How to get co-ordinates or location and size of images in PDF using PDFBox

Apache PDFBox Tutorial – We shall learn how to get co-ordinates or location and size of images in pdf from all the pages using PDFStreamEngine.

The class org.apache.pdfbox.contentstream.**PDFStreamEngine** handles and executes some of the operations in processing a PDF document by providing a callback interface.

To get co-ordinates or location and size of images in pdf, we shall extend this PDFStreamEngine class, intercept and implement **processOperator(Operator operator, List<COSBase> operands)** method.

COSBase is the base class that all objects in the PDF document will extend.

For each object in the PDF document, the above mentioned method processOperator() is called in PDFStreamEngine.processPage(page). For each of the object in PDF document, we shall check if the object is an image object and get its properties like (X,Y) co-ordinates and size.

Get co-ordinates or location and size of images in pdf

Following is a step by step process to get co-ordinates or location and size of images in pdf :

1. Extend PDFStreamEngine

Create a Java Class and extend it with PDFStreamEngine.

Extend PDFStreamEngine

```
public class
```

```
public class GetImageLocationsAndSize extends PDFStreamEngine
```

2. Call processPage()

For each of the pages in PDF document, call the method processPage(page).

Call processPage() for each page in the PDF

```
for( PDPage page :
```

```
for( PDPage page : document.getPages() )
{
    pageNum++;
    printer.processPage(page);
}
```

3. Override processOperator()

For each of the object in PDF page, processOperator is called in processPage(). We shall override processOperator().

Override processOperator()

```
@Override
protected void
```

```
    @Override
protected void processOperator( Operator operator, List<COSBase> operands) throws IOException{
    . . .
}
```

4. Check for Image

Check if the object that has been sent to processOperator() is an image object.

Check for Image

```
if( xobject instanceof
PDImageXObject){
```

```
if( xobject instanceof PDImageXObject){
    . . .
}
```

5. Print Locations and Size

If the object is an image object, print the locations and size of the image.

Example Java Program to get location and size of images in pdf

Get co-ordinates or location and size of images in pdf

```
import
org.apache.pdfbox.cos.
```

```
import org.apache.pdfbox.cos.COSBase;
import org.apache.pdfbox.cos.COSName;
```

```
import org.apache.pdfbox.cos.COSName;
import org.apache.pdfbox.pdmodel.PDDocument;
import org.apache.pdfbox.pdmodel.PDPage;
import org.apache.pdfbox.pdmodel.graphics.PDXObject;
import org.apache.pdfbox.pdmodel.graphics.form.PDFormXObject;
import org.apache.pdfbox.pdmodel.graphics.image.PDImageXObject;
import org.apache.pdfbox.util.Matrix;
import org.apache.pdfbox.contentstream.operator.DrawObject;
import org.apache.pdfbox.contentstream.operator.Operator;
import org.apache.pdfbox.contentstream.PDFStreamEngine;

import java.io.File;
import java.io.IOException;
import java.util.List;

import org.apache.pdfbox.contentstream.operator.state.Concatenate;
import org.apache.pdfbox.contentstream.operator.state.Restore;
import org.apache.pdfbox.contentstream.operator.state.Save;
import org.apache.pdfbox.contentstream.operator.state.SetGraphicsStateParameters;
import org.apache.pdfbox.contentstream.operator.state.SetMatrix;

/**
* This is an example on how to get the x/y coordinates of image location and size of image.
*/
public class GetImageLocationsAndSize extends PDFStreamEngine
{
    /**
     * @throws IOException If there is an error loading text stripper properties.
     */
    public GetImageLocationsAndSize() throws IOException
    {
        // preparing PDFStreamEngine
        addOperator(new Concatenate());
        addOperator(new DrawObject());
        addOperator(new SetGraphicsStateParameters());
        addOperator(new Save());
        addOperator(new Restore());
        addOperator(new SetMatrix());
    }

    /**
     * @throws IOException If there is an error parsing the document.
     */
    public static void main( String[] args ) throws IOException
    {
        PDDocument document = null;
        String fileName = "apache.pdf";
        try
        {
```

```
        document = PDDocument.load( new File(fileName) );
        GetImageLocationsAndSize printer = new GetImageLocationsAndSize();
        int pageNum = 0;
        for( PDPage page : document.getPages() )
        {
            pageNum++;
            System.out.println( "\n\nProcessing page: " + pageNum +"\n--------------------------------");
            printer.processPage(page);
        }
    }
    finally
    {
        if( document != null )
        {
            document.close();
        }
    }
}

/**
 * @param operator The operation to perform.
 * @param operands The list of arguments.
 *
 * @throws IOException If there is an error processing the operation.
 */
@Override
protected void processOperator( Operator operator, List<COSBase> operands) throws IOException
{
    String operation = operator.getName();
    if( "Do".equals(operation) )
    {
        COSName objectName = (COSName) operands.get( 0 );
        // get the PDF object
        PDXObject xobject = getResources().getXObject( objectName );
        // check if the object is an image object
        if( xobject instanceof PDImageXObject)
        {
            PDImageXObject image = (PDImageXObject)xobject;
            int imageWidth = image.getWidth();
            int imageHeight = image.getHeight();

            System.out.println("\nImage [" + objectName.getName() + "]");

            Matrix ctmNew = getGraphicsState().getCurrentTransformationMatrix();
            float imageXScale = ctmNew.getScalingFactorX();
            float imageYScale = ctmNew.getScalingFactorY();

            // position of image in the pdf in terms of user space units
            System.out.println("position in PDF = " + ctmNew.getTranslateX() + " " + ctmNew.getTranslateY() + " in user
```

```
            System.out.println("position in PDF = " + ctmNew.getTranslateX() + ", " + ctmNew.getTranslateY() + " in user
space units");
            // raw size in pixels
            System.out.println("raw image size  = " + imageWidth + ", " + imageHeight + " in pixels");
            // displayed size in user space units
            System.out.println("displayed size  = " + imageXScale + ", " + imageYScale + " in user space units");
         }
         else if(xobject instanceof PDFormXObject)
         {
            PDFormXObject form = (PDFormXObject)xobject;
            showForm(form);
         }
      }
      else
      {
         super.processOperator( operator, operands);
      }
   }

}
```

Output

```
Processing page: 1
---------------------
```

```
Processing page: 1
---------------------------------

Image [X0]
position in PDF = 36.506977, 695.3907 in user space units
raw image size = 429, 175 in pixels
displayed size = 214.69952, 87.58139 in user space units

Image [X1]
position in PDF = 36.506977, 617.8186 in user space units
raw image size = 300, 300 in pixels
displayed size = 75.06976, 75.06976 in user space units

Image [X2]
position in PDF = 36.506977, 138.37305 in user space units
raw image size = 600, 383 in pixels
displayed size = 496.96182, 317.29486 in user space units


Processing page: 2
---------------------------------

Image [X0]
position in PDF = 36.506977, 495.70514 in user space units
raw image size = 600, 383 in pixels
displayed size = 496.96182, 317.29486 in user space units

Image [X1]
position in PDF = 245.20093, 307.53027 in user space units
raw image size = 212, 146 in pixels
displayed size = 106.0986, 73.0679 in user space units


Processing page: 3
---------------------------------


Processing page: 4
---------------------------------
```

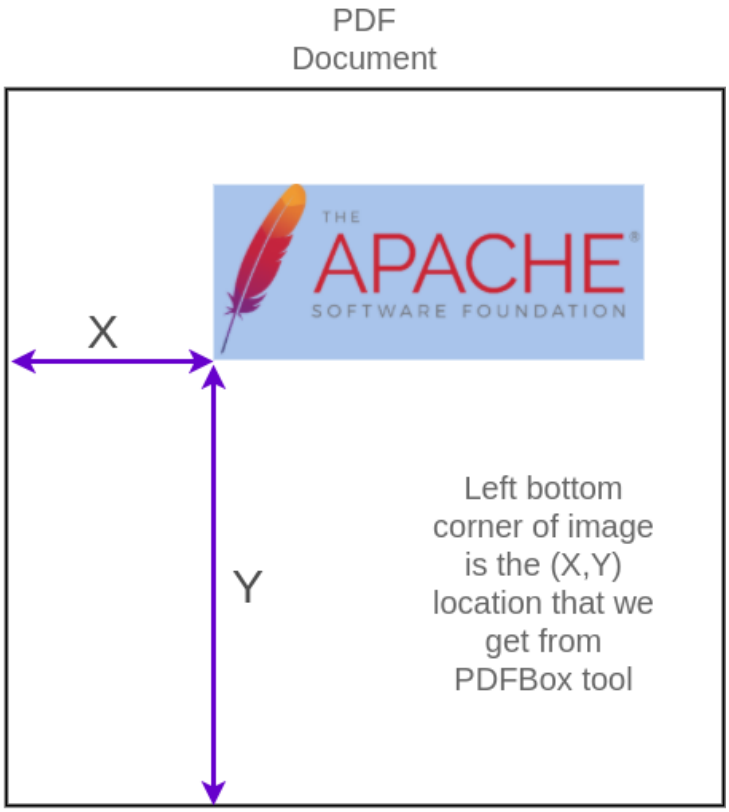
Download the pdf document here apache.pdf if you would like use the same PDF file. Else you may assign the fileName in the Java program with your PDF file path.

Raw Size vs Displayed Size

The size of image displayed in the pdf could be different from the actual size of original (or raw) image.

(X,Y) location of image in PDF

Left bottom corner of image is the (X,Y) location that we get from PDFBox tool.



(X,Y) location of image in PDF

Conclusion :

In this Apache PDFBox Tutorial we have learnt to get co-ordinates or location and size of images in pdf document and also learnt what x and y coordinates mean for an image in a pdf.

PDFBox

- PDFBox Tutorial
- Setup Java Project with PDFBox

Text Processing

- Create a PDF file with Text
- Read all the text from PDF
- Extract coordinates or position of characters in PDF
- Extract words from PDF
- Read text line by line from PDF
- PDFBox - Split PDF Document
- PDFBox - Merge multiple PDFs

Image Processing

- Get Location and Size of Images
- Extract Images from PDF