

Release Notes

XMLMill v 1.40

1.0

xmlmill

© 2000-2003 Pecunia Data Systems bvba. All rights reserved.

NOTICE: All information contained herein is the property of Pecunia Data Systems bvba.

This publication and the information herein are furnished AS IS, are subject to change without notice, and should not be construed as a commitment by Pecunia Data Systems bvba. Pecunia Data Systems bvba assumes no responsibility or liability for any errors or inaccuracies, makes no warranty of any kind (express, implied, or statutory) with respect to this publication, and expressly disclaims any and all warranties of merchantability, fitness for particular purposes, and noninfringement of third-party rights. (First printing, May 2003)

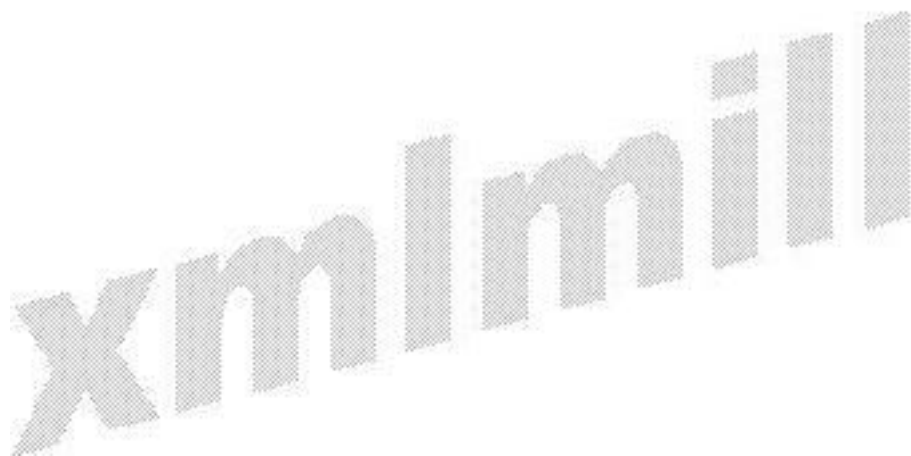
Table of Contents

1. Preface	6
2. New tags	7
2.1. <encryption>	7
2.1.1. Attribute: user-password	8
2.1.2. Attribute: owner-paswword	8
2.1.3. Attribute: allow-assembly	8
2.1.4. Attribute: allow-degraded-printing	8
2.1.5. Attribute: allow-fill-in	8
2.1.6. Attribute: allow-copy	8
2.1.7. Attribute: allow-modify-annotations	9
2.1.8. Attribute: allow-modify-contents	9
2.1.9. Attribute: allow-printing	9
2.1.10. Attribute: allow-screen-readers	9
2.1.11. Attribute: allow-screen-readers	9
3. Existing tags	11
3.1. <header>	11
3.1.1. Attribute: margin-top	11
3.1.2. Attribute: show-on-first-page	11
3.2. <footer>	11
3.2.1. Attribute: margin-bottom	11
3.2.2. Attribute: show-on-first-page	12
3.3. <field>	12
3.3.1. Attribute: rowspan	12
3.3.2. Attribute: width	12
3.4. <file>	13
3.5. <date>	13
3.6. <time>	13
3.7. <datetime>	14
3.7.1. Attribute: locale	14
3.7.2. Usage in the <p> and <field> tag	14
3.8. <page-number>	14
3.9. <page-total>	15
3.10. <table>	15
3.10.1. Attribute: type	15
3.10.2. Simple table type	15
3.10.3. Complex table type	16
4. Existing attributes	18
4.1. Attribute: color; background-color; border-color; border-top-color; border-right-color; border-bottom-color; border-left-color	18
5. Java API	23
5.1. Class: PDXTransform	23
5.1.1. Method: setFeature(String, boolean)	23
5.1.2. Method: boolean getFeature(String)	23
5.1.3. Method: void printComments(String)	23
5.1.4. Method: void setXSLTTransformationOnly(boolean b)	23
5.2. Class: PDELogFile	24



6. Known bugs.25

7. Disclaimer. 26



xmlmill

1. Preface

The XMLMill application is intended for software developers or end-users who want to generate .pdf documents from xml and/or xsl.

This release notes document describes all new (or enhanced) functionalities of version 1.40 of XMLMill.

For an overview of how to use XMLMill please consult in the **docs/** directory:.

- ◆ **apidoc/** -- The JavaDoc concerning the **PDX** api.
- ◆ **userguide.pdf** -- XMLMill user's guide.
- ◆ **dtdguide.pdf** -- An explanation of XMLMill's tag and their attributes.

If you have questions, please do not hesitate to send a mail to support@xmlmill.com.

☞ *This document is completely generated with XMLMill 1.40 using **rnotes140.xml** and **rnotes.xsl**. These files can be found in the directory **samples/docs/rnotes** in the download.*

xmlmill

2. New tags

This section chapter gives an overview of all new tags in this version.

2.1. <encryption>

Description:

Adding the <encryption> tag indicates that the document should be encrypted.

This version of XMLMill can be used to encrypt PDF documents as they are created. XMLMill uses the Acrobat's standard encryption handler which uses symmetric encryption.

If a file has a user or owner password or any permissions restrictions set, it will be encrypted.

For more information concerning how to use owner password and user password, please consult Adobe's PDF Reference manual.

The length of the encryption keys used for protecting documents depends on the PDF compatibility level chosen:

- ◆ For PDF versions up to and including 1.3 (i.e., Acrobat 4) the key length is 40 bits
- ◆ For PDF version 1.4 the key length is 128 bits (this requires Acrobat 5).

Features:

- ◆ Encrypts whole file (text and images).
- ◆ Uses 40-bit or 128-bit encryption.
- ◆ Password protection of a document.
- ◆ Control what a user can do with an opened file (using permissions).
- ◆ Uses Adobe's security.
- ◆ No extra downloads or plug-in required for recipient.

Appears in:

<document> tag

Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE documents SYSTEM "xmlmill.dtd">
<documents whitespace-collapse="on">
  <document file="output\encryption1.pdf">
    <encryption user-password="xmlmill"
      owner-password="xmlmill140"
      allow-assembly="on"
      allow-degraded-printing="on"
      allow-fill-in="on"
      allow-copy="on"
      allow-modify-annotations="on">
```

```
allow-modify-contents="on"  
allow-printing="on"  
allow-screen-readers="on"  
encryption-strength="128" />  
...
```

2.1.1. Attribute: **user-password**

The **user-password** attribute defines the user password of the document. The password can have any length, although only the first 32 characters will be used to calculate the encryption key.

2.1.2. Attribute: **owner-paswword**

The **owner-password** attribute defines the owner password of the document. The password can have any length, although only the first 32 characters will be used to calculate the encryption key.

2.1.3. Attribute: **allow-assembly**

Acrobat will allow inserting, deleting, or rotating pages and creating bookmarks and thumbnails, even if **allow-modify** hasn't been specified. This attribute can only be used if 128- bit encryption has been enabled.

Values

- ◆ **on**
- ◆ **of**

2.1.4. Attribute: **allow-degraded-printing**

Acrobat will allow high-resolution printing. If allow-printing hasn't been specified printing is restricted to the »print as image« feature which prints a low-resolution rendition of the page. This attribute can only be used if 128- bit encryption has been enabled.

Values

- ◆ **on**
- ◆ **of**

2.1.5. Attribute: **allow-fill-in**

Acrobat will allow form field filling, even if **allow-modify-annotations** hasn't been specified. This attribute can only be used if 128- bit encryption has been enabled.

Values

- ◆ **on**
- ◆ **of**

2.1.6. Attribute: **allow-copy**

Acrobat allows copying and extracting text or graphics, and will enable the accessibility interface

Values

- ◆ **on**
- ◆ **of**

2.1.7. Attribute: **allow-modify-annotations**

Acrobat allows adding or changing comments or form fields.

Values

- ◆ **on**
- ◆ **of**

2.1.8. Attribute: **allow-modify-contents**

Acrobat allows users adding form fields or making other changes.

Values

- ◆ **on**
- ◆ **of**

2.1.9. Attribute: **allow-printing**

Acrobat allows printing the file.

Values

- ◆ **on**
- ◆ **of**

2.1.10. Attribute: **allow-screen-readers**

Acrobat allows extracting text or graphics for accessibility purposes (such as a screenreader program).

Values

- ◆ **on**
- ◆ **of**

2.1.11. Attribute: **allow-screen-readers**

Acrobat allows extracting text or graphics for accessibility purposes (such as a screenreader program).

This attribute can only be used if 128-bit encryption has been enabled.

Values

◆ **on**

◆ **of**

☞ *On the web you will find commercially available cracking software that can easily disable 40-bit PDF security settings with a brute-force attack, depending on the length and complexity of the password. Therefore it is advised to use 128-bit encryption if possible (this requires Acrobat 5 for all users of the document).*

☞ *The access permissions set are enforced by the client-viewer used (most of the time this will be Adobe's Acrobat Reader). However, this not necessarily holds true for third-party PDF viewers. These tools can completely ignore these permissions. There is nothing inherent in PDF encryption that enforces the document permissions specified in the encryption dictionary. It is up to the implementors of PDF viewers to respect the intent of the document creator by restricting user access to an encrypted PDF file according to the permissions contained in the file*

xmlmill

3. Existing tags

This chapter describes the changes of the behavior or attributes of existing tags (this means: tags present in a previous version).

☞ *Please note that this chapter only describes the new attributes added to existings tags. For an overview of all attributes of a tag, please visit our [website](#).*

3.1. <header>

3.1.1. Attribute: **margin-top**

The **margin-top** attribute can be used to define the distance from the top edge of the paper to the top edge of the header.

Values

99 Measurement in pixels (px omitted).

99px Measurement in pixels.

99in Measurement in inches.

99cm Measurement in centimeters.

99mm Measurement in millimeters.

Example

```
<header margin-top="3cm" align="center">XMLMill Corporation</header>
```

3.1.2. Attribute: **show-on-first-page**

The **show-on-first-page** attribute can be used to define if the header should be displayed on the first page of the document. As the header is defined in three sections (left, center, right) it is possible to display only a certain part of the header.

Values

◆ **on**

◆ **off**

Example

```
<header align="left" show-on-first-page="on" >XMLMill Corporation</header>  
<header align="right" show-on-first-page="off" ><page-number/></footer>
```

3.2. <footer>

3.2.1. Attribute: **margin-bottom**

The **margin-bottom** attribute can be used to define the distance from the top edge of the paper to the

bottom edge of the header.

Values

99 Measurement in pixels (px omitted).

99px Measurement in pixels.

99in Measurement in inches.

99cm Measurement in centimeters.

99mm Measurement in millimeters.

Example

```
<footer margin-bottom="3cm" align="center"> XMLMill Corporation </footer>
```

3.2.2. Attribute: **show-on-first-page**

The **show-on-first-page** attribute can be used to define if the footer should be displayed on the first page of the document. As the footer is defined in three sections (left, center, right) it is possible to display only a certain part of the header.

Values

◆ **on**

◆ **off**

Example

```
<footer align="left" show-on-first-page="on" >XMLMill Corporation</footer>  
<footer align="right" show-on-first-page="off" ><page-number/></footer>
```

3.3. <field>

The **<field>** tag has been extended with an attribute to indicates the number of rows a field should span.

3.3.1. Attribute: **rowspan**

The **rowspan** attribute can be used to define the number of rows a field should span.

Values

99 A number indicating the number of rows to span.

Example

```
<table>  
  <rows>  
    <row>  
      <field rowspan="2">XMLMill Corporation </field>  
      ...  
    </row>  
  </rows>  
</table>
```

3.3.2. Attribute: **width**

The **width** attribute can be used when defining a complex table. It allows you to define the width of the **<field>** tag.

☞ Please visit the [table](#) chapter for information regarding this new type of table.

Values

- 99** Measurement in pixels (px omitted).
- 99px** Measurement in pixels.
- 99in** Measurement in inches.
- 99cm** Measurement in centimeters.
- 99mm** Measurement in millimeters.
- 99%** Measurement in centimeters (% of the table's width).

Example

```
<table>
  <rows>
    <row>
      <field rowspan="2">XMLMill Corporation </field>
      ...
    </row>
  </rows>
</table>
```

3.4. <file>

The **<file>** tag can now be used inside a **<p>** and **<field>**.

Appears in:

<header>, **<footer>**, **<textbox>**, **<inline>**, **<p>**, **<field>** tag.

Example:

```
<p align="left"><inline font-style="bold"><file/></inline></p>
```

3.5. <date>

The **<date>** tag can now be used inside a **<p>** and **<field>**.

Appears in:

<header>, **<footer>**, **<textbox>**, **<inline>**, **<p>**, **<field>** tag.

Example:

```
<p align="left"><inline font-style="bold"><date/></inline></p>
```

3.6. <time>

The **<time>** tag can now be used inside a **<p>** and **<field>**.

Appears in:

<header>, <footer>, <textbox>, <inline>, <p> and <field> tag.

Example:

```
<p align="left"><inline font-style="bold"><time/></inline></p>
```

3.7. <datetime>

With the <datetime> tag the format of the required date and/or time can be defined.

3.7.1. Attribute: locale

A new attribute is added to make the presentation of the datetime value in the document specific for a geographical, political, or cultural region.

Values

The **locale** attribute has the format of **xx_XX**, meaning:

xx the lowercase represents a valid ISO Language Code. These codes are the lower-case two-letter codes as defined by ISO-639. You can find a full list of these codes at a number of sites, such as:

<http://www.ics.uci.edu/pub/ietf/http/related/iso639.txt>.

XX the uppercase represents a valid ISO Country Code. These codes are the upper-case two-letter codes as defined by ISO-3166. You can find a full list of these codes at a number of sites, such as:

http://www.chemie.fu-berlin.de/diverse/doc/ISO_3166.html.

Example

```
<datetime format="EEE" locale="fr_CA"/>
```

3.7.2. Usage in the <p> and <field> tag

The <datetime> tag can now be used inside a <p> and <field>.

Appears in:

<header>, <footer>, <textbox>, <inline>, <p> and <field> tag.

Example:

```
<p align="left">
  <inline font-style="bold">
    <datetime format="yyyy.MM.dd G 'at' hh:mm:ss z"/>
  </inline>
</p>
```

3.8. <page-number>

The <page-number> tag can now be used inside a <p> and <field>.

Appears in:

<header>, <footer>, <textbox>, <inline>, <p>, <field> tag.

Example:

```
<p align="left">
  <inline font-style="bold">
    <page-number/>
  </inline>
</p>
```

3.9. <page-total>

The <page-total> tag can now be used inside a <p> and <field> tag.

Appears in:

<header>, <footer>, <textbox>, <inline>, <p>, <field> tag

Example:

```
<p align="left">
  <inline font-style="bold">
    <page-total/>
  </inline>
</p>
```

3.10. <table>

The <table> tag now supports complex tables. This means tables of which the rows do not all have the same number of fields (or width). Furthermore the tags to build a table were simplified, in order to facilitate the process of converting a table in HTML to the XMLMill format. The type of table is defined using the **type** attribute.

3.10.1. Attribute: type

The <type> attribute defines if the table is of type **simple** or **complex**.

Values

simple Defines the table as a 'simple' type (the default).

complex Defines the table as a 'complex' type.

☞ *The tables generated in previous versions of XMLMill are of **simple** type.*

3.10.2. Simple table type**Example:**

To make a simple table (the default), nothing changes. The current way to build a table stays the same:

```
<table type="simple" width="11cm" repeatheader="on">
  <columns>
    <cname width="5cm" />
```

```
<cname width="3cm"/>
<cname width="3cm"/>
</columns>
<rows border-style="solid">
  <thead background-grayscale="8%" padding="2px">
    <field >Last Name</field>
    <field >First Name</field>
    <field >City</field>
  </thead>
  <row padding="2px">
    <field >Fitgerald</field>
    <field >Eileen</field>
    <field >London</field>
  </row>
  <row padding="2px">
    <field >Fitgerald</field>
    <field >Eileen</field>
    <field >London</field>
  </row>
</rows>
</table>
```

To define a simple table, use following tags in following order:

1. The **<table>** tag (with the optional **type="simple"** attribute).
2. The **<columns>** tag defining the columns' **width** . Using the **columns** tag eliminates the need to define for each field the width (in each row). Moreover, it strictly limits the number of columns used in the table to the ones defined in this tag.
3. The **<rows>** tag defining the properties that each row should have.
4. The **<thead>** tag defining the attributes for the header-row.
5. The **<row>** tag defining the attributes for each field in the row.
6. The **<field>** tag defining the field's attributes.

☞ *The use of the **id** attribute in the **<field>** tag has been suppressed. Although you can still use an **id**, it will not be used anymore to define the a field's width. As from this version the **order** of the **<field>** tag in the **<row>** tag defines the width (based on the order of the **<cname;>** tags in the **<columns>** tag. So it is allowed to suppress the usage of the **id** attribute in the **<columns>** tag. The **<id>** attributes now just like regular **<id>** attribute.*

3.10.3. Complex table type

The complex table type was introduced to facilitate the generation of a table based on HTML tags. The complex table type eliminates the need to use the **<columns>** tag and each row can have a different number of columns with each their specific width. A complex table can be made, taking into consideration following rules:

- ◆ Specify that the table is of a complex type (using the **type="complex"** attribute in the **<table>** tag.
- ◆ Do not use the **<columns>** tag or its children.
- ◆ Define for each **<field>** its width, using the **width** attribute. This is a new attribute added specifically to make complex tables possible.
- ◆ For each row, the number of **<field>** tags can differ.
- ◆ The sum of all the **<field>width** should not exceed the table's width (defined in the **<table>** tag.

Example:

```
<table type="complex" width="11cm" repeatheader="on">
  <rows border-style="solid" padding="2px">
    <thead background-grayscale="8%">
      <field width="5cm">Last Name</field>
      <field width="3cm">First Name</field>
      <field width="3cm">City</field>
    </thead>
    <row>
      <field width="5cm">Fitgerald</field>
      <field width="3cm">Eileen</field>
      <field width="3cm">London</field>
    </row>
    <row>
      <field width="5cm">Fitgerald</field>
      <field width="3cm">Eileen</field>
      <field width="1.5cm">London</field>
      <field width="1.5cm">London</field>
    </row>
  </rows>
</table>
```

To define a complex table, use following tags in following order:

1. The **<table>** tag (with the **requiredtype="complex"** attribute).
2. The **<rows>** tag defining the properties that each row should have.
3. The **<thead>** tag defining the attributes for the header-row.
4. The **<row>** tag defining the attributes for each field in the row.
5. The **<field>** tag defining the field's attributes.

☞ You cannot use the **colspan** attribute in a complex table type, as you need to define a **width** attribute for each **<field>**.

4. Existing attributes

This chapter describes the changes of the behavior of attributes.

4.1. Attribute: color; background-color; border-color; border-top-color; border-right-color; border-bottom-color; border-left-color

The number of predefined color-names have been extended. Following is a list of colors that can be named (including their rgb color value):

aliceblue	rgb(240, 248, 255)
antiquewhite	rgb(250, 235, 215)
aqua	rgb(0, 255, 255)
aquamarine	rgb(127, 255, 212)
azure	rgb(240, 255, 255)
beige	rgb(245, 245, 220)
bisque	rgb(255, 228, 196)
black	rgb(0, 0, 0)
blanchedalmond	rgb(255, 235, 205)
blue	rgb(0, 0, 255)
blueviolet	rgb(138, 43, 226)
brown	rgb(165, 42, 42)
burlywood	rgb(222, 184, 135)
cadetblue	rgb(95, 158, 160)
chartreuse	rgb(127, 255, 0)
chocolate	rgb(210, 105, 30)
coral	rgb(255, 127, 80)
cornflowerblue	rgb(100, 149, 237)
cornsilk	rgb(255, 248, 220)
crimson	rgb(220, 20, 60)
cyan	rgb(0, 255, 255)
darkblue	rgb(0, 0, 139)
darkcyan	rgb(0, 139, 139)
darkgoldenrod	rgb(184, 134, 11)
darkgray	rgb(169, 169, 169)
darkgreen	rgb(0, 100, 0)
darkgrey	rgb(169, 169, 169)
darkkhaki	rgb(189, 183, 107)
darkmagenta	rgb(139, 0, 139)

darkolivegreen	rgb(85, 107, 47)
darkorange	rgb(255, 140, 0)
darkorchid	rgb(153, 50, 204)
darkred	rgb(139, 0, 0)
darksalmon	rgb(233, 150, 122)
darkseagreen	rgb(143, 188, 143)
darkslateblue	rgb(72, 61, 139)
darkslategray	rgb(47, 79, 79)
darkslategrey	rgb(47, 79, 79)
darkturquoise	rgb(0, 206, 209)
darkviolet	rgb(148, 0, 211)
deeppink	rgb(255, 20, 147)
deepskyblue	rgb(0, 191, 255)
dimgray	rgb(105, 105, 105)
dimgrey	rgb(105, 105, 105)
dodgerblue	rgb(30, 144, 255)
firebrick	rgb(178, 34, 34)
floralwhite	rgb(255, 250, 240)
forestgreen	rgb(34, 139, 34)
fuchsia	rgb(255, 0, 255)
gainsboro	rgb(220, 220, 220)
lightpink	rgb(255, 182, 193)
lightsalmon	rgb(255, 160, 122)
lightseagreen	rgb(32, 178, 170)
lightskyblue	rgb(135, 206, 250)
lightslategray	rgb(119, 136, 153)
lightslategrey	rgb(119, 136, 153)
lightsteelblue	rgb(176, 196, 222)
lightyellow	rgb(255, 255, 224)
lime	rgb(0, 255, 0)
limegreen	rgb(50, 205, 50)
linen	rgb(250, 240, 230)
magenta	rgb(255, 0, 255)
maroon	rgb(128, 0, 0)
mediumaquamarine	rgb(102, 205, 170)
mediumblue	rgb(0, 0, 205)
mediumorchid	rgb(186, 85, 211)

mediumpurple	rgb(147, 112, 219)
mediumseagreen	rgb(60, 179, 113)
mediumslateblue	rgb(123, 104, 238)
mediumspringgreen	rgb(0, 250, 154)
mediumturquoise	rgb(72, 209, 204)
mediumvioletred	rgb(199, 21, 133)
midnightblue	rgb(25, 25, 112)
mintcream	rgb(245, 255, 250)
mistyrose	rgb(255, 228, 225)
moccasin	rgb(255, 228, 181)
navajowhite	rgb(255, 222, 173)
navy	rgb(0, 0, 128)
oldlace	rgb(253, 245, 230)
olive	rgb(128, 128, 0)
olivedrab	rgb(107, 142, 35)
orange	rgb(255, 165, 0)
orangered	rgb(255, 69, 0)
orchid	rgb(218, 112, 214)
palegoldenrod	rgb(238, 232, 170)
palegreen	rgb(152, 251, 152)
paleturquoise	rgb(175, 238, 238)
palevioletred	rgb(219, 112, 147)
papayawhip	rgb(255, 239, 213)
peachpuff	rgb(255, 218, 185)
peru	rgb(205, 133, 63)
pink	rgb(255, 192, 203)
plum	rgb(221, 160, 221)
powderblue	rgb(176, 224, 230)
purple	rgb(128, 0, 128)
red	rgb(255, 0, 0)
rosybrown	rgb(188, 143, 143)
royalblue	rgb(65, 105, 225)
saddlebrown	rgb(139, 69, 19)
salmon	rgb(250, 128, 114)
ghostwhite	rgb(248, 248, 255)
gold	rgb(255, 215, 0)
goldenrod	rgb(218, 165, 32)

gray	rgb(128, 128, 128)
grey	rgb(128, 128, 128)
green	rgb(0, 128, 0)
greenyellow	rgb(173, 255, 47)
honeydew	rgb(240, 255, 240)
hotpink	rgb(255, 105, 180)
indianred	rgb(205, 92, 92)
indigo	rgb(75, 0, 130)
ivory	rgb(255, 255, 240)
khaki	rgb(240, 230, 140)
lavender	rgb(230, 230, 250)
lavenderblush	rgb(255, 240, 245)
lawngreen	rgb(124, 252, 0)
lemonchiffon	rgb(255, 250, 205)
lightblue	rgb(173, 216, 230)
lightcoral	rgb(240, 128, 128)
lightcyan	rgb(224, 255, 255)
lightgoldenrodyellow	rgb(250, 250, 210)
lightgray	rgb(211, 211, 211)
lightgreen	rgb(144, 238, 144)
lightgrey	rgb(211, 211, 211)
sandybrown	rgb(244, 164, 96)
seagreen	rgb(46, 139, 87)
seashell	rgb(255, 245, 238)
sienna	rgb(160, 82, 45)
silver	rgb(192, 192, 192)
skyblue	rgb(135, 206, 235)
slateblue	rgb(106, 90, 205)
slategray	rgb(112, 128, 144)
slategrey	rgb(112, 128, 144)
snow	rgb(255, 250, 250)
springgreen	rgb(0, 255, 127)
steelblue	rgb(70, 130, 180)
tan	rgb(210, 180, 140)
teal	rgb(0, 128, 128)
thistle	rgb(216, 191, 216)
tomato	rgb(255, 99, 71)



turquoise	rgb(64, 224, 208)
violet	rgb(238, 130, 238)
wheat	rgb(245, 222, 179)
white	rgb(255, 255, 255)
whitesmoke	rgb(245, 245, 245)
yellow	rgb(255, 255, 0)
yellowgreen	rgb(154, 205, 50)

xmlmill

5. Java API

This chapter describes the changes of in the **PDX** API in this release.

☞ *For a full overview of all methods, please visit the JavaDoc API in the **docs/apidocs/** directory in the download.*

5.1. Class: PDXTransform

5.1.1. Method: setFeature(String, boolean)

This method enables a particular feature in the **com.xmlmill.PDXTransformHandler** instance.

The **String** parameter represents the feature. The **boolean** parameter defines if the parameter needs to be set on or off.

Following parameters can be defined:

- ◆ **http://com.xmlmill/transform/print-comments**
- ◆ **http://com.xmlmill/transform/print-status-messages**
- ◆ **http://com.xmlmill/transform/XSLTTransformationOnly**

The **http://com.xmlmill/transform/print-comments** parameter replaces the **printComments(boolean)** methods of previous versions.

The **http://com.xmlmill/transform/XSLTTransformationOnly** parameter replaces the **setXSLTTransformationOnly(boolean)** methods of previous versions.

The **http://com.xmlmill/transform/print-status-messages** parameter enables or disables the printing of following messages in the (standard) log (error) stream:

- ◆ **Transforming (...) to (...)**
- ◆ **-- Done (xxxx bytes - xxxx elapsed (milliseconds)) --**

Default above messages are printed in the (standard) log (error) stream.

☞ *This method can be usefull on an application server that generates a high volume of PDF documents. This way the number of messages written in a standard application server log file are minimized.*

5.1.2. Method: boolean getFeature(String)

Return if the indicated feature is on or off.

5.1.3. Method: void printComments(String)

This method has been deprecated. Use **setFeature(String, boolean)** instead.

5.1.4. Method: void setXSLTTransformationOnly(boolean b)

This method has been deprecated. Use **setFeature(String, boolean)** instead.

5.2. Class: PDELogFile

This class has been **deprecated**. Please use the **PDXLogFile** class instead.

xmlmill

6. Known bugs

Following are known bugs in this version:

1. When using a relative url in an external-destination (like **file:/rnotes.xml**), it is not correctly converted.
2. A 16 bit BMP image is not correctly displayed.
3. A 4-bit rle BMP image is not correctly displayed.
4. A PCX file with 4 color-planes 4 is correctly displayed.
5. The border-width of a row (or field) in a table cannot be defined.
6. A **java.lang.NullPointerException** is printed in error log when a DOM is transformed (the document will be correctly generated).
7. Image not printed in <field> when height="100%".

xmlmill

7. Disclaimer

This software redistributes parts of The Apache Software Foundation binary code, hence the disclaimer below:

The Apache Software License, Version 1.1

- * Copyright (c) 1999 The Apache Software Foundation. All rights reserved.
- *
- * Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:
- *
- * 1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
- *
- * 2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
- *
- * 3. The end-user documentation included with the redistribution, if any, must include the following acknowledgment:
 - * "This product includes software developed by the
 - * Apache Software Foundation (<http://www.apache.org/>)."
 - * Alternately, this acknowledgment may appear in the software itself, if and wherever such third-party acknowledgments normally appear.
- *
- * 4. The names "Xalan" and "Apache Software Foundation" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact apache@apache.org.
- *
- * 5. Products derived from this software may not be called "Apache", nor may "Apache" appear in their name, without prior written permission of the Apache Software Foundation.
- *
- * THIS SOFTWARE IS PROVIDED ``AS IS'' AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.
- * =====
- *
- * This software consists of voluntary contributions made by many individuals on behalf of the Apache Software Foundation and was originally based on software copyright (c) 1999, Lotus Development Corporation., <http://www.lotus.com>. For more information on the Apache Software Foundation, please see [<http://www.apache.org/>](http://www.apache.org/).
- *