

Cours IUT Paris 8 – January 2012

An Introduction to using
Excel Files
With Java Apache POI

Arnaud Nauwynck

This document :

[http://arnaud.nauwynck.chez-alice.fr/
devPerso/Pres/Intro-Excel-Java-POI.pdf](http://arnaud.nauwynck.chez-alice.fr/devPerso/Pres/Intro-Excel-Java-POI.pdf)

Table Of Content

- Introduction to Excel
- Apache POI library
- Excel Document Model overview
 - Workbook, worksheet, row, col, cell, formula, ...
- Samples



Excel...

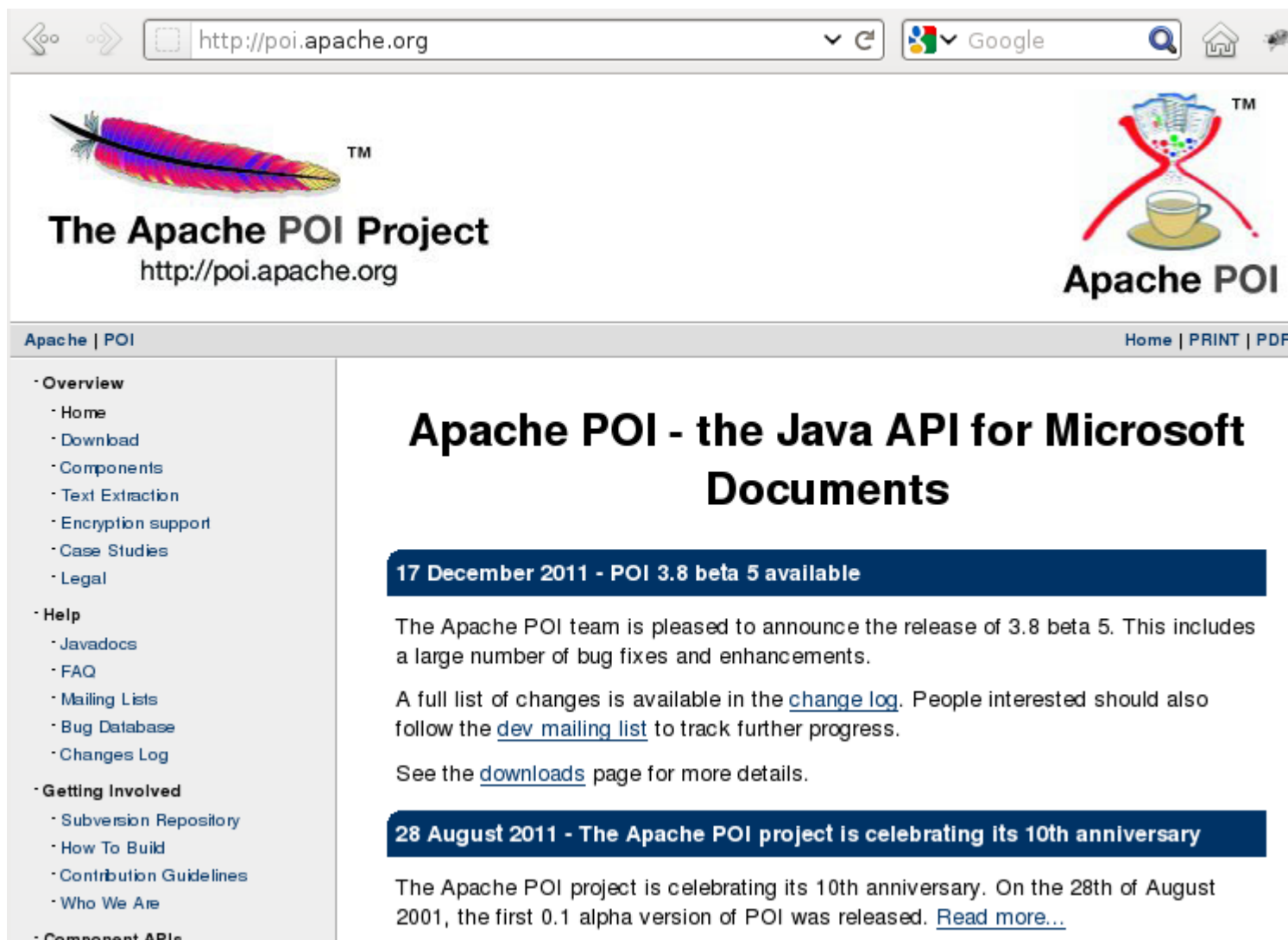


- Historic: “Multiplan” on CP/M, then dos, then windows, office...
- Main reason of Microsoft success, before Windows
- Excel is a “Table” editor / formula evaluator
- Extremely used in enterprises...
- Quick & Dirty for office / app development
- ... can be extended with Macros(VBA, ..), Add-ins

.xls, .xlsx, .xlsm, .xlsb, .xltm, .xla, .xll Microsoft way of Standards...

- Before 2007
 - Format : proprietary BINARY
- After 2007
 - Now zipped Xml (~Open Document) “.xlsx”
- File format is now open ... but not real standard
- Read it is “easy”... not interpreting VBA!

Http://poi.apache.org



The screenshot shows the Apache POI Project website. At the top, there's a browser address bar with 'http://poi.apache.org' and a Google search bar. Below the browser bar, the website header features the Apache feather logo on the left and the Apache POI logo on the right. The main content area has a left sidebar with navigation links and a main section with a title and two news items.

The Apache POI Project
<http://poi.apache.org>

Apache POI

Apache | POI [Home](#) | [PRINT](#) | [PDF](#)

- Overview
 - Home
 - Download
 - Components
 - Text Extraction
 - Encryption support
 - Case Studies
 - Legal
- Help
 - Javadocs
 - FAQ
 - Mailing Lists
 - Bug Database
 - Changes Log
- Getting Involved
 - Subversion Repository
 - How To Build
 - Contribution Guidelines
 - Who We Are
- Component APIs

Apache POI - the Java API for Microsoft Documents

17 December 2011 - POI 3.8 beta 5 available

The Apache POI team is pleased to announce the release of 3.8 beta 5. This includes a large number of bug fixes and enhancements.

A full list of changes is available in the [change log](#). People interested should also follow the [dev mailing list](#) to track further progress.

See the [downloads](#) page for more details.

28 August 2011 - The Apache POI project is celebrating its 10th anniversary

The Apache POI project is celebrating its 10th anniversary. On the 28th of August 2001, the first 0.1 alpha version of POI was released. [Read more...](#)

POI ... not only Excel

- Component APIs

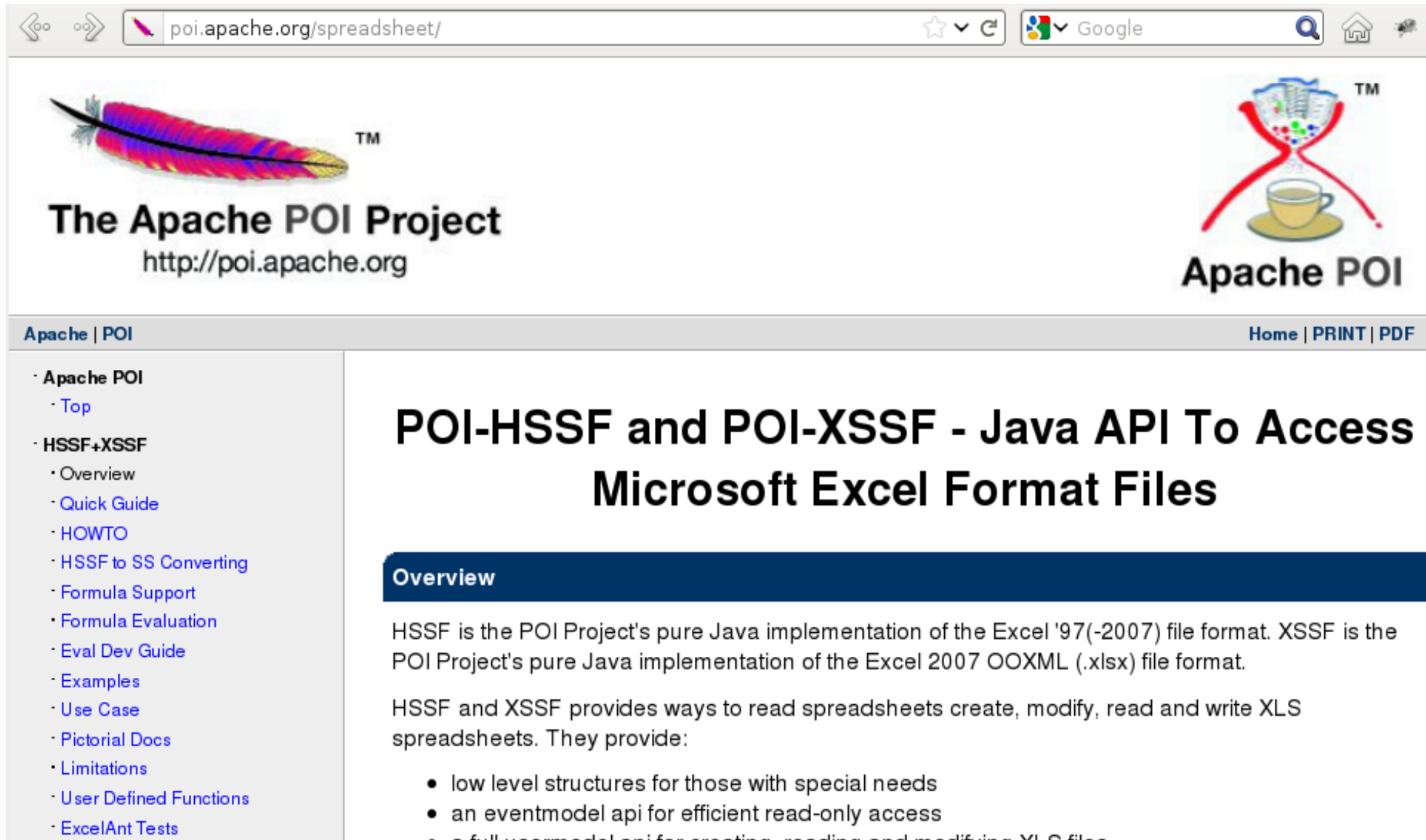
- Excel (SS=HSSF+XSSF)
- Word (HWPf+XWPF)
- PowerPoint (HSLF+XSLF)
- OpenXML4J (OOXML)
- OLE2 Filesystem (POIFS)
- OLE2 Document Props (HPSF)
- Outlook (HSMF)
- Visio (HDGF)
- TNEF (HMEF)
- Publisher (HPBF)

- HSSF+XSSF

- Overview
- Quick Guide
- HOWTO
- HSSF to SS Converting
- Formula Support
- Formula Evaluation
- Eval Dev Guide
- Examples
- Use Case
- Pictorial Docs
- Limitations
- User Defined Functions
- ExcelAnt Tests

- In POI... Excel = HSSF / XSSF :
Horrible **S**pread **S**heet **F**ormat (<2007)
Xml **S**pread **S**heet **F**ormat (>= 2007)

Http://poi.apache.org/spreadsheet



The screenshot shows a web browser window with the address bar displaying "poi.apache.org/spreadsheet/". The page features the Apache POI Project logo on the left, which includes a colorful feather and the text "The Apache POI Project http://poi.apache.org". On the right, there is a logo for "Apache POI" featuring a red ribbon forming an 'X' over a yellow cup. The main content area has a dark blue header with the text "POI-HSSF and POI-XSSF - Java API To Access Microsoft Excel Format Files". Below this, a section titled "Overview" explains that HSSF is the pure Java implementation of the Excel '97(-2007) file format and XSSF is the pure Java implementation of the Excel 2007 OOXML (.xlsx) file format. It also states that HSSF and XSSF provide ways to read, create, modify, and write XLS spreadsheets. A list of features follows: low level structures for special needs, an eventmodel api for efficient read-only access, and a full usermodel api for creating, reading, and modifying XLS files. A left sidebar contains a navigation menu with links to "Apache POI", "Top", "HSSF+XSSF", "Overview", "Quick Guide", "HOWTO", "HSSF to SS Converting", "Formula Support", "Formula Evaluation", "Eval Dev Guide", "Examples", "Use Case", "Pictorial Docs", "Limitations", "User Defined Functions", and "ExcelAnt Tests". The top right of the page has links for "Home", "PRINT", and "PDF".

poi.apache.org/spreadsheet/

Google

 **The Apache POI Project**
http://poi.apache.org

 **Apache POI**

Apache | POI

Home | PRINT | PDF

POI-HSSF and POI-XSSF - Java API To Access Microsoft Excel Format Files

Overview

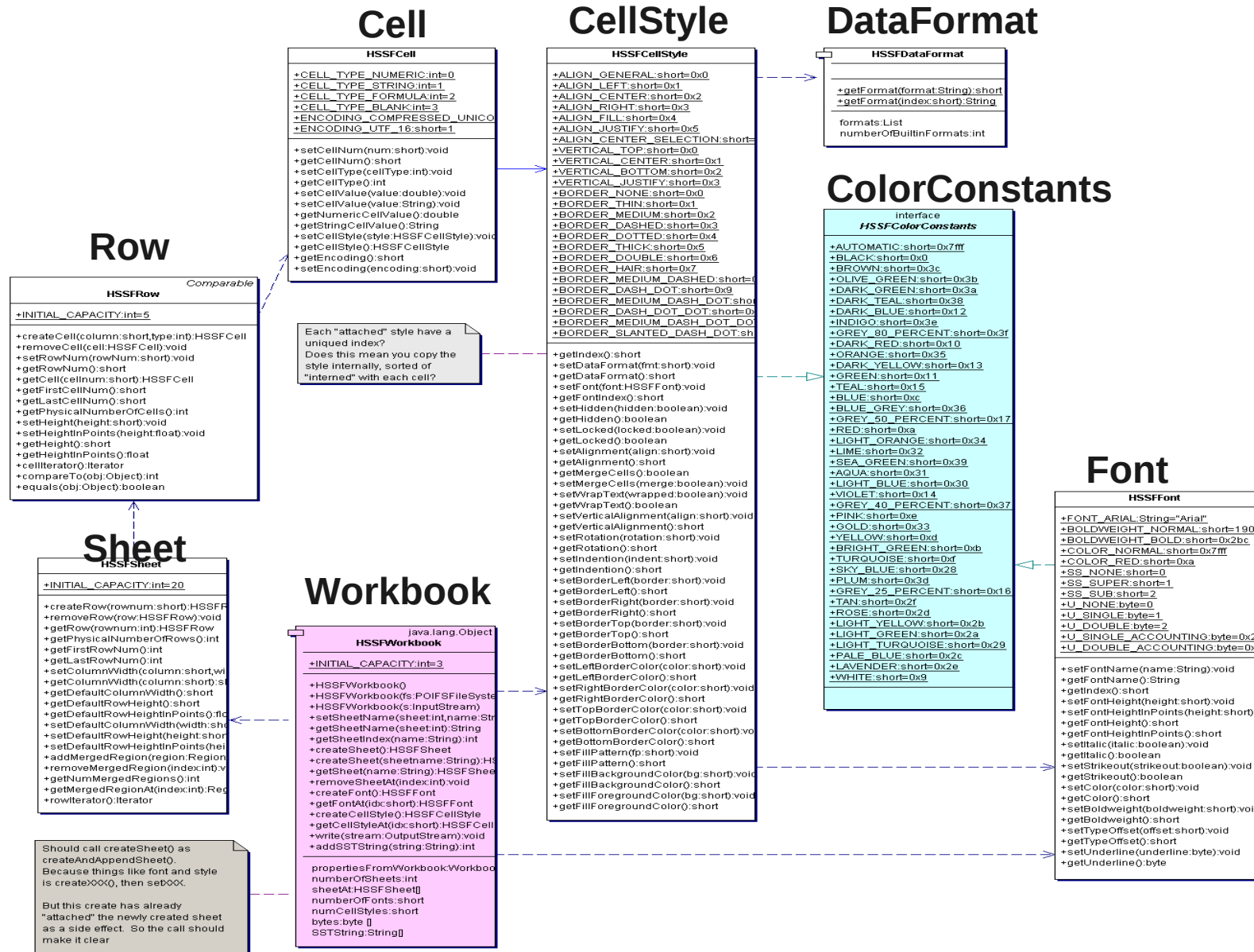
HSSF is the POI Project's pure Java implementation of the Excel '97(-2007) file format. XSSF is the POI Project's pure Java implementation of the Excel 2007 OOXML (.xlsx) file format.

HSSF and XSSF provides ways to read spreadsheets create, modify, read and write XLS spreadsheets. They provide:

- low level structures for those with special needs
- an eventmodel api for efficient read-only access
- a full usermodel api for creating, reading and modifying XLS files

- Apache POI
- Top
- HSSF+XSSF
 - Overview
 - Quick Guide
 - HOWTO
 - HSSF to SS Converting
 - Formula Support
 - Formula Evaluation
 - Eval Dev Guide
 - Examples
 - Use Case
 - Pictorial Docs
 - Limitations
 - User Defined Functions
 - ExcelAnt Tests

<http://poi.apache.org/spreadsheet/diagram1.html>



POI in Java Maven Project

Search jar version in Nexus

<https://oss.sonatype.org/index.html#nexus-search>

Search 			
GAV Search ▾	Group: org.apache.poi	Artifact:	Version:
Group	Artifact	Version ▲	Download
org.apache.poi	poi-scratchpad	3.8-beta4	pom , jar , sources.jar
org.apache.poi	poi-ooxml-schemas	3.8-beta4	pom , jar
org.apache.poi	poi-ooxml	3.8-beta4	pom , jar , sources.jar
org.apache.poi	poi	3.8-beta4	pom , jar , sources.jar

```
<dependency>
  <groupId>org.apache.poi</groupId>
  <artifactId>poi-ooxml</artifactId>
  <version>3.8-beta4</version>
</dependency>
```

Typical Usages

- Web Reporting generated on server side
 - Export: on Click => servlet => query database
 - => load template xls from classloader resource (file)
 - => fill data (add rows, set cell values)
 - => write to output (file)
- Web Import (bulk update)
- Periodic Batch tool
 - Quartz Sheduler => generate => send email/ write file / ftp
- Client Standalone tool
 - Swing export from Jtable
 - Import + Export tool, compare

POI Limitations

- Import + Re-Export is 99.9% OK
... (unimplemented things are preserved)
- Formula evaluation
 - ~80% of built-in formula are translated in Java
 - NOT VBA addins !!
 - POI is NOT an excel gui ... use Real Excel/OpenOffice
- Pivot Tables
- Graphs, Forms

Read & Download Samples

- Many resources ... Monkey See, Monkey Do
- Start with Doc & samples at <http://poi.apache.org>
 - .../spreadsheet/quick-guide.html
 - .../spreadsheet/examples.html
- Then checkout Code (Source + sample + doc)
- Then .. google

[http://poi.apache.org/ spreadsheet/quick-guide.html](http://poi.apache.org/spreadsheet/quick-guide.html)

Index of Features

- [How to create a new workbook](#)
- [How to create a sheet](#)
- [How to create cells](#)
- [How to create date cells](#)
- [Working with different types of cells](#)
- [Iterate over rows and cells](#)
- [Getting the cell contents](#)
- [Text Extraction](#)
- [Aligning cells](#)
- [Working with borders](#)
- [Fills and color](#)
- [Merging cells](#)
- [Working with fonts](#)
- [Custom colors](#)
- [Reading and writing](#)
- [Use newlines in cells.](#)
- [Create user defined data formats](#)
- [Fit Sheet to One Page](#)
- [Set print area for a sheet](#)
- [Set page numbers on the footer of a sheet](#)
- [Shift rows](#)
- [Set a sheet as selected](#)
- [Set the zoom magnification for a sheet](#)
- [Create split and freeze panes](#)
- [Repeating rows and columns](#)
- [Headers and Footers](#)
- [Drawing Shapes](#)
- [Styling Shapes](#)
- [Shapes and Graphics2d](#)
- [Outlining](#)
- [Images](#)
- [Named Ranges and Named Cells](#)
- [How to set cell comments](#)
- [How to adjust column width to fit the contents](#)
- [Hyperlinks](#)
- [Data Validation](#)
- [Embedded Objects](#)
- [Autofilters](#)
- [Conditional Formatting](#)

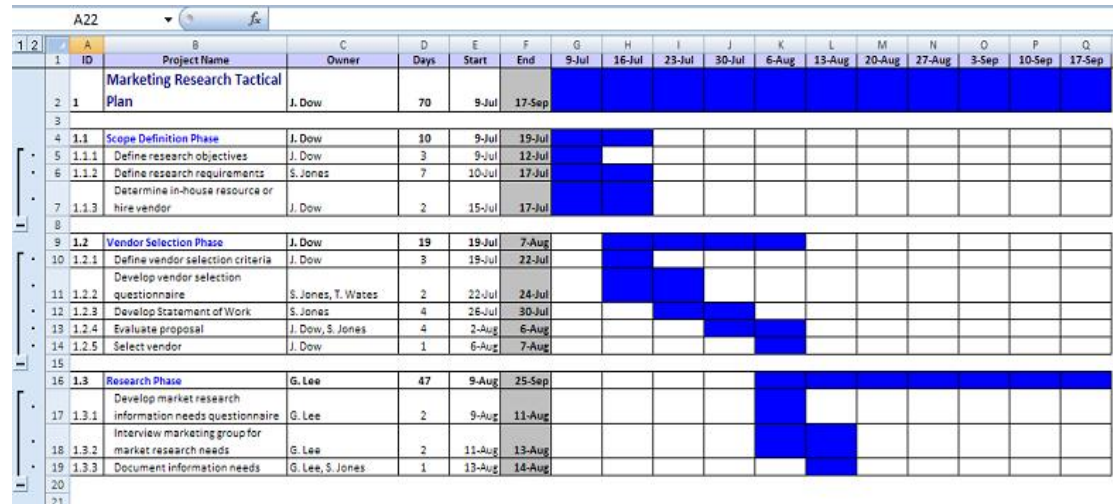
[http://poi.apache.org/ spreadsheet/quick-guide.html](http://poi.apache.org/spreadsheet/quick-guide.html) #ReadWriteWorkbook

Reading and Rewriting Workbooks

```
InputStream inp = new FileInputStream("workbook.xls");  
//InputStream inp = new FileInputStream("workbook.xlsx");  
  
Workbook wb = WorkbookFactory.create(inp);  
Sheet sheet = wb.getSheetAt(0);  
Row row = sheet.getRow(2);  
Cell cell = row.getCell(3);  
if (cell == null)  
    cell = row.createCell(3);  
cell.setCellType(Cell.CELL_TYPE_STRING);  
cell.setCellValue("a test");  
  
// Write the output to a file  
FileOutputStream fileOut = new FileOutputStream("workbook.xls");  
wb.write(fileOut);  
fileOut.close();
```

<http://poi.apache.org/spreadsheet/examples.html>

- Extract
- 1/4 BusinessPlan
1 file ... 300 lines
- 2/4 CalendarDemo
1 file ... 200 lines



A screenshot of a spreadsheet application showing a project plan. The active cell is A22. The spreadsheet has columns for dates from 9-Jul to 17-Sep and rows for project tasks. The 'Marketing Research Tactical Plan' is the main project, with sub-tasks like 'Scope Definition Phase', 'Vendor Selection Phase', and 'Research Phase'. Each task is assigned to an owner and has a duration in days. The dates in the header row are: 9-Jul, 16-Jul, 23-Jul, 30-Jul, 6-Aug, 13-Aug, 20-Aug, 27-Aug, 3-Sep, 10-Sep, 17-Sep.

ID	Project Name	Owner	Days	Start	End	9-Jul	16-Jul	23-Jul	30-Jul	6-Aug	13-Aug	20-Aug	27-Aug	3-Sep	10-Sep	17-Sep
1	Marketing Research Tactical Plan	J. Dow	70	9-Jul	17-Sep											
1.1	Scope Definition Phase	J. Dow	10	9-Jul	19-Jul											
1.1.1	Define research objectives	J. Dow	3	9-Jul	12-Jul											
1.1.2	Define research requirements	S. Jones	7	10-Jul	17-Jul											
1.1.3	Determine in-house resource or hire vendor	J. Dow	2	15-Jul	17-Jul											
1.2	Vendor Selection Phase	J. Dow	19	19-Jul	7-Aug											
1.2.1	Define vendor selection criteria	J. Dow	3	19-Jul	22-Jul											
1.2.2	Develop vendor selection questionnaire	S. Jones, T. Wates	2	22-Jul	24-Jul											
1.2.3	Develop Statement of Work	S. Jones	4	26-Jul	30-Jul											
1.2.4	Evaluate proposal	J. Dow, S. Jones	4	2-Aug	6-Aug											
1.2.5	Select vendor	J. Dow	1	6-Aug	7-Aug											
1.3	Research Phase	G. Lee	47	9-Aug	25-Sep											
1.3.1	Develop market research information needs questionnaire	G. Lee	2	9-Aug	11-Aug											
1.3.2	Interview marketing group for market research needs	G. Lee	2	11-Aug	13-Aug											
1.3.3	Document information needs	G. Lee, S. Jones	1	13-Aug	14-Aug											



A screenshot of a spreadsheet application showing a calendar for January 2009. The active cell is A1. The spreadsheet has columns for days of the week (Sunday to Saturday) and rows for dates (1 to 31). The dates are: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29, 30, 31.

	Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
1					1	2	3
4		5	6	7	8	9	10
11		12	13	14	15	16	17
18		19	20	21	22	23	24
25		26	27	28	29	30	31

Checkout Code & Sample

```
$ git clone git://git.apache.org/poi.git
Cloning into poi...
^C
$ ls
build build.xml doap_POI.rdf forrest.properties KEYS legal li
$
$ find . -type d -name examples
./src/examples
./src/examples/src/org/apache/poi/hslf/examples
./src/examples/src/org/apache/poi/hsmf/examples
./src/examples/src/org/apache/poi/ss/examples
./src/examples/src/org/apache/poi/hssf/usermodel/examples
./src/examples/src/org/apache/poi/hssf/eventusermodel/examples
./src/examples/src/org/apache/poi/hpsf/examples
./src/examples/src/org/apache/poi/xssf/usermodel/examples
./src/examples/src/org/apache/poi/xssf/eventusermodel/examples
./src/excelant/testcases/org/apache/poi/ss/examples
```

Questions ?

Arnaud.nauwynck@gmail.com

Alors TP !

This document:

[http://arnaud.nauwynck.chez-alice.fr/
devPerso/Pres/Intro-Excel-Java-POI.pdf](http://arnaud.nauwynck.chez-alice.fr/devPerso/Pres/Intro-Excel-Java-POI.pdf)