



# JPdfUnit - framework for testing pdf documents



Benjamin Bratkus, [bratkus@users.sourceforge.net](mailto:bratkus@users.sourceforge.net)

Tobias Kieninger, [xynoo@users.sourceforge.net](mailto:xynoo@users.sourceforge.net)

**Orientation in Objects GmbH**

Weinheimer Str. 68  
68309 Mannheim

[www.oio.de](http://www.oio.de)  
[info@oio.de](mailto:info@oio.de)

Version: 1.0

# Agenda



- Overview
- Usage Scenarios
- Resources

# Agenda



- **Overview**
- Usage Scenarios
- Resources

- Framework for testing pdf documents
- Ready-to-use assertions for JUnit tests
- High level api to access PDFs with Java
- Lot of possibilities in pdf document handling
  - Access meta data
  - Read / search fragments of content

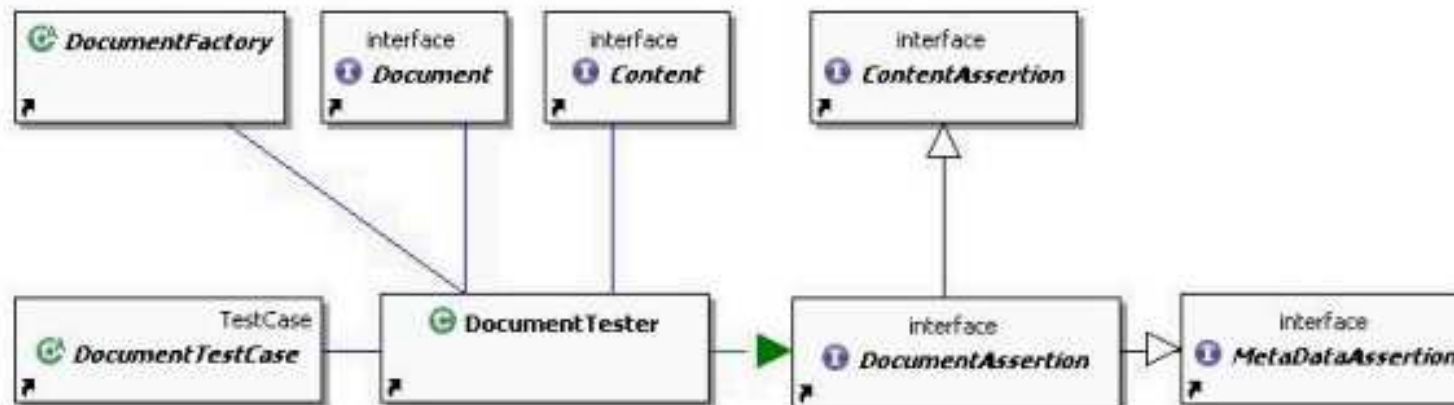
## Dependencies

- Extends the JUnit testing Framework
- Adapts PdfBox, a PDF API
- Has to use Log4J because of other dependencies
- Is hosted on Sourceforge.net



## Extends the JUnit testing Framework

- DocumentTestCase class extends TestCase
  - Transparent document handling without setUp() and tearDown()
  - Provides a template method onSetup() for user specific setup
  - Easy to use assertions
- DocumentTester uses Assertions to check values of the document

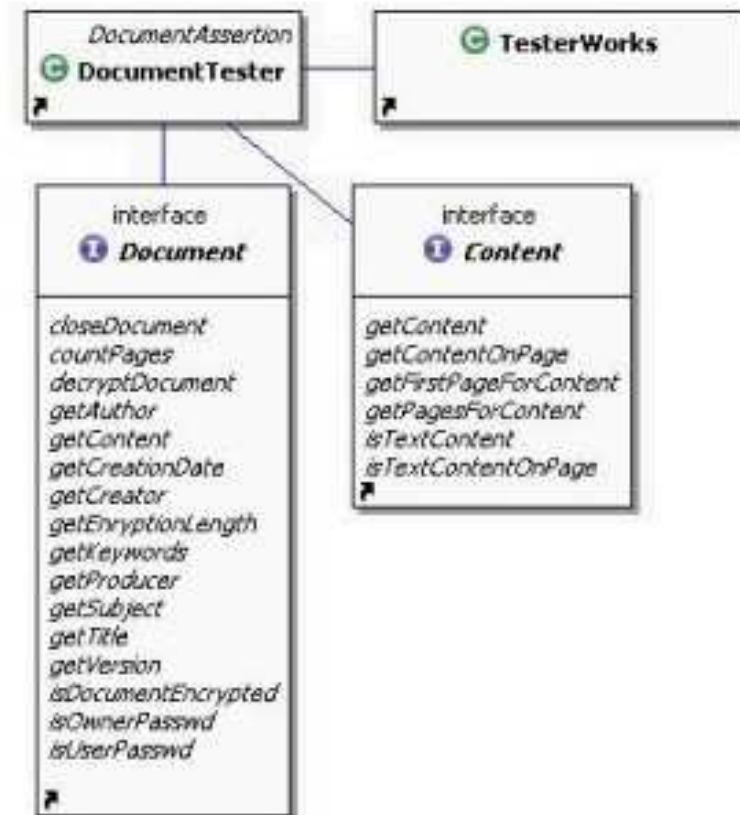


## Adapts PdfBox



Orientation in Objects

- PdfBoxAnalyser extends TextStripper
  - Extension needed for special features
- DocumentTester provides access
  - Document
    - **information about the document**
  - Content
    - **content related**



# Agenda



- Overview
- **Usage Scenarios**
- Resources



- Extend the class DocumentTestCase
  - Easiest way to use
  - Document handling transparent
- Just four steps to go
  - Write a test class with the JUnit conventions
  - Extend DocumentTestCase
  - Implement the method getDataSource() in your test class
  - Use the assertXY() methods of the DocumentTestCase

## Usage Scenarios I example

### Extend the DocumentTestCase class



```
public class OioTest extends DocumentTestCase {
    public OioTest(String name) {
        super(name);
    }
    protected DocumentDataSource getDataSource() {
        DocumentDataSource ds = new
                                PdfDataSource("MyPdf.pdf");
        return datasource;
    }
    public void testDocumentEncryption() {
        this.assertIsDocumentEncrypted(true);
    }
}
```

- Do not Extend the DocumentTestCase class
  - Same functionality accessible to the user
  - Useful for inheriting another testing framework, i.e. JWebUnit
- Little more work to do
  - Write a test class with the JUnit conventions
  - Inherit the other TestCase class
  - Use setup() to instantiate your DocumentTester
  - Use tearDown() to close the DocumentTester

## Usage Scenarios II example

Do not Extend the DocumentTestCase class

```
public class DocumentEncryptionTest extends TestCase {  
    DocumentTester tester;  
    public DocumentEncryptionTest(String name) {  
        super(name);  
    }  
    protected void setUp() throws Exception {  
        tester = new DocumentTester("MyPdf.pdf");  
    }  
    protected void tearDown() throws Exception {  
        tester.close();  
    }  
    public void testDocumentEncryption() {  
        tester.assertIsDocumentEncrypted(true);  
    }  
}
```

- Use the DocumentTester in an application
- Abstract document handling via high level api
  - Work with the Interfaces Document and Content
- Meta information and content accessible
- No dependencies to JUnit left
- Do not use the assertXY() methods of the DocumentTester
- Be good and close the document at the end

## Usage Scenarios III example

### Use the DocumentTester in a application

```
public class TesterWorks {  
  
    private static DocumentTester myTester =  
        new DocumentTester("MyPdf.pdf");  
  
    public static void main (String []args) throws  
        Exception {  
        Document doc = myTester.getDocument();  
        System.out.println("Page Count ?");  
        System.out.println(doc.countPages());  
        myTester.close();  
    }  
}
```

# Agenda



- Overview
- Usage Scenarios
- **Resources**

- Junit
  - <http://www.junit.org>
- PDFBox
  - <http://www.pdfbox.org>
- Log4j
  - <http://logging.apache.org/log4j>
- JPdfUnit
  - <http://jpdfunit.sourceforge.net>





**Thank you for your  
attention.  
End.**

**Orientation in Objects GmbH**

Weinheimer Str. 68  
68309 Mannheim

[www.oio.de](http://www.oio.de)  
[info@oio.de](mailto:info@oio.de)

Version: 1.0



**Questions ?**

**Orientation in Objects GmbH**

Weinheimer Str. 68  
68309 Mannheim

[www.oio.de](http://www.oio.de)  
[info@oio.de](mailto:info@oio.de)