

---

XMLMill

---

Release Notes

**XMLMill v 1.51**

**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.

# Table of Contents

<b>1. Preface.</b> .....	<b>5</b>
<b>2. Upgrading from previous versions.</b> .....	<b>6</b>
2.1. Upgrading from version 1.50. ....	6
2.2. Upgrading from earlier versions. ....	6
<b>3. Support for GIF images.</b> .....	<b>7</b>
3.1. Up to 256 colors. ....	7
3.2. Transparent GIF images. ....	7
3.3. Interlaced images. ....	7
3.4. Animated images. ....	7
<b>4. New tags.</b> .....	<b>9</b>
4.1. <creationdate>. ....	9
4.2. <break>. ....	10
<b>5. Bugfixes.</b> .....	<b>12</b>
<b>6. Known Bugs.</b> .....	<b>13</b>

xmlmill

xmlmill

## 1. Preface

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

This release notes document describes all new (or enhanced) functionalities of version 1.51 of XMLMill.

The main changes are:

- ◆ Extended support for the GIF image type.
- ◆ A new tag `<creationdate>` is introduced allowing the document's creation date to be defined (previously it was always the system date).
- ◆ A new tag `<break>` is introduced facilitating inserting a line-feed at a predefined point.
- ◆ The applet's certificate has been renewed so XMLMill can continue to be used in an applet.
- ◆ Bug fixes.

For an overview of how to use XMLMill in general 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 tags and their attributes.

If you have questions, please do not hesitate to send a mail to [support@xmlmill.com](mailto:support@xmlmill.com).

☞ *This document is completely generated with XMLMill 1.51 using **rnotes151.xml** and **rnotes.xsl**. These files can be found in the directory **samples/docs/rnotes** in the download.*

## 2. Upgrading from previous versions

### 2.1. Upgrading from version 1.50

One change has been done that could influence the presentation of your .pdf documents:

The point size of the default font was wrongly defined in the default **config.xml** file (located at **com/xmlmill/conf/** path in the **xmlmill.jar** file).

It was defined at **12pt** instead of **10pt** (this was the default value in all versions prior to version 1.50). In order to keep compatibility with the versions before 1.50 it has been set back to **10pt**. If necessary you need to adapt your .xsl/.mill files so that elements relying on the default font size needs to have an explicit attribute: **font-size="12pt"**.

### 2.2. Upgrading from earlier versions

The best procedure to make your transition as smooth as possible is as follows:

1. Back up all your xml/xsl/mill files.
2. Read all release notes [online](#) from your current version to the latest version in correct order (from oldest release notes to newest).
3. Modify your xml/xsl/mill files accordingly to the release notes.

xmlmill

## 3. Support for GIF images

### Description:

This version supports the two types of GIF specifications:

- ◆ GIF87a specification.
- ◆ GIF89a specification.

Following formats are accepted:

- ◆ Up to 256 colors using 1 to 8 bits.
- ◆ Transparent GIF images.
- ◆ Interlaced images.
- ◆ Animated images.

### 3.1. Up to 256 colors

This version accepts GIF images with a color table containing 1 to 256 colors. If a local colortable is defined, the local colortable will be used. If no local colortable is defined, the global colortable will be used.

### Example:

```

```

### 3.2. Transparent GIF images

This version accepts GIF images with a transparent color defined using the **Transparent Color Index** in the **Graphic Control Extension** block. The image will automatically be transparent with the default background of the document.

### Example:

```

```

### 3.3. Interlaced images

XMLMill supports interlaced images. No special action is required.

### Example:

```

```

### 3.4. Animated images

If an animated image is detected, the first image defined in the GIF file will be transformed.

**Example:**

```

```

xmlmill

## 4. New tags

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

### 4.1. <creationdate>

#### Description:

With the `<creationdate>` the creation date of the document can be defined.

#### Appears in:

`<info>` tag.

#### Example:

```
<info>
  <creationdate>D:199812231952-08'00'</creationdate>
</info>
```

#### Usage:

PDF defines a standard date format, which closely follows that of the international standard ASN.1 (Abstract Syntax Notation One), defined in ISO/IEC 8824. A date is a string of the form:

**D:YYYYMMDDHHmmSSOHH'mm'** where:

1. **YYYY** is the year.
2. **MM** is the month.
3. **DD** is the day (01±31).
4. **HH** is the hour (00±23).
5. **mm** is the minute (00±59).
6. **SS** is the second (00±59).
7. **O** is the relationship of local time to Universal Time (UT), denoted by one of the characters +, -, or Z (see below).
8. **HH** followed by ' is the absolute value of the offset from UT in hours (00±23).
9. **mm** followed by ' is the absolute value of the offset from UT in minutes (00±59).
10. The quotation mark character (') after **HH** and **mm** is part of the syntax.

☞ *All fields after the year are optional. (The prefix **D:**, although also optional, is strongly recommended.)*

The default values for **MM** and **DD** are both 01; all other numerical fields default to zero values. A plus sign (+) as the value of the **O** field signifies that local time is later than UT, a minus sign (-) that local time is earlier than UT, and the letter **Z** that local time is equal to UT. If no UT information is specified, the relationship of the specified time to UT is considered to be unknown. Whether or not the time zone is

known, the rest of the date should be specified in local time.

For example, December 23, 1998, at 7:52 PM, U.S. Pacific Standard Time, is represented by the string **D:199812231952-08'00'**

☞ *If the creation date is not defined, the creation date will be by default the system date.*

## **4.2. <break>**

### **Description:**

The **<break>** tag facilitates defining where in the text a line-feed should occur.

### **Appears in:**

- ◆ **<header>** tag.
- ◆ **<field>** tag.
- ◆ **<footer>** tag.
- ◆ **<inline>** tag.
- ◆ **<p>** tag.
- ◆ **<textbox>** tag.

### **Example:**

```
<ml:p>I want a CR here <ml:break> and to continue on the next line.
```

### **Usage:**

In previous versions a line-feed could only be defined by entering a hard-coded carriage return in a .mill or .xsl document:

**In a .mill document:**

```
<ml:p>I want a CR here  
and to continue on the next line.
```

or

**In a .xsl document:**

```
<ml:p>I want a CR here <xsl:text>&#xD;</xsl:text> and to continue on the next line.
```

The problem is that if the **linefeed-treatment** is defined as **ignore** it is impossible to keep the carriage returns as defined above as these are filtered away.

To solve this issue the **<break>** tag is introduced (see example above):

The **<break>** tag ensures that a line-feed is correctly generated in the .pdf document at the specified position (irrespective of the value of the **linefeed-treatment** attribute).

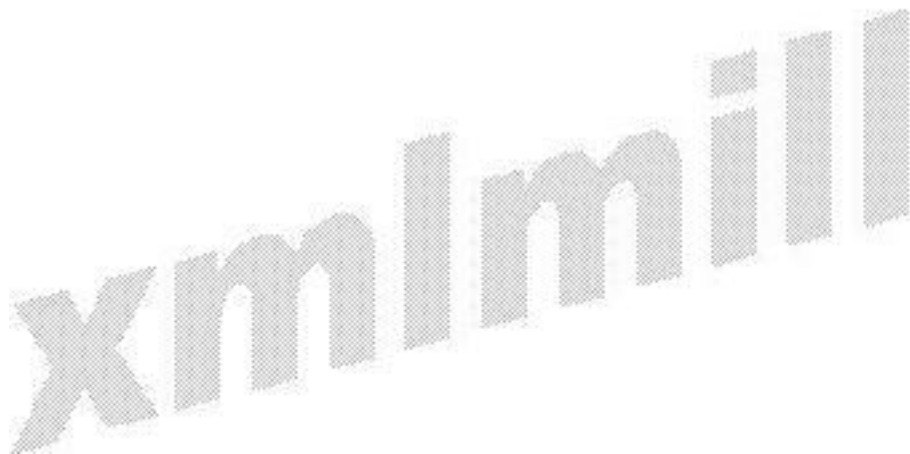
☞ *This also facilitates greatly transforming **xhtml** code to .pdf as the **<br />** tag can easily be converted to **<ml:break>** tag.*

xmlmill

## 5. Bugfixes

Following bugfixes were done:

1. Unicode characters not printed. A last minute code change prevented characters defined as unicode (for example: 0x20AC) as being printed. This is solved.
2. If a cell had a rowspan bigger than 1 and was in the last row of the table, the content was not printed.
3. Added `<?xml version="1.0" encoding="UTF-16" ?>` in all .xsl examples in the **samples/xmlxsl** directory.
4. Empty lines in a **.pfa** or **.afm** file prevented XMLMill from reading the the fontmetrics or fontprogram correctly.
5. GIF images are now correctly transformed.
6. The Domino connector now correctly reads document fields with a length larger the 80 characters. No additional carriage returns are added anymore.



## 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%".
8. The pagenummering collides with the surrounding words (if any) when displayed in a header or footer).

xmlmill