

DITACMS

DITA CMS Release 4.1: Detailed Release Notes

Table of contents

About this document	
Release notes	
Update the index definition for assigned roles and performance issues	4
Working with embedded raster graphics in SVG files	5
Enabling embedded raster images in SVG	6
Concurrent Localization Manager: Specify the format of the localization files generated	8
Set the maximum memory available for the PDF XEP transformation	8
Update the outgen-init-client.xml file to fix the environment path and outgen.job.ant.java.memory property	9

About this document

This document describes the new features and bug fixes introduced in this build of DITA CMS Release 4.1.

Release notes

Update the index definition for assigned roles and performance issues

You need to update the index definition to solve the following two issues:

- Active assignments were not showing when assigned to multiple users.
- A performance issue was reported when indexing large documents (over 12 MB).

To solve this issue, you need to update two indexes in the Index Definition document. Updating these indexes will have a **major** impact on the indexing process, so IXIASOFT recommends that you change the Index Definition document during off hours only. This operation will require a reindexing of the TEXTML Server, which will reduce the performance of the TEXTML Server and may impede user actions (for example, newly created topics will only be searchable using the Advanced Search).

To update the indexes:

1. **Open the TEXTML Administration perspective.**
2. **Connect to your server and Content Store.**
3. **Expand the Content Store node to display the Index Definition branch.**
4. **Right-click Index Definition and select Check out.**
5. **Open the Index Definition Document in an XML editor.**
6. **Look for the following index:**

```
<index CUSTOMPROPERTY="True" NAME="assigned_roles" SYNC="True">
  ...
</index>
```

7. **Change the contents of the index to the following:**

```
<index CUSTOMPROPERTY="True" NAME="assigned_roles" SYNC="True">
  <!-- System index required by the DITA CMS -->
  <stringindex KEEPEXTRACTEDVALUES="True">
    <elements>
      <element DEPTH="INFINITE" XPATH="for $s in //assignedTo return
concat($s/parent::node()/attribute::role , ' ; ' , $s , ' ; ' , //status)"/>
    </elements>
  </stringindex>
</index>
```

8. Look for the following index:

```
<index NAME="tags">
  ...
</index>
```

9. Change the contents of the index to the following:

```
<index NAME="tags">
  <stringindex KEEPEXTRACTEDVALUES="True">
    <elements>
      <element DEPTH="INFINITE" XPATH="if (/*[contains(@class, ' project/project ') or
contains(@class, ' imagemeta/imagemeta ') or contains(@class, ' formatted/formatted ')
or contains(@class, ' resourcemeta/resourcemeta ') or contains(@class, '
multistream/multistream ') or contains(@class, ' resourcepkg/map ')]) then ('') else
distinct-values(for $e in (//*) return local-name($e))"/>
    </elements>
  </stringindex>
</index>
```

10. Save, close, and check in the index definition document.

Working with embedded raster graphics in SVG files

The DITA CMS now supports SVG images with embedded raster graphics.

When you add an SVG image to the DITA CMS, it adds the main SVG image as well as any embedded images, as shown below:

Images				
Format Name	Mime Type	Width (px)	Height (px)	
Lowres				
Medres				
▲ Hires	image/svg+xml	339	333	
Hires_cartoon_1	image/jpeg	50	50	
Hires_cartoon_2	image/jpeg	50	50	
PDF				
Web				
Print				

You can edit the main SVG image as well as its embedded images. When you edit an SVG image from the DITA CMS (by clicking **Edit** in the Show/Edit Image dialog), you can make the following changes:

- Resize the main image or its embedded images
- Embed a new image
- Delete an embedded images

When you save the image and return to the DITA CMS, the images are updated in the Content Store and the Image dialog is updated with the new image information.

See the *DITA CMS User Guide* for more information about images.

Enabling embedded raster images in SVG

The DITA CMS now supports SVG images with embedded raster graphics.

If you are upgrading an existing installation to 4.1.6, you need to add the SVG 1.0 and 1.1 plugins to the DITA CMS and Output Generator.

1. Download the plugins for the SVG DTDs from the following URL:

http://cms.ixiasoft.com/downloads/system_config/4.1/SVG_plugins_Release_Notes_4.1.6.zip

2. Extract the downloaded file to a temporary directory; for example:

```
C:\ixiasoft\temp\svg\plugins
```

The `plugins` directory now includes the following two plugins:

- `org.w3c.svg1.0`
- `org.w3c.svg1.1`

3. Open the TEXTML Administration perspective.

4. Connect to your server and Content Store.

5. Navigate to the `Repository/system/` directory.

6. Right-click the `system` directory and select Insert Documents...

The **Insert Documents Dialog** window appears.

7. Click Add Folder and browse to the directory where you downloaded the plugins.

For example, `C:\ixiasoft\temp\svg\plugins`.

8. Select the `plugins` directory and click OK.

The folder is added to your list of files to be imported.

9. In the Set Options:

- Select **Add**.
- Select **Replace**.
- Clear **Indexable**.

10. Click OK.

The plugins are imported into the `plugins` directory.

11. Navigate to the *Repository/system/catalogs* directory.
12. Check out and open *master-catalog.xml*.
13. Before the closing `</catalog>` element, add the following:

```
<!-- SVG DTDs -->
<nextCatalog catalog="../../../plugins/org.w3c.svg1.0/catalog.xml"/>
<nextCatalog catalog="../../../plugins/org.w3c.svg1.1/catalog.xml"/>
```

14. Save, close, and check in *master-catalog.xml*.
15. Navigate to the *system/plugins/com.ixiasoft.dita.dtd/dtd* directory.
16. Check out and open *Imagemeta.mod*.
17. Look for the following lines:

```
<!ELEMENT imageinfo      (%formatname;, %mime-type;, %width;,
                           %height;)?                                     >
```

18. Change the second line to the following:

```
<!ELEMENT imageinfo      (%formatname;, %mime-type;, %width;,
                           %height;, (%imageinfo;)*)?                   >
```

19. Save, close, and check in *Imagemeta.mod*.
20. Synchronize the configuration.
21. Copy the *org.w3c.svg1.0* and *org.w3c.svg1.1* plugins to the `%OutputGenDir%/data/%OT_Dir%/plugins` directory.
22. Go to the directory where the version of the DITA OT that you are using is installed.
For example, if you are using the 1.8.5 DITA-OT, go to the `%OutputGenDir%/data/DITA-OT1.8.5/` directory.
23. Double-click `startcmd.bat` (Windows) or run `startcmd.sh` (Linux).
24. At the command prompt, enter:

```
ant -f integrator.xml
```

The integration build runs and you should soon see BUILD SUCCESSFUL.

Note: If a message similar to the following is displayed, you can safely ignore it:

```
Unable to locate tools.jar. Expected to find it in C:\Program Files\Java\jre6\lib\tools.jar
```

This message means that you are using a JRE instead of a JDK to run the Output Generator.

25. Close the command window.

Concurrent Localization Manager: Specify the format of the localization files generated

The Concurrent Localization Manager now includes a configuration parameter to specify whether to produce XLIFF or DITA files.

This parameter, `bundle.creation.xliffFormat`, lets you specify whether to generate the localization kit in XLIFF ("true") or DITA ("false").

See the "Using the concurrent localization method" topic in the *DITA CMS System Administrator's Guide* for the detailed procedure.

Set the maximum memory available for the PDF XEP transformation

You can now configure the maximum memory available for the JVM that runs the RenderX XEP engine to generate the PDF.

A new parameter, `outgen.job.custom.xep.memory`, lets you specify the maximum memory available for the transformation.

You can set this parameter in the *outgen-init-client.xml.orig* file or specifically for a transformation scenario.

For example, to set the parameter in the *outgen-init-client.xml.orig* file, enter the following:

```
<!--
*****
* Config for xep
*****
-->
<!--<property name="outgen.job.xep.dir" location="/usr/local/RenderX/XEP"/>-->
<property name="outgen.job.xep.dir" location="C:\Program Files\RenderX\XEP"/>
<!-- If you need a custom setting file for different output types,
do not set the value here; set it instead in each target that calls the renderer. -->
<!--<property name="outgen.job.custom.xep.config"
location="{outgen.resources.dir.client}/xep.xml"/>-->
<property name="outgen.job.custom.xep.memory" value="1024m"/>
```

By default, if you do not configure this parameter, 500 MB is available for the transformation.

See *Installing and Configuring the Output Generator* for more information.

Update the `outgen-init-client.xml` file to fix the environment path and `outgen.job.ant.java.memory` property

The path environment variable set in the `outgen-init-client.xml` configuration file was not passed properly in Linux. Also, the file contained a syntax error in the `outgen.job.ant.java.memory` property.

To fix these errors:

1. **Open the `%OutputGenDir%/data/outgen-init-client.xml` file.**
2. **Look for the following lines:**

```
<!--<property name="outgen.job.ant.path"
value="${outgen.ot.dir}${file.separator}tools${file.separator}ant
${file.separator}bin${path.separator}${env.Path}"/>-->
```

3. **If you have these lines, change them to the following lines:**

```
<!-- Use this line for Windows -->
<!--<property name="outgen.job.ant.path"
value="${outgen.ot.dir}${file.separator}tools${file.separator}ant
${file.separator}bin${path.separator}${env.Path}"/>-->
<!-- Use this line for Linux -->
<!--<property name="outgen.job.ant.path"
value="${outgen.ot.dir}${file.separator}tools${file.separator}ant
${file.separator}bin${path.separator}${env.PATH}"/>-->
```

Note: If you have set a value for the `outgen.job.ant.path` property, and the value is different than the two values above, keep it.

4. **Look for the following line:**

```
<!--<property name="outgen.job.ant.java.memory="512m"/>-->
```

5. **If it's in the file, change it to:**

```
<!--<property name="outgen.job.ant.java.memory" value="512m"/>-->
```

6. **Save and close the file.**