

Preguntas Más Frecuentes (FAQ)

Questions

1. [¿Por qué la lectura de una hoja de cálculo simple lleva tanto tiempo?](#)
2. [¿Qué es el "eventmodel" \(modelo de evento\) de HSSF?](#)
3. [¿Por qué no puedo leer el documento que creé utilizando Star Office 5.1?](#)
4. [¿Por qué recibo una excepción cada vez que intento leer mi hoja de cálculo?](#)
5. [¿Soporta HSSF hojas de cálculo protegidas?](#)
6. [¿Cómo se sabe si un campo contiene una fecha con HSSF?](#)
7. [Estoy intentando ver un fichero XLS enviado como flujo \(stream\) desde un servlet y tengo complicaciones. ¿Cuál es el problema?](#)
8. [Quiero dar formato a una celda \(Data format of a cell\) de una hoja excel como ###,###,###.#### o ###,###,###.0000. ¿Es posible hacer esto con POI?](#)
9. [¿Cómo añadir un borde alrededor de una celda unida \(merged\)?](#)
10. [Intenté escribir valores en celdas así como cambiar el nombre de la hoja Excel en mi lengua nativa, pero no pude hacerlo. :\(](#)

Answers

1. ¿Por qué la lectura de una hoja de cálculo simple lleva tanto tiempo?

Probablemente hayas habilitado el registro (logging). Dicho registro es una herramienta útil para la búsqueda de errores (debug). Tenerlo habilitado reducirá el rendimiento en un factor de al menos 100. El registro es útil para comprender por qué POI no puede leer algún fichero o para el propio desarrollo de POI. Los errores importantes se lanzan como excepciones, lo cual significa que probablemente no necesites el registro (log).

2. ¿Qué es el "eventmodel" (modelo de evento) de HSSF?

El paquete "eventmodel" de HSSF es un nuevo API para la lectura más eficiente de ficheros XML. Requiere mayor conocimiento por parte del usuario, pero reduce el consumo de memoria a una décima parte. Está basado en el modelo de eventos AWT en combinación con SAX. Si necesita acceso de sólo-lectura a un fichero XML determinado, esta es la mejor manera de hacerlo.

3. ¿Por qué no puedo leer el documento que creé utilizando Star Office 5.1?

Star Office 5.1 escribe algunos registros utilizando el viejo estándar BIFF. Esto provoca algunos problemas con POI que sólo soporta BIFF8.

4. ¿Por qué recibo una excepción cada vez que intento leer mi hoja de cálculo?

Es posible que su hoja de cálculo contenga alguna característica que no esté soportada actualmente por HSSF. Por ejemplo - hojas de cálculo que contengan celdas con formato RTF (rich text) no están soportadas actualmente.

5. ¿Soporta HSSF hojas de cálculo protegidas?

Al proteger una hoja de cálculo, ésta se cifra. No tocaremos el cifrado, porque no tenemos el suficiente conocimiento legal y no estamos seguros de las implicaciones que conllevaría el intentar implementar esto. Si desea intentarlo, es libre de hacerlo y de añadirlo como un módulo enchufable (plugin). Sin embargo, no lo guardaremos aquí.

6. ¿Cómo se sabe si un campo contiene una fecha con HSSF?

Excel almacena las fechas como números. Así la única manera para determinar si una celda está realmente almacenada como una fecha consiste en mirar su formato. Hay un método de ayuda (helper) en HSSFDateUtil (desde la distribución 1.7.0-dev) que lo comprueba. Gracias a Jason Hoffman por proporcionar la solución.

```
case HSSFCell.CELL_TYPE_NUMERIC:
    double d = cell.getNumericCellValue();
    // test if a date!
    if (HSSFDateUtil.isCellDateFormatted(cell)) {
        // format in form of M/D/YY
        cal.setTime(HSSFDateUtil.getJavaDate(d));
        cellText =
            (String.valueOf(cal.get(Calendar.YEAR)).substring(2);
        cellText = cal.get(Calendar.MONTH)+1 + "/" +
            cal.get(Calendar.DAY_OF_MONTH) + "/" +
            cellText;
    }
```

7. Estoy intentando ver un fichero XLS enviado como flujo (stream) desde

un servlet y tengo complicaciones. ¿Cuál es el problema?

El problema normalmente se manifiesta como un montón de caracteres basura en la pantalla. El problema persiste incluso aunque hayas configurado el tipo mime correcto.

La respuesta breve es: no dependas de IE para mostrar un fichero binario. Escribe un documento adjunto como es debido si lo envías a través de un servlet. Toda versión de IE tiene diferentes fallos (bugs) en este sentido.

El problema en la mayoría de las versiones de IE reside en que no utiliza el tipo mime de la respuesta HTTP para determinar el tipo de fichero; en su lugar utiliza la extensión del fichero en la petición. Así podría añadir un **.xls** a su cadena de petición. Por ejemplo: `http://yourserver.com/myServlet.xls?param1=xx`. Esto se consigue fácilmente a través del mapeo de URL en cualquier contenedor servlet. A veces una petición como `http://yourserver.com/myServlet?param1=xx&dummy=file.xls` también funciona.

Para garantizar la correcta apertura del fichero en Excel desde IE, escribe tu fichero a un fichero temporal bajo su raíz web desde tu servlet. Envía entonces una respuesta http al navegador para que haga una redirección en el lado del cliente a tu fichero temporal. (Si haces una redirección en el lado del servidor utilizando `RequestDispatcher`, tendrás que añadir **.xls** a la petición como se ha mencionado más arriba)

Date cuenta de que cuando pides un documento que se abre con un manejador externo, IE a veces realiza dos peticiones al servidor web. Así que si tu proceso generador es pesado, tiene sentido escribir a un fichero temporal, para que peticiones múltiples utilicen el fichero estático.

Nada de esto pertenece a Excel. El mismo problema ocurre cuando intentas generar cualquier fichero binario dinámicamente a un cliente IE. Por ejemplo, si generas ficheros pdf utilizando [FOP](#), te encontrarás con los mismos problemas.

8. Quiero dar formato a una celda (Data format of a cell) de una hoja excel como ###,###,###.#### o ###,###,###.0000. ¿Es posible hacer esto con POI?

HSSF no soporta todavía formatos de datos personalizados, sin embargo, debería ser una facilidad razonablemente sencilla de añadir y aceptaremos gustosas contribuciones en este área.

Estos son los formatos incluidos que soporta:

<http://jakarta.apache.org/poi/javadocs/javadoc/org/apache/poi/hssf/usermodel/HSSFFormat java.html#H>

9. ¿Cómo añadir un borde alrededor de una celda unida (merged)?

Añade celdas vacías alrededor de donde las celdas hubieran estado normalmente y configura los bordes individualmente para cada celda. Probablemente mejoraremos HSSF en el futuro para facilitar este proceso.

10. Intenté escribir valores en celdas así como cambiar el nombre de la hoja Excel en mi lengua nativa, pero no pude hacerlo. :(

Por defecto HSSF utiliza valores de celdas y nombres de hoja en unicode comprimido, así que para soportar la localización deberías utilizar Unicode. Para hacerlo deberías configurarlo manualmente:

```
//  
// para el nombre de la hoja  
//  
HSSFWorkbook wb = new HSSFWorkbook();  
HSSFSheet s = wb.createSheet();  
wb.setSheetName( 0, "SomeUnicodeName", HSSFWorkbook.ENCODING_UTF_16 );  
  
//  
// para el valor de la celda  
//  
HSSFRow r = s.createRow( 0 );  
HSSFCell c = r.createCell( (short)0 );  
c.setCellType( HSSFCell.CELL_TYPE_STRING );  
c.setEncoding( HSSFCell.ENCODING_UTF_16 );  
c.setCellValue( "\u0422\u0435\u0441\u0442\u0435\u043d\u0430\u0435\u043c\u0430\u0442\u0430" );
```

Asegúrate de que haces la llamada a `setEncoding()` antes de llamar a `setCellValue()`, si no, lo que le pases no será interpretado correctamente.