

Apache POI - HWPf - Java API to Handle Microsoft Word Files

Overview

by Nicola Ken Barozzi, Andrew C. Oliver, Ryan Ackley, Rainer Klute

1. Overview

HWPf is the name of our port of the Microsoft Word 97(-2007) file format to pure Java. It also provides limited read only support for the older Word 6 and Word 95 file formats.

The partner to HWPf for the new Word 2007 .docx format is *XWPF*. Whilst HWPf and *XWPF* provide similar features, there is not a common interface across the two of them at this time.

HWPf is still in early development. It is in the [scratchpad section of the SVN](#). You will need to ensure you either have a recent SVN checkout, or a recent SVN nightly build (including the scratchpad jar!)

Source in the *org.apache.poi.hwpf.model* tree is the old legacy code refactored into an object model. Source code in the *org.apache.poi.hwpf.extractor* tree is a wrapper of this to facilitate easy extraction of interesting things (eg the Text). Source code in the *org.apache.poi.hdf* tree is the old legacy code.

1.1. XWPF Patches Required!

At the moment, *XWPF* covers many common use cases for reading and writing .docx files. Whilst this is a great thing, it does mean that *XWPF* does everything that the current POI committers need it to do, and so none of the committers are actively adding new features.

If you come across a feature in *XWPF* that you need, and isn't currently there, please do send in a patch to add the extra functionality! More details on contributing patches are available on the ["Contribution to POI" page](#).

1.2. HWPf Pointman Needed!

At the moment we unfortunately do not have someone taking care for HWPF and fostering its development. What we need is someone to stand up, take this thing under his hood as his baby and push it forward. Ryan Ackley, who put a lot of effort into HWPF, is no longer on board, so HWPF is an orphan child waiting to be adopted.

If **you** are interested in becoming the new HWPF pointman, you should look into the Microsoft Word internals. A good starting point seems to be Ryan Ackley's [overview](#). Full details on the word format is available from [Microsoft](#), but the documentation can be a little hard to get into at first... Try reading the [overview](#) first, and looking at the existing code, then finally look up the documentation for specific missing features.

As a first step you should familiarize yourself with the source code, examples, test cases, and the HWPF patches available at [Bugzilla](#) (if any). Then you should compile an overview of

- the current HWPF status,
- the patches in [Bugzilla](#) to be checked in (and those that should better be ditched),
- the available test cases and the test cases still to be written,
- the available documentation and the docs to be written,
- anything else that seems reasonable

When you start coding, you will not yet have write access to the SVN repository. Please submit your patches to [Bugzilla](#) and nag [the dev list](#) until someone commits them. Besides the actual checking in of HWPF patches, current POI committers will also do some minor reviews now and then of your source code patches, test cases and documentation to help ensure software quality. But most of the time you will be on your own. However, anyone offering useful contributions over a period of time will be offered committership!

Please do not forget to write [JUnit](#) test cases and documentation! We won't accept code that doesn't come with test cases. And please consider that other contributors should be able to understand your source code easily. If you need any help getting started with JUnit test cases for HWPF, please ask on the developers' mailing list! If you show that you are prepared to stick at it you will most likely be given SVN commit access. See "[Contribution to POI](#)" page for more details and help getting started.

Of course we will help you as best as we can. However, presently there is no committer who is really familiar with the Word format, so you'll be mostly on your own. We are looking forward for you and your contributions! Honor and glory of becoming a POI committer are waiting!