sistema de información nuevo y robusto, se había realizado previamente la implantación en dos fábricas cerveceras de tamaño pequeño y mediano, y el tercer turno correspondió a la que está ubicada en el estado de Zacatecas, siendo la más grande en Latinoamérica.

⁴ Se conocen como archivos las tablas de la base de datos relacional.

⁵ En la plataforma AS/400 la base de datos relacional y todos los componentes se manejan como objetos, siendo conocidas como bibliotecas las bases de datos que contienen las tablas de cada sistema de información, y puede también contener otros objetos como los programas ejecutables, consultas, etc.

Metodología, equipos y herramientas

La utilización de cualquier equipo o software requiere la capacitación al personal involucrado, tanto a nivel operativo como al técnico. En esta etapa se realizan las pruebas para verificar la concordancia de la información antigua con la que se ha migrado, para lo cual es indispensable contar con información real y no solamente con la que se genera por el nuevo sistema. Es aquí donde el aplicativo original para la transferencia de los datos históricos desarrollado por el corporativo se implantó y adecuó a las características locales del sistema antiguo. Como la duración de la transferencia se estimaba en varias horas, se ejecutaba solamente por periodos cortos de tiempo para efectos del comparativo descrito. Por la duración tan corta de las pruebas se transferían solamente algunos cientos de registros, de lo cual no se valoró su impacto en tiempo para la posterior transferencia completa, pues se atribuyó a que como el sistema AS/400 estaba operando bajo la carga normal de trabajo durante el periodo laboral, provocaba que la migración no fuera tan rápida como se esperaba, retrasando los planes previstos de inicio de trabajo con el nuevo sistema.

Método de investigación

El trabajo de investigación consistió en analizar las causas de tan lentos tiempos de respuesta, tanto por la poca cantidad de registros transferidos como el impacto en los tiempos de respuesta de la plataforma de cómputo a los trabajos operativos de los usuarios. Los dos aspectos iniciales a revisar son la posible inconsistencia en la información antigua, y la lógica de programación del código del aplicativo.

La primera consistió en revisar los datos contenidos en las tablas para comprobar que todos los datos son consistentes y válidos mediante consultas de SQL (Structured Query Language) y el apoyo del gestor de consultas integrado Query/400, mientras que en el análisis del código se comprueba que no hay instrucciones susceptibles al bloqueo de registros, con lo que se determinó que estas situaciones no son la causa del problema. Bajo estas circunstancias se planificó la ejecución de una transferencia completa durante un periodo no laboral con el objetivo de contar con datos de la eficiencia en un periodo de tiempo mayor. Para tal fin se recomendó destinar la totalidad de los recursos del servidor AS/400 a este trabajo, sometiéndolo en un subsistema de memoria exclusivo creado para tal efecto y sin que hubiera actividad en el subsistema QINTER⁶ al cual se asignan los trabajos interactivos de los usuarios. La administración de memoria en la plataforma AS/400 consiste en segmentarla en bloques llamados "subsistemas" para asignarles una cantidad específica que sea adecuada a los procesos que se ejecutan, permitiendo con ello el balance en los tiempos de respuesta, además de proporcionarles una configuración inicial a cada usuario.

La prueba se ejecutó en estas condiciones controladas durante un lapso de 13 horas, al final del cual se encontró que apenas sí se había transferido un 2.23% de los registros, siendo 47,000 de un total de 2'100,000. Los registros transferidos se refieren no solamente al de la tabla principal de pedidos de compra, sino también a los relacionados en otras tablas.

⁶ La plataforma AS/400 se divide en subsistemas en los que se agrupan los trabajos de acuerdo a su naturaleza. De usuarios interactivos (QINTER), trabajos programados y en segundo plano (QBATCH), de comunicaciones, etc.

Un pedido tiene relacionadas cotizaciones, tipos de cambio, proveedores y sus giros, artículos y productos, pasivos contables e información de pagos en el sistema de Cuentas por Pagar, y etapas de autorización, y a su vez generando mas registros en las tablas del nuevo sistema dado el nuevo diseño normalizado y estándares de programación modernos. Debido a la tasa tan baja de transferencia se revisaron los *logs* que genera el sistema operativo, que son las anotaciones que se guardan sobre la actividad de los trabajos que corren en esta plataforma. Esta inspección detallada no arrojó ningún dato que indique la causa de la lentitud del proceso, determinando así que se debe a la codificación del programa migrativo. De aquí se decide repetir la prueba pero ahora modificando el orden de la información origen, para que se inicie por la mas reciente, lo cual además contribuyó a que en las pruebas administrativas pudieran realizar el comparativo de su información mas actual y pruebas en paralelo. Se sometió nuevamente la ejecución del proceso el cual tuvo una duración de 14 horas antes de su interrupción. Como se esperaba, el resultado fue similar al anterior pues se transfirieron apenas poco mas de 51,000 registros, un 2.4% del total.

Aunque no se recomendaba sino hasta que se determinara objetivamente la razón de la ineficiencia del proceso, se planificó una tercera prueba, luego de la cual se revisa la cantidad de información transferida no siendo relevante la diferencia respecto a la obtenida en las ejecuciones anteriores, llegado a menos 57,000 registros.

Medición de resultados del Testing

La siguiente fase de la implantación del nuevo Sistema Único de Compras requería contar con la migración del total de la información, para lo cual se programó la tarea para ejecutarse durante un fin de semana completo, periodo no laborable en la empresa, dado que en las pruebas anteriores no habían concluido.

Transcurrieron 43 horas, tiempo en el cual se inicia nuevamente la operación de la empresa por lo que al encontrar que no había concluido el proceso se aborta para la verificación de resultados. El conteo de los datos migrados fue de 566,000 pedidos siendo apenas un 27% (Tabla 1).

El cálculo del tiempo total que se requeriría para completar este trabajo se realizó mediante la obtención de la ecuación de la línea de tendencia central a través del método de mínimos cuadrados, ajustándolo a un polinomio de primer grado.

Horas	Registros Transferidos
13	47,000
14	51,000
14	56,700
43	566,000

Tabla 1 Medición de la eficiencia del aplicativo que realiza la migración de datos.

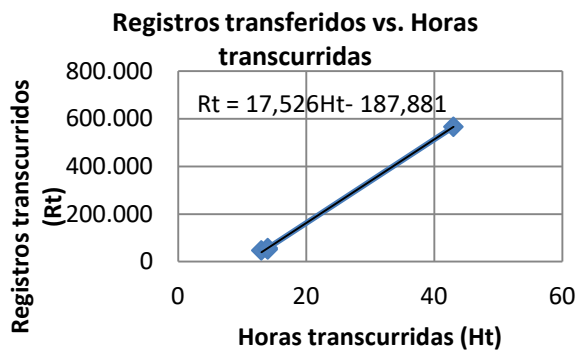


Gráfico 1 Medición de la eficiencia del aplicativo que realiza la migración de datos.

$$Rt = 17,526Ht - 187,881 \tag{1}$$

De la ecuación (1) despejamos la variable independiente que correspondiente a las horas transcurridas, y se hace la extrapolación al total de registros de datos a transferir ($Rt = 2'100,000$):

$$Ht = \frac{(Rt + 187,881)}{17526} = \frac{2',100,000 + 187,881}{17,526} = 130.54 \text{ horas} \tag{2}$$

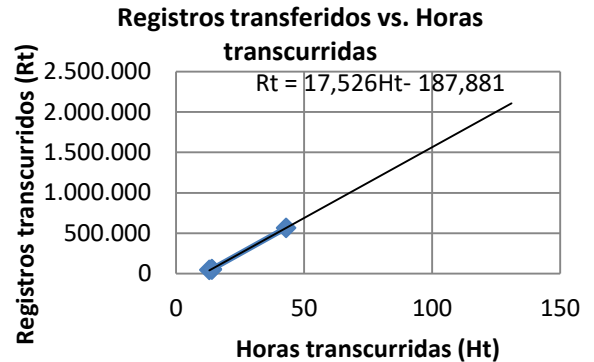


Gráfico 2 Extrapolación a escenario con la totalidad de los datos.

Con este resultado se estimó que la duración sería de casi 131 horas (Gráfico 2).

Estos cálculos se habían efectuado sin considerar el inicio del proceso cuyo punto es el origen (0,0). Así que se complementó la tabla de datos de medición (Tabla 2), con lo que se obtiene otra ecuación que es también analizada para ver si varía notablemente el tiempo proyectado.

Horas	Registros Transferidos
0	0
13	47,000
14	51,000
14	56,700
43	566,000

Tabla 2 Medición de la eficiencia del aplicativo que realiza la migración de datos.

$$Rt = 14,366Ht - 97,213 \tag{3}$$

De la misma manera se calcula la línea de tendencia central y se despeja la variable del número de horas transcurridas:

$$Ht = \frac{(Rt + 97,213)}{14,366} = \frac{2',100,000 + 97,213}{14,366} = 152.95 \text{ horas} \tag{4}$$

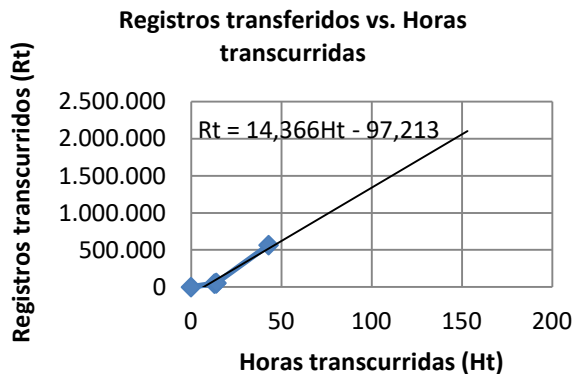


Gráfico 3 Medición de la eficiencia del aplicativo que realiza la migración de datos considerando el origen y extrapolando con el total de los datos.

Con esto se proyectan casi 153 horas, y dadas las dos estimaciones equivalen en promedio a 6 días laborales completos y eso contando con que el servidor estuviera dedicado solamente a esto proceso, lo cual resultaba totalmente inviable.

Al haber descartado ya los bloqueos de registros por uso del sistema de compras, se planteó la teoría de que podrían ocurrir bloqueos a nivel tabla por el proceso de respaldo de las bibliotecas de usuario, el cual estaba programado automáticamente para ejecutarse diariamente durante la madrugada. Esto implicaba cancelar dicho respaldo. Rápidamente también se rechazada debido que al ejecutarse éste en la parte final del periodo establecido, se deberían haber transferido la mayoría de los datos y no la minoría, además de que en la industria en una compañía tan grande es delicado no contar con un respaldo en caso de presentarse alguna contingencia.

Otra teoría propuesta era acerca de la mala sintonización⁷ del sistema AS400, lo cual inmediatamente se percibió poco factible y también se descartó ya que se estaba configurado como sintonización automática⁸ (tuning automático), y se había dedicado todo poder del servidor exclusivamente para la migración.

Optimización del software aplicativo

Es aquí donde recomiendan las técnicas de optimización en la codificación, pues en AS/400 así como en cualquier otra plataforma, el llamado a programas externos y la apertura y cierre de archivos para operaciones de I/O son procesos que generan gran carga de trabajo y tráfico.

Como resultado de la revisión al código, el diseño de la lógica no tiene problemas, pero tenían un exhaustivo llamado a rutinas externas que estaban modularizadas con el fin de encargarse de tareas repetitivas como lo son los cálculos de las fechas de entrega a partir de la fecha del pedido y los días de entrega del producto solicitado, o bien la operación inversa para calcular el número de días hábiles entre dos fechas dadas.

Una de estas rutinas que llamó rápidamente la atención es la llamada recurrente a los procesos que calculan los días hábiles entre dos fechas, lo cual se realiza haciendo una conversión a partir de que el día 01 de enero de 1901 le corresponde el día 1, el 2 de enero de 1901 al día número 2, y así sucesivamente, hasta llegar a la fecha recibida como parámetro. El número calculado se conoce como fecha juliana⁹.

⁷ Distribución de la memoria RAM de este equipo a los diferentes subsistemas que lo conforman para balancear o determinar su tiempo de respuesta.

⁸ Sintonización manual y automática.

⁹ Una fecha juliana se representa como un número entero consecutivo de acuerdo un criterio establecido.

Luego se hace lo mismo con la segunda fecha, y se calcula la diferencia en días, a lo cual sigue un llamado a un programa que lee de una tabla de días inhábiles y se le resta al número calculado anteriormente. Rápidamente se concluye que este proceso simplemente realiza demasiadas operaciones inútiles al realizar en promedio 2,000 operaciones en cada ciclo por cada conversión, pues los pedidos de compra mas antiguos eran del año 1995. Hasta 12 llamados por cada registro, dado que cada pedido de compra cuenta en promedio con tres cotizaciones que calculan la fecha de entrega prometida a partir de la fecha de fincamiento del pedido, además de que cada una cuenta con días o fecha de vigencia. $12 \times 2,000 = 24,000$ instrucciones por cada registro a transferir, que multiplicado por $2 \times 100,000$ da un total de 50,400,000,000 de instrucciones para los cálculos de fechas, además de las correspondientes a la lógica del propio programa principal.

Seguido de esto se ejecutó la transferencia para otra prueba adicional, y mediante las utilerías para el monitoreo de los trabajos se comprueba el frecuente llamado a programas externos mostrado por la pila de llamadas, TRA001/QCMDEXC/TRA002,TRA002A; confirmando así que el llamado al cálculo de fechas es un proceso que consume gran cantidad de tiempo.

TRA001 es el programa principal que lee la tabla de pedidos de compra del sistema anterior, y de cada registro lee los atributos correspondientes y a partir de ellos genera los equivalentes hacia las tablas en el nuevo sistema. QCMDEXC es el API que se llama implícitamente al hacer un CALL en el código RPG, TRA002 es la rutina externa que convierte una fecha a formato juliana y TRA002A realiza la operación inversa.

Un aspecto sumamente importante a considerar en el procesamiento masivo de información como en el presente caso de estudio, es que el llamado a cualquier programa en RPG ocasiona entre otras las siguientes operaciones, realizadas por la API implícita QRGXINIT:

- Se abren automáticamente los archivos que no están condicionados a apertura manual.
- Todas las variables definidas en el programa se inicializan a cero en el caso de ser numéricas, y a blanco (*BLANKS) en las de tipo alfanumérico.
- Los campos de las estructuras de datos se inicializan con el valor que corresponda.
- Todos los indicadores¹⁰ se inicializan en *OFF.

El otro aspecto de mayor impacto es que se cuenta con dos modos para que un programa RPG termine su ejecución.

1. Lo más común es encender el indicador *INLR (indicador de Last Record).
2. Mediante la sentencia RETRN.

Utilizar la primera opción ocasiona la asignación de valores a las variables de parámetro que devolverán algún valor, la asignación el valor cero o blancos las variables declaradas, y el cierre de archivos declarados en el programa. Con RETRN no se ejecutan estas operaciones si se queda *INLR con valor *OFF.

¹⁰ Los indicadores en RPG son banderas automáticas con valores de *ON y *OFF cuyo valor se asigna según el éxito o no de operaciones de lectura a los archivos, y también en operaciones de comparación, asignación de variables, llamado a programas externos, etc.

Para hacer eficiente el llamado a procesos recurrentes, se recomendó cambiar la programación para terminar la ejecución de los programas externos mediante RETRN, pero cambiando el diseño para tomar en cuenta que los archivos utilizados permanecerán posicionados donde hayan quedado y no se reinicia ninguna variable ni indicador. La siguiente llamada al programa RPG (dentro del mismo trabajo) no ejecutará QRGXINIT, por lo que se tendrá que planear y modificar la codificación para que el programador se asegure que todas las variables, el posicionamiento en los archivos y los indicadores se les asigne un valor inicial correspondiente, de lo contrario podrían presentarse fallas catastróficas en los cálculos si se asume que el RPG hará estas tareas durante cada invocación.

El otro aspecto equivocado en la programación de las mencionadas rutinas externas, es el desperdicio de procesamiento al realizar cálculos secuenciales sin sentido. Para subsanarlo se creó una tabla que contenía como campo clave las fechas en orden ascendente, un número consecutivo como fecha juliana, y otra columna adicional que indicaba si el registro correspondía a un día inhábil o no. De esta manera se hace una simple lectura directa con las fechas solicitadas se cuenta la diferencia en días y se lee en un ciclo solamente el intervalo de fechas para contar los días inhábiles, reduciendo con ello dramáticamente el total de instrucciones a ejecutar por la aplicación de migración.

Resultados

En lugar de 24,000 instrucciones en memoria por cada registro migrar, mas la consulta a la tabla de días inhábiles, se tuvieron solamente un promedio de 7 a 8 operaciones le lectura de este archivo, siendo éstas las dos lecturas de las fechas y un promedio de 5 a 7 para la búsqueda de los días inhábiles entre ellas, lo que evidencia una enorme diferencia de carga para el procesador del AS/400, además de que se eliminó el llamado implícito que se hace a QRGXINIT con las operaciones que ello ocasiona, quedando su ejecución únicamente al inicio del aplicativo principal.

Llegada la fecha planeada de migración definitiva de la información hacia el nuevo sistema, se puso en marcha la aplicación, encargando igualmente al operador del sistema AS400 el monitoreo de este proceso, a lo cual le dio seguimiento, y terminó completa y normalmente en un lapso de menos de 9 horas.

Conclusiones

Los programadores novatos y de mediana experiencia, son menos consientes del performance al ir aumentando el poder de procesamiento de los equipos de cómputo y la velocidad en las comunicaciones.

En sistemas de alta carga de trabajo la codificación de los algoritmos y procesos debe orientarse a la optimización para evitar consumir tiempo y recursos mediante la eliminación de operaciones no necesarias que pueden dar pie a situaciones en que no sean factibles los tiempos de respuesta o de implantación. Igualmente se debe considerar esto para procesos que se ejecutan durante horas de trabajo pico y no solamente en situaciones de carga promedio.

En el caso de la plataforma AS/400 esto se logra evitando el llamado recurrente a programas externos que utilicen *INLR, y en su lugar cambiarlos por RETRN, teniendo en cuenta la inicialización de variables e indicadores, y el posicionamiento de los archivos utilizados. La primera vez que es llamado un programa en RPG, éste inicializa automáticamente las variables e indicadores, pero en llamados subsecuentes no será así utilizando esta técnica, por lo que se deberá adaptar el código considerando esta situación. Las variables y arreglos que se requieran inicializar una sola vez, trasladarlos a la Subrutina Inicial (*INZSR). En otras plataformas como los lenguajes de programación orientada a objetos esto se traslada al Constructor. Si se requiere inicializar arreglos es más óptimo mediante la instrucción CLEAR que es ejecutado en una sola instrucción, en lugar de hacer un ciclo para inicializar elemento por elemento.

Analizar y visualizar el escenario en el que se realizará el trabajo. Si se ha realizado previamente en forma exitosa, se debe comparar el escenario con el nuevo que se pretende implantar, para considerar las características de cantidad de información, capacidad de procesamiento, carga de trabajo y concurrencia de operaciones, y no suponer que todo será igual, para calcular con esto los estimados del tiempo. Si se han realizado trabajos previos tanto definitivos como los realizados en la etapa de testing, a partir del volumen de información procesado se podrá hacer un cálculo cuantitativo del tiempo que tomará aplicar el proceso en el nuevo escenario mediante técnicas de extrapolación como lo son mediante la determinación de la curva de mejor ajuste, pudiendo ser de tipo polinomial a través del método de mínimos cuadrados, o bien de otra naturaleza de acuerdo a los valores medidos de datos y tiempos de respuesta.

Glosario

IBM

International Business Machines (Corp). Es una reconocida empresa multinacional estadounidense de tecnología y consultoría con sede en Armonk, Nueva York. Fabrica y comercializa hardware y software para computadoras, y ofrece servicios de infraestructura, alojamiento de Internet, y consultoría.

RPG

Report Program Generator. Lenguaje de programación desarrollado por IBM diseñado originalmente para generar informes comerciales o de negocios para sistema 360 y AS/400.

ILE-RPG

Integrated Language Environment – RPG. ILE fue una actualización masiva con muchas mejoras al ambiente de programación, y se diseñó para ser compatible con el ambiente RGP III existente. Entre ellas se mejoró la habilidad de integrar múltiples lenguajes.

CL

Control Language. Es el grupo de comandos (Lenguaje de scripts) que le dicen al sistema de dónde obtener la entrada de datos, cómo procesarla, y dónde poner los resultados, llamar a programas, etc. en la plataforma AS/400 de IBM.¹ Permite a los programadores y a los administradores del Sistema escribir programas utilizando los comandos del sistema operativo OS/400.

OPM

Original Program Model. Los lenguajes de programación de IBM para AS/400 que eran suficientes para la creación, compilación y manejo de programas, pero no permitían el llamado a procedimientos en otros lenguajesⁱⁱ. The set of functions, processes, and rules that are used to directly create and run a program object is known as the original program model (OPM)

API

La Interfaz de Programación de Aplicaciones, abreviada como API (del inglés: Application Programming Interface), es el conjunto de subrutinas, funciones y procedimientos (o métodos, en la programación orientada a objetos) que ofrece cierta biblioteca para ser utilizado por otro software como una capa de abstracción.

Indicadores

Se trata de 99 variables de tipo booleano, numerados del 01 al 99, que incluye el lenguaje RPG para indicar el resultado de las operacionesⁱⁱⁱ. Además hay otros indicadores de control (INL1...INL9, y el de último registro (*INLR).

Testing

Las pruebas de software (en inglés Software Testing) son las investigaciones empíricas y técnicas cuyo objetivo es proporcionar información objetiva e independiente sobre la calidad del producto a la parte interesada o stakeholder. Es una actividad más en el proceso de control de calidad

RISC

RISC (Reduced Instruction Set Computer) Computadoras con un Conjunto de instrucciones reducido. Arquitectura basada en registros de propósito general que operan siempre sobre operandos que se encuentran almacenados en el procesador, cerca de la unidad de ejecución.

Subsistema

Un subsistema es un único entorno operativo predefinido mediante el cual el sistema coordina el flujo de trabajo y el uso de los recursos.

Sintonización

Ajuste el balance de los recursos para obtener el mejor rendimiento de las aplicaciones.^{iv}

Performance

Desempeño con respecto al rendimiento de una computadora, un dispositivo, un sistema operativo, un programa o una conexión a una red. En informática, medida o cuantificación de la velocidad/resultado con que se realiza una tarea o proceso.

Bibliotecas

Una biblioteca es un objeto OS/400 que se utiliza para contener otros objetos OS/400 en la base de datos. Se organiza como jerarquía single-level (LIBRARY/OBJECT), desemejante de la estructura del directorio encontrada en las PC, por lo que no puede contener a otra biblioteca dentro.

Objetos

El sistema AS-400 consta de un único nivel de almacenamiento, todo lo que se almacena son objetos. Un objeto es aquello que no necesita de una herramienta para ser usado.

Puede ser una biblioteca, un archivo de datos, un archivo lógico, un archivo de

fuentes de código, un programa ejecutable, etc.

Consulta as400

Consulta/400 es un programa bajo licencia IBM que puede utilizarse para obtener información de la base de datos del AS/400. Puede obtener información de cualquier archivo de base de datos definido en el sistema utilizando especificaciones de descripción de datos (DDS) del Operating System/400 (OS/400)^v

Referencias

Chapra Chapra Steven C, (2007) Métodos numéricos para ingenieros, McGraw Hill, México.

Cruikshank Dan, Stewart Jame (1997). System iNEWS Magazine: <http://iprodeveloper.com/systems-management/case-leaving-las-vegas-lr>.

Kohan John (2003). System iNEWS Magazine: <http://search400.techtarget.com/tip/The-ins-and-outs-of-INLR>.

International Business Machines Corporation (1994) RPG/400 User's Guide, SC09-1816-00, McGraw Hill, México.

IBM AS/400 - IBM Archives (2003) A Brief History of the IBM AS/400 and iSeries: http://www-03.ibm.com/ibm/history/exhibits/rochester/rochester_4010.html

Rochester chronology - IBM Archives (2003) A Brief History of the IBM AS/400 and iSeries <https://www-03.ibm.com/ibm/history/documents/pdf/as400.pdf>