

Jakarta POI - API Java Para Acceder Ficheros con Formato Microsoft

by Andrew C. Oliver, Glen Stampoultzis, Avik Sengupta, Agustín Martín Barbero

1. Propósito

El proyecto POI consiste en APIs para manipular varios formatos de ficheros basados en el formato de Documento Compuesto OLE 2 de Microsoft, utilizando Java puro. En concreto, se pueden leer y escribir ficheros MS Excel utilizando Java. Pronto se podrá leer y escribir ficheros Word utilizando Java. POI es su solución Java Excel así como su solución Java Word. En cualquier caso, tenemos un API completo para portar otros formatos de Documento Compuesto OLE 2 y todo aquel que quiera participar será bienvenido.

Entre los ficheros basados en el formato de Documento Compuesto OLE 2 de Microsoft se incluyen la mayor parte de los ficheros de Microsoft Office tales como XLS y DOC así como formatos de fichero basados en el API de serialización MFC.

Como regla general intentamos colaborar lo más posible con otros proyectos para proporcionar esta funcionalidad. Algunos ejemplos: [Cocoon](#) para el que hay serializadores para HSSF; [Open Office.org](#) con quienes colaboramos en la documentación del formato XLS; y [Lucene](#) para el que pronto tendremos intérpretes del formato de fichero. Cuando es práctico, donamos componentes directamente a los proyectos para dotarles de capacidad-POI.

1.1. ¿Por qué/cuándo utilizar POI?

Abordaremos esto a nivel de componente. POI se refiere al proyecto completo.

Así que, ¿por qué debería utilizar POIFS o HSSF?

Utilizarías POIFS si tuvieras un documento escrito en el Formato de Documento Compuesto OLE 2, probablemente escrito utilizando MFC, que necesitaras leer en Java. Alternativamente, utilizarías POI para escribir en el Formato de Documento Compuesto OLE 2 si necesitaras inter-operar con programas ejecutándose en la plataforma Windows. No nos estamos jactando cuando decimos que ¡POIFS es la adaptación más completa y correcta de

este formato de fichero hasta la fecha!

Utilizarías HSSF si necesitaras leer o escribir un fichero Excel utilizando Java (XLS). También se pueden leer y modificar hojas de cálculo utilizando este API, aunque ahora mismo la escritura está más madura.

2. Componentes a Día de Hoy

2.1. Visión General

Un pensamiento erróneo generalizado es que POI escribe ficheros Excel. POI es el nombre del proyecto. POI contiene varios componentes, uno de los cuales, HSSF, escribe ficheros Excel. Siguen a continuación los componentes del proyecto POI completo y un pequeño resumen de su propósito.

2.2. POIFS

POIFS es la parte más vieja y más estable del proyecto. Es nuestra adaptación del Formato de Documento Compuesto OLE 2 a Java puro. Soporta funcionalidad de lectura y escritura. Todos nuestros componentes se sirven de él por definición. Por favor, vea [la página del proyecto POIFS \[EN\]](#) para más información.

2.3. HSSF

HSSF es nuestra adaptación del formato de fichero de Microsoft Excel 97(-2002) (BIFF8) a Java puro. Soporta lectura y escritura. Por favor, vea [la página del proyecto HSSF \[EN\]](#) para más información.

2.4. HWPF

HWPF es nuestra adaptación del formato de fichero de Microsoft Word 97 a Java puro. Soporta lectura y escritura. Por favor, vea [la página del proyecto HWPF \[EN\] para más información](#). Este componente está en la fase inicial de diseño. Ya puede leer y escribir ficheros sencillos. ¡Entra!

2.5. HPSF

HPSF es nuestra adaptación del formato de conjunto de propiedades OLE 2 a java puro. Los conjuntos de propiedades se utilizan mayoritariamente para almacenar las propiedades de un documento (título, autor, fecha de la última modificación etc.), pero también pueden ser

utilizados para propósitos específicos de una aplicación. Actualmente HPSF soporta sólo funcionalidad de lectura. Por favor, vea [la página del proyecto HPSF \[EN\]](#) para más información.

3. Contribuyendo

Así que ¿te gustaría contribuir al proyecto? ¡Genial! Necesitamos gente entusiasta, que trabaje duro, que tenga talento para ayudarnos con el proyecto en varias áreas. ¡La primera es petición de nuevas funciones y aviso de errores! La segunda es documentación - estaremos a tu entera disposición si tienes alguna crítica o te gustaría contribuir o mejorar de alguna forma la documentación. ¡Especialmente no nos vendría mal algo de ayuda en documentar el formato de fichero HSSF! ¡Por último, aunque no por ello menos importante, nos vendría bien algunos programadores Java que mastiquen binario, para que le echen el diente a la convolución que caracteriza los formatos de fichero de Microsoft y para que nos ayude a adaptar nuevos formatos a una plataforma Java superior!

¡Así que si estás motivado, listo, y tienes tiempo, únete a las listas de correo y estaremos encantados de ayudarte a empezar en el proyecto!

Copyright (c) @year@ The Apache Software Foundation All rights reserved.