

ATTITUDES

Información sobre Power Systems, incluidos AS/400, iSeries y System i

Año 28 - Marzo-Abril 2014

Nº 296

Precio: 7 Euros

COLABORACIONES

La bomba con retardo del 4 GB Access Path Size

Muchas empresas trabajan con software para IBM i creado hace más de 10 años. Si bien les permite ejecutar las aplicaciones más antiguas en el hardware más actual, las aplicaciones de mayor antigüedad pueden causar problemas con ciertos ficheros que ya no son adecuados para el procesamiento de hoy en día. Esta semana, vamos a ver un parámetro presente en

ficheros antiguos que si no es modificado, puede detener el procesamiento de las aplicaciones: El tamaño de la vía de acceso de 4GB.

La bomba de relojería de la que hablo es el parámetro ACCPTHISZ (Access path size) de algunos ficheros. En las versiones antiguas del sistema operativo OS/400, precursor del actual IBM i OS, este pará-

metro por defecto estaba establecido en *MAX4GB. Con los ficheros *MAX4GB, el tamaño máximo de la vía de acceso al fichero no podía ser mayor de 4 gigabytes (4GB). Y con muchos ficheros que venían con software de aplicaciones de terceros el parámetro se establecía automáticamente en *MAX4GB.

Sigue en página 3

Utilizar triggers previos para permitir cambios

Recientemente, durante un proyecto de modernización, he estado utilizando la opción ALWREPCHG (Allow Repeated Change) con "before triggers" (triggers previos). El mandato ALWREPCHG permite un "before trigger" para realizar cambios en el registro que está siendo añadido o actualizado y hacer cierta magia con su base de datos.

El problema: Una tabla contiene una fecha almacenada en una columna numé-

rica empaquetada. Se necesita cambiar el tipo de dato de la columna a un tipo de dato de fecha más apropiado. Este cambio requerirá cambios de código de al menos 20 programas.

Pensar en cambiar, probar e implementar los 20 programas de un plumazo no es tan siquiera ni una opción. Los programas se irán cambiando según sea necesario durante varios meses.

Sigue en página 8

Cómo encontrar y corregir con el SQL los datos decimales inválidos

Estoy seguro de que a ningún lector de esta publicación le gustan los errores de datos decimales. Tan solo un único error puede arruinar un día entero. El sabio programador utiliza herramientas y técnicas para mantener a raya los datos numéricos inválidos. Una de esas herramientas que puede utilizar para encontrar

y corregir los datos inválidos es el SQL.

Supongamos que un archivo físico tiene un número de cuenta de cliente almacenado como decimal empaquetado de 7 dígitos, sin posiciones decimales. Supongamos que hay uno o más registros en blanco en el archivo físico.

Sigue en página 15

SUMARIO

Colaboraciones

La bomba con retardo del 4 GB Access Path Size	3
Utilizar triggers previos para permitir cambios	8
SQL Técnicas especiales	11
Como encontrar y corregir con el SQL los datos decimales inválidos	15

Novedades

Acceso a las bases de datos remotas	2
Transferencia segura de archivos entre diferentes plataformas	6
Envío de archivos de base de datos y de ficheros spool desde el System i	16

