

# MINE SIGHT®

in the Foreground

Volumen 22, Número 10, Noviembre del 2006

## Current Affairs

# Nuevas rutinas de control de procesos en MineSight® V3.60

Este artículo trata dos rutinas de comandos generales que se incluyen en el CD de Actualización: **contoursFromDir.py** y **partialsFromDir.py**.

Estas dos rutinas procesan múltiples objetos de geometría tales como los que habitualmente se encuentran en las aplicaciones de control de procesos.

### Opción para contornear todas las superficies en un directorio – contoursFromDir.py

La rutina o script, **contoursFromDir.py**, crea objetos geométricos que tienen líneas de contorno para todas las superficies y sólidos en determinado directorio. Esto es muy útil cuando las superficies a contornear son varias. Dichas superficies pueden residir en uno o varios objetos geométricos. Este script creará objetos geométricos con líneas de contorno para cada elemento sólido o de superficie de cada objeto de geometría dentro del directorio. Se trata de una rutina de comandos EM-GRAIL que solo puede ejecutarse desde MineSight® 3-D.

Para ejecutar el archivo de comandos, utilice la opción **Run Script...** en la barra de menú de MineSight® (**File | Scripts | Run Script**) y busque la rutina **contoursFromDir.py** con la opción **Browse**:

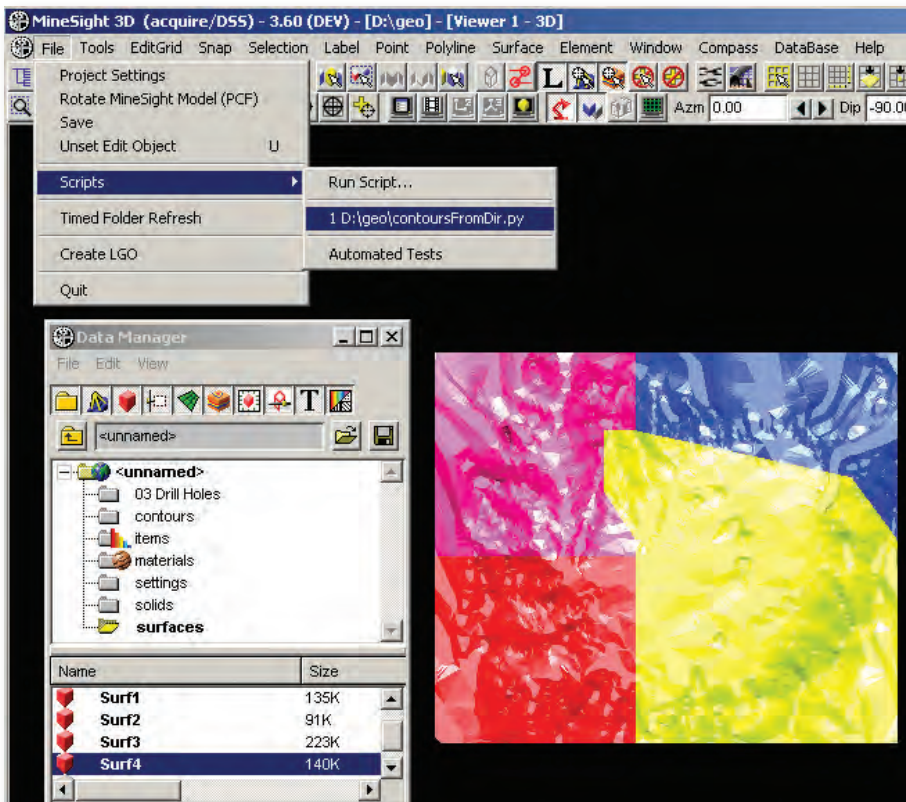


Figura 1 Ejecutar rutina desde MineSight® 3-D

(continúa en página 3)

### Temas en esta edición:

- Aumento del Precio de la Licencia MineSight®... página 2
- Calendario de Entrenamiento 2006... página 11
- Current Affairs
  - Nuevas rutinas de control de procesos en MineSight® v.3.60... página 1
- Directorio de Mintec... página 2
- Exposiciones y Seminarios... página 12
- Mintec, Inc. abrirá una Nueva Oficina en Australia, Perth... página 2
- Seminario en Tucson... página 12
- Tips from Tech Support:
  - Cargar datos de compositos a MineSight®... página 7

# Current Affairs

*A Window on Software Engineering*

(viene de la página 1)

La rutina de comandos presenta una interfaz gráfica de usuario (GUI) donde debe ingresar los parámetros requeridos, y, si lo desea, algunos datos opcionales (Figura 2).

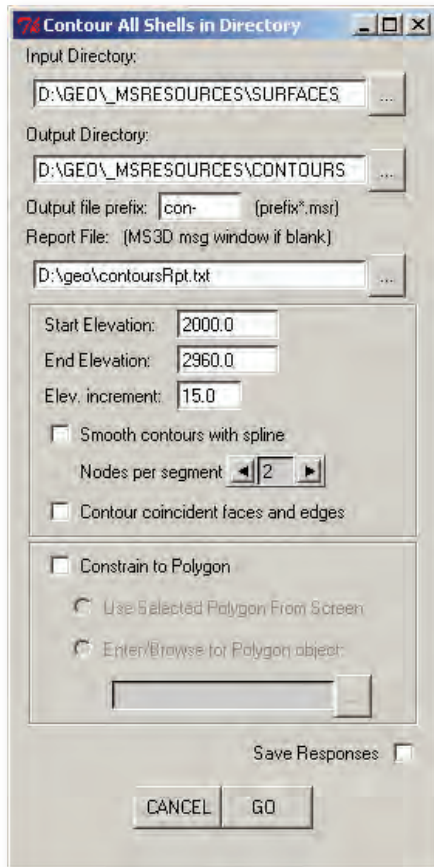


Figura 2 Interfaz gráfica del usuario

Los parámetros requeridos son el directorio de entrada, **Input Directory**, el directorio de destino, **Output Directory**, el prefijo del archivo generado, **Output File Prefix**, la elevación inicial y la final, **Start** y **End Elevation**, y el valor de incremento de la elevación, **Elevation Increment**.

El **Input Directory** es el directorio donde residen los objetos geométricos con superficies y sólidos.

El **Output Directory** es un directorio donde se crean los nuevos objetos de contorno.

El **Output File Prefix**, es el prefijo empleado para conformar el nombre del objeto con los contornos para cada **.msr** de entrada. Si no escribe ningún prefijo, por defecto se emplea el prefijo "con-." El nombre del archivo generado se construye de la siguiente manera: prefijo + nombre del archivo de entrada.

A su vez, las opciones para elevación inicial, final e incremento de la elevación, **Start Elevation**, **End Elevation**, y **Elevation Increment**, respectivamente, se emplean

para especificar las elevaciones para la operación de contorneado.

El archivo de informe, **Report File**, es la ruta donde se escribe el informe de auditoría de la rutina ejecutada con los detalles del procesamiento de los distintos objetos. Si deja este campo en blanco, la información de la auditoría se escribirá en la ventana de mensajes, Message Window, de MineSight® 3-D.

Más adelante en este mismo artículo, discutiremos las opciones para alisar contornos con afinamiento, **Smooth contours with spline**, para contornear caras y aristas coincidentes, **Contour coincident faces and edges**, para limitar al polígono, **Constrain to Polygon** y para guardar las respuestas, **Save Responses**.

En los ejemplos que aparecen en las Figuras 1 y 2, el directorio de entrada, **Input Directory**, se denomina **SURFACES** y contiene cuatro superficies en los objetos **Surf1-Surf4**. A su vez, el directorio de salida, **Output Directory**, solicitado recibe el nombre de **CONTOURS**. En este caso se requieren los contornos para las elevaciones de 2000 a 2960 a intervalos de 15. El informe de auditoría se escribirá en el archivo **contoursRpt.txt**.

Una vez cargados los parámetros deseados, presione la tecla **GO** para iniciar la operación de contorneado. Una vez completado el contorno, aparecerá el mensaje **Processing Complete** en la parte inferior de la ventana.

Los archivos que contienen los contornos (**con-Surf1.ms**, **...con-Surf4.ms**) se crearon en el directorio **msresources\CONTOURS**; regenere el directorio de salida (**CONTOURS**), con la opción Refresh en el administrador de datos, Data Manager, para que estos objetos estén disponibles en la sesión actual de MineSight® 3-D (Figura 3).

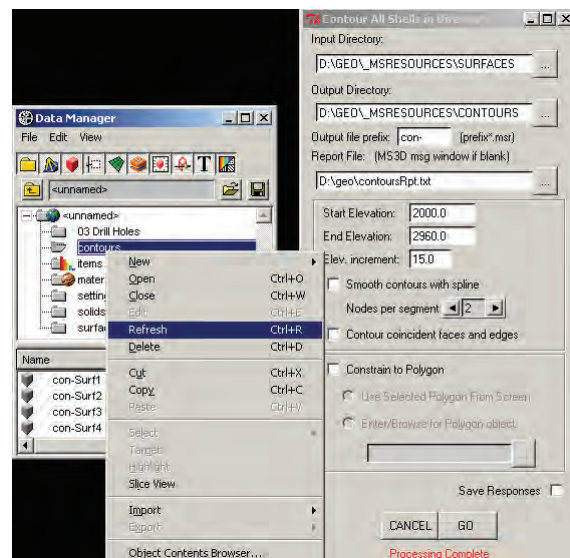


Figura 3 Regenerar el directorio de salida después de que se completa el procesamiento. (continúa en página 4)

(viene de la página 3)

La Figura 4 muestra los resultados de la operación de contornear, con los cuatro objetos nuevos (**con-Surf1** – **con-Surf4**) abiertos. La Figura 5 presenta el contenido del archivo de informe/auditoria, **contoursRpt.txt**, tal como se abre en el bloc de notas.

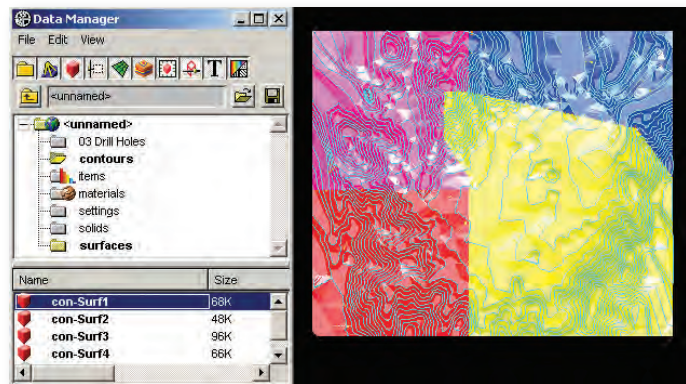


Figura 4 Contornos resultantes



Figura 5 Mensajes de informe/auditoria

Las otras opciones disponibles son equivalentes a sus correspondientes en el diálogo de contornos Contour de MineSight® 3-D.

### Smooth contours with spline

Si tilda la opción para alisar contornos con afinamiento, **Smooth contours with spline**, el elemento resultante se suaviza aplicando el algoritmo de alisado con afinamiento a la polilínea/polígono en todos los nodos dentro del elemento. Este algoritmo de alisado con afinamiento no respeta nodos al alisar curvas. El valor entero para los nodos por segmento se escribe en **Nodes per segment**, y especifica cuántos nodos se han de agregar durante el proceso de alisamiento.

### Contour coincident faces and edges

Dado que la rutina de comandos genera contornos exactos, si un contorno cae sobre una arista expuesta en la superficie, el algoritmo normal modificará la escala del

contorno eb una pequeña magnitud para encontrar una locación que no corresponda con una arista. En algunas situaciones puede que el usuario desee realizar un contorno exacto de los datos; en ese caso, tilde el casillero para contornear caras y aristas coincidentes, **Contour coincident faces and edges**, lo que significa que el usuario acepta que el contorno sea siguiendo las aristas. Esta opción puede ser necesaria para las triangulaciones que hayan sido importadas de otros sistemas, o si se ha creado una triangulación sin la nueva opción para minimizar área plana, **Minimize Flat Area**, en el diálogo triangulación.

### Constrain to Polygon

Para limitar los contornos resultantes con un límite poligonal, tilde el casillero **Constrain to Polygon**, y luego elija la opción para emplear el polígono seleccionado en pantalla: **Use Selected Polygon from Screen**; o para indicar/ubicar un objeto geométrico que contiene un elemento poligonal: **Enter/Browse for Polygon Object**. Tenga en cuenta que si las superficies a procesar son muchas, la opción limitar al polígono: **Constrain to Polygon**, puede resultar en un proceso prolongado. El medidor de progreso de MineSight® 3-D muestra el avance del proceso.

La Figura 6 presenta los resultados del proceso de contorno con todas las opciones adicionales: alisar contornos con afinamiento **Smooth contours with spline**, 4 nodos por segmento, contornear caras y aristas coincidentes (**Contour coincident faces and edges**) y limitar a un polígono (**Constrain to Polygon**) utilizando un polígono seleccionado en pantalla.

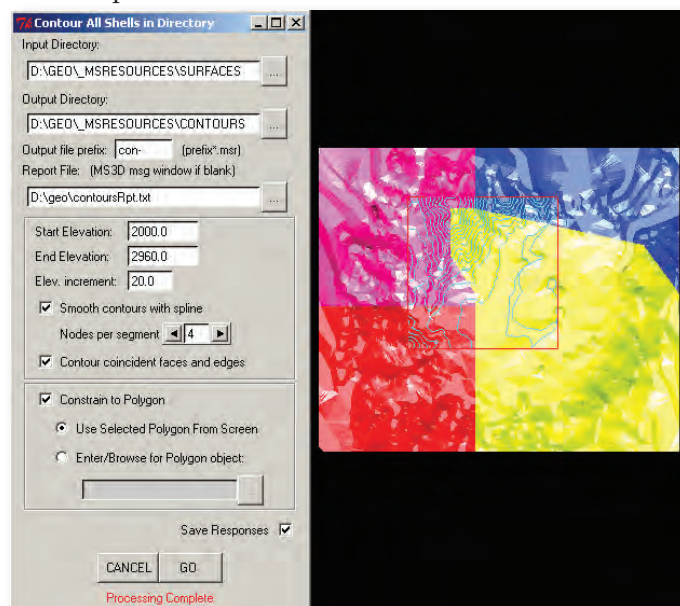


Figura 6 Contornos con la función limitar a un área cuadrada sobre cuatro superficies diferentes

### Save Responses

Si marca el casillero para guardar respuestas **Save Responses**, la información que proporcionada en la Interfaz Gráfica de Usuario se almacena en un archivo de

(continúa en página 5)

(viene de la página 4)

configuración denominado **contoursFromDir.ini**, en el mismo directorio donde reside la rutina de comandos. Esta información se emplea para cargar la GUI la próxima vez que se ejecute la rutina. La Figura 7 presenta el contenido del archivo **contoursFromDir.ini**.

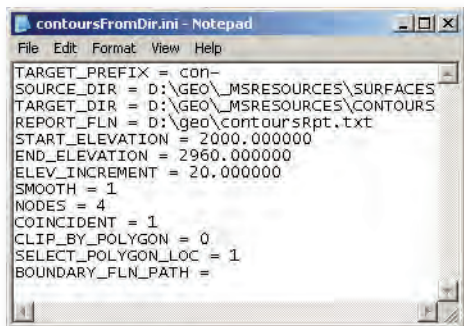


Figura 7 Ejemplo de archivo .ini

### Opción para crear parciales para todos los sólidos en un directorio: Create Partial for all Solids in Directory - partialsFromDir.py

La rutina de comandos **partialsFromDir.py** realiza un ciclo a través de un directorio inspeccionando todos los archivos .msr en busca de sólidos. Por cada elemento sólido que encuentra, crea el archivo parcial correspondiente. Esta rutina de comandos puede ejecutarse desde MineSight® 3-D, en forma independiente desde la línea de comandos o bien desde otro programa. Puede correr en forma silenciosa, utilizando la información de los parámetros desde el archivo **partialsFromDir.ini**, o bien puede presentar una interfaz (GUI) para que el usuario ingrese los parámetros.

La rutina, **partialsFromDir.py**, utiliza la información en un Archivo de Control de Proyecto (PCF) MineSight® para definir la forma en que se crean los archivos de parciales. Los valores del PCF se emplean como argumentos del procesador Voxel independiente, para computar los archivos de parciales.

Si el script se ejecuta desde MineSight® 3-D, siempre se presentará la interfaz de usuario tal como aparece en la Figura 8. Si se corre desde la línea de comandos o si se ejecuta desde otro programa, la GUI se suprime si despliegua la rutina con un argumento uno: **C:\> partialsFromDir.py 1**.

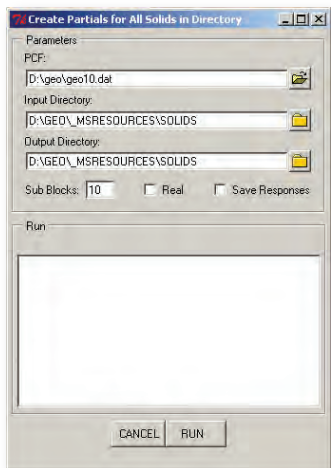


Figura 8 Interfaz del usuario de partialsFromDir.py

Para poder ejecutar la rutina el usuario debe especificar el PCF y los directorios de entrada en **Input** y de salida en **Output**. En el ejemplo de la Figura 8, el archivo PCF es **D:\geo\geo10.dat**, el directorio de entrada es **D:\geo\\_msresources\solids** y el de salida es **D:\geo\\_msresources\solids**.

Los parciales siempre se generan como porcentajes de bloque (valores de 0 a 100) y por defecto se presentan como enteros, pero el usuario tiene la opción **Real** que le permite solicitar parciales en valores reales (con precisión de 3 decimales).

También existe una opción, casillero de **Sub Blocks**, para alterar el conteo por defecto de (10) sub-bloques que es el que se emplea para computar valores de parciales integrados.

Si guarda los parámetros, tildando la opción **Save Responses**, la rutina se podrá ejecutar sin la interfaz gráfica. Los parámetros también se utilizarán para configurar la GUI cuando vuelva a ejecutar. Seleccione la opción **Save Responses** y los parámetros se almacenarán en el archivo **partialsFromDir.ini** en el directorio actual.

En el ejemplo de la Figura 9, el directorio de entrada contiene dos objetos: **solid1** y **solid3**. El objeto **solid3** tiene dos elementos sólidos mientras que **solid1** tiene un único elemento sólido.

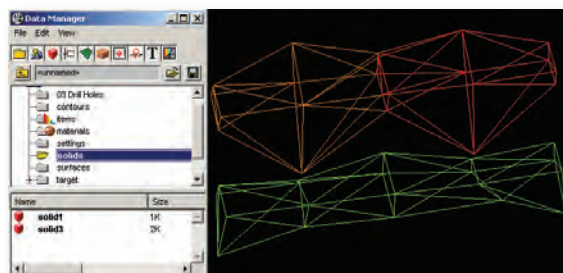


Figura 9 Dos objetos con tres sólidos en total

Después de cargar los Parámetros, tal como se muestra en la Figura 10, y al presionar el botón para ejecutar **RUN**, la rutina de comandos crea archivos de parciales para cada sólido dentro del directorio indicado en **Input Directory** y los coloca en el directorio que aparece en **Output Directory**.

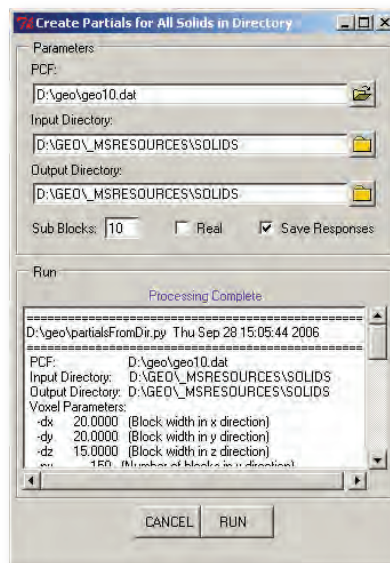


Figura 10: Procesamiento completado.

(continúa en página 6)

(viene de la página 5)

Cuando se completa el proceso, aparecen los detalles (auditoría) de la corrida en la ventana **Run** de la interfaz de usuario. La Figura 11 presenta un ejemplo de auditoría que incluye los parámetros que se emplearon para crear los parciales y la información sobre cada operación para cada uno de los sólidos procesados.

```

=====
D:\geo\partialsFromDir.py Thu Sep 28 15:05:44 2006
=====
PCF:          D:\geo\geo10.dat
Input Directory:  D:\GEO\_MSRESOURCES\SOLIDS
Output Directory: D:\GEO\_MSRESOURCES\SOLIDS
Voxel Parameters:
-dx      20.00 (Block width in x direction)
-dy      20.00 (Block width in y direction)
-dz      15.00 (Block width in z direction)
-nx      150 (Number of blocks in x direction)
-ny      125 (Number of blocks in y direction)
-nz      64 (Number of blocks in z direction)
-xmin    1000.00 (Min model limit in x direction)
-ymin    4000.00 (Min model limit in y direction)
-zmin    2000.00 (Min model limit in z direction)
-sb      10 (Sub blocks)
-----
Input geometry object:  solid1.msr
Output partials file 1: solid1-1.prt
voxel ...
Reading data from disk ... done
-----
Input geometry object:  solid3.msr
Output partials file 1: solid3-1.prt
voxel ...
Reading data from disk ... done

Output partials file 2: solid3-2.prt
voxel ...
Reading data from disk ... done

-----
Total geometry objects processed: 2
Total elements processed: 3
=====

```

Figura 11 Ejemplo de la información de auditoría para una corrida completa.

Se crea un archivo de parcial independiente para cada sólido que haya en el directorio. El nombre del archivo de parciales generado se construye de la siguiente manera: (nombre del objeto) + (-número de elemento) + (.prt). Para los tres sólidos en nuestro ejemplo, los archivos de parciales resultantes se denominan **solid1-1.prt**, **solid3-1.prt**, y **solid3-2.prt**.

La Figura 12 muestra el contenido de uno de los archivos de parciales creados (solid-1.prt). La 13 presenta el archivo .ini creado cuando se marca la opción Save Responses.

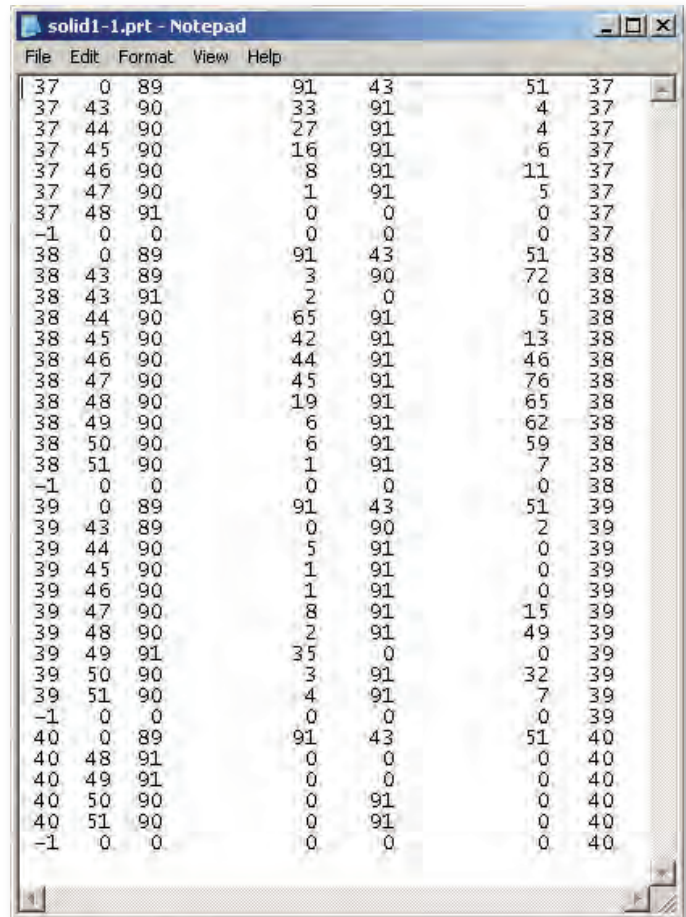


Figura 12 Ejemplo del contenido del archivo de parciales

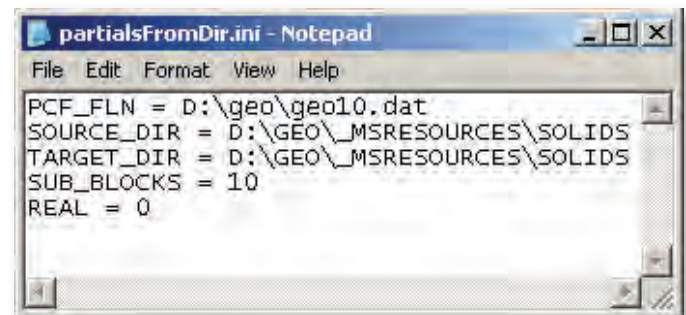


Figura 13 Ejemplo de archivo .ini.

Para ejecutar las dos rutinas de comandos descritas, la biblioteca de rutinas Grail debe estar en la variable de entorno PYTHONPATH, y los procesadores independientes **Clip.exe** (para **contoursFromDir.py**) y **Voxel.exe** (para **partialsFromDir.py**) deben estar en la variable de entorno PATH.