DHIS2 Documentation Guide

2.23

Revision 2174

2016-05-22 10:32:35
Version 2.23

-->

Chapter 1. DHIS2 Documentation Guide

1.1. DHIS 2 Documentation System Overview

DHIS 2 is a web-based aggregate information management system under very active development. Given the modular nature of the system, its wide user base and distributed, global nature of development, a comprehensive documentation system is required. An in-depth discussion of the need for documentation of DHIS 2 has been considered previously. [Store2007] DocBook is a comprehensive XML based system for creation of books, papers and other technical documents maintained by OASIS .

1.2. Introduction

One of the main advantages of DocBook is that there is complete separation between the content and presentation. DocBook is a pure XML format, and is well documented. It is believed that only a very small subset of its features will be required in order to achieve much higher quality documentation for DHIS. There are hundreds of separate mark-up elements that cater to almost any level of technical documentation needs, but in reality, only a few dozen of these element will probably need to be employed to achieve high-quality documentation for DHIS 2, both for printed as well as on-line formats such as HTML or integrated help systems within the application itself.

There exist a wide range of text editors which can be used for the creation of DocBook files. A fairly complete list of possibilities is located here. It is currently recommended to use the <oXygen/>® XML editor for editing DocBook source files. In principle, any text editing program or XML editor can be used to author DocBook files. Other good possibilities are excellent and freely available text editorjEdit with the XML plugin installed, or the cross-platform Emacs text editor. Both of these programs support editing DocBook XML.

One of the key concepts to keep in mind when authoring documentation in DocBook, or other presentation neutral formats, is that the content of the document should be considered in the first instance. The presentation of the document will take place in a separate step, where it will be rendered into different formats, such as HTML and PDF. It is therefore important that the document is will organised and structured, with appropriate DocBook tags and structural elements being considered.

It is good practice to break your document in to various sections using the "sect", or section element. Section elements can also be nested within each other, such as "Section 1" and "Section 2". This concept is essentially the same as Microsoft Word® or other word processing programs. DocBook will automatically take care of numbering the sections for you when the document is produced. Two other important elements are the "itemizedlist" and "numberedlist". These are quite similar, but an itemised list corresponds to a bulleted list, which a numbered list will be rendered with each element being numbered sequentially. Other key elements are "screenshot" and "table" which should be self-explanatory.

1.3. Getting started with GitHub

Currently, the documentation system is part of the source code housed at GitHub. GitHub is a collaborative platform that enables multiple people to work on software projects collaboratively. In order for this to be possible, a version control system is necessary in order to manage all the changes that multiple users may make. GitHub uses the git source control system. While it is beyond the scope of this document to describe the functionality of git, users

who wish to create documentation will need to gain at least a basic understanding of how the system works. A basic guide is provided in the next section.

In order to start adding or editing the documentation, you should first perform a checkout of the source code. If you do not already have a GitHub account, you will need to get one. This can be done here. Once you register with GitHub, you will need to request access to the dhis2-documenters group. Login to GitHub, and then file an issue here. Your request will need to be approved by the group administrators. Once you have been granted access to the group, you can commit changes to the documentation branch and send and receive notifications if you wish.

1.4. Getting the document source

In order to edit the documentation, you will need to download the source pages of the documentation to your computer. GitHub uses a version control system known as git . There are different methods for getting Git working on your system, depending on which operating system you are using. A good step-by-step guide for Microsoft® operating systems can be viewed here. Alternatively, if you are comfortable using the command line, you can download git from this pageIf you are using Linux, you will need to install git on your system through your package manager, or from source code. A very thorough reference for how git is used is available in a number of different formats here.

Once you have installed git on your system, you will need to download the document source. Just follow this procedure:

1. Make sure you have git installed.
2. On Windows® systems, visit https://github.com/dhis2/dhis2docs and press "Clone in Desktop". If you are using the command line, just type git clone git@github.com:dhis2/dhis2docs.git
3. The download process should start and all the documentation source files will be downloaded to the folder that you specified.
4. The DHIS 2 documents depend on other branches for their documentation. Be sure to keep these these up to date as well. When you build the documentation, the necessary submodules will be downloaded automatically as part of the build process ,if you have not already done so.

1.5. Editing the documentation

Once you have downloaded the source, you should have a series of folders inside of the dhis2docs directory. All documents should be placed in the `dhis2docs/src/docbkx/XX` folder. Note that the `XX` represents the ISO 639-1 (two-letter) language code of the documentation. If you are developing English language documentation, place it inside the `/dhis2docs/src/docbkx/en/` folder. Place any image files that may be linked to your document in the `/dhis2docs/src/docbkx/XX/resources/images` folder and link these inside your DocBook document using a relative file link. When the documentation is built, in a separate step, the images will be automatically copied over to the correct directory during the build process.

1.6. DHIS2 Bibliography

DHIS2 has a rich set of academic literature which can serve as useful resources during implementations, project proposals and more detailed reading than what is appropriate for a general user manual. A specialized bibliography has been added to the source code of the application. BibTeX is a specialized language which is widely used in the academic world to handle bibliographic databases. A large number of free and open source tools are capable of

working with BibTeX. Currently, the recommended tool of choice for manipulating the DHIS2 bibliography is JabRef.

To get started with the bibliography, download a copy of JabRef and open the `/src/bibliography/dhis2_bibliography.bib file.` . Add some new references, then export the bibliography back to the `/src/docbkx/en/dhis2_bibliography.xml` file, using the DocBook 4.4 export format. The updated bibliography will automatically be included in the documentation after you commit your changes.

1.7. Using images

Screen shots are very useful for providing information to users on how particular actions should be performed. DocBook has no intrinsic mechanisms to know exactly how an image should be rendered in the final document. Therefore, it is necessary to provide instructions through element attributes. The following XML code fragment demonstrates how an image can be specified to occupy 80% of the available page width. For screen shots in landscape format, this seems to be an appropriate amount. You may need to experiment a bit to obtain a proper width for your image. Alternatively, you can edit the resolution of the image itself, in order to obtain a proper size during rendering.

```
<screenshot>
 <title>DHIS2 Login screen</title>
  <mediaobject>
   <imageobject>
    <imagedata fileref="dhis2_login_screen.jpg" format="JPG" width="80%"/>
   </imageobject>
  </mediaobject>
</screenshot>
```

For other images, depending on their size, a different value may be necessary. If you do not specify a width for you image, and its intrinsic size is larger than the available screen width, the image may overflow in certain document types with a fixed width, such as PDF.

1.8. Linking documents together

DocBook provides a modular framework where many separate documents can be linked together into a master document. Fragments from different documents can also be reused in different contexts. It is therefore important to consider whether your document should be constructed as an article or a chapter. Chapters are essentially portions of a book, and can therefore be linked together into a larger document very easily. Articles are essentially standalone documents, but they can also be assembled together into a larger document at the component level.

Should you wish to link several articles together into a book, DocBook provides a mechanism to assign an id to a section. In the example below, a section has been assigned an id. This id must be unique within the document.

```
  <section xml:id="mod2_1">
<title>Getting started with DHIS2</title> ....
```

In order to include an article into a book, an Xinclude statement must be used. The following example shows how.

```
<chapter>
<title>Getting started with DHIS2</title>
<xi:include xmlns:xi="http://www.w3.org/2001/XInclude" href="dhis2_user_man_mod2.xml"
 xpointer="mod2_1" encoding="UTF-8"/>
...
```

Note that the file name and id have been assigned in the parent document, referring to the actual file (href) and particular fragment of the child document that should be referenced in the parent document (xpointer).

Including chapters in a book is very simple. The example below illustrates how:

```
<xi:include xmlns:xi="http://www.w3.org/2001/XInclude" href="dhis2_user_man_
mod1.xml" encoding="UTF-8"/>
```

In this case, there is no need to explicitly reference a part of the document, unless you only want to include a portion of the chapter. If you want to use a section of the chapter, you can assign an id to that section, and then reference that section through an xpointer.

1.9. Handling multilingual documentation

The directory structure of the documentation has been created in order to facilitate the creation of documents in any language. If you want to create a new set of documents in a given language, simply create a new directory in the `dhis2-docbook-docs/src/docbkx/` directory. Be sure to use the ISO 639-1 code for the language you are going to create documents in. A complete list of these codes can be found here. Add a new folder for images in a sub-directory , replacing XX with the actual ISO 639-1 code for the language you will create documents in. You will also need to edit the `pom.xml` file in the main dhis2docs directory. If you are unsure of what changes need to be made to this file, ask on the mailing list first, as this file controls the generation of all the documentation.

1.10. Building the documentation

One of the key advantages of the DocBook format is that the source documentation can be transformed into a wide variety of formats, including HTML, chunked HTML, XHTML, PDF, and a number of other formats. There are a wide variety of tools that are capable of performing this task. Basically the XML source of the document is transformed using the standard DocBook XSL style sheets into the desired format. The complete list of tools capable of transforming DocBook will not be listed here, but a few examples are provided below.

Latest builds of the documentation are available from the DHIS2 website. The latest snapshot builds are available through the continuous integration server located here.

1.10.1. Building the documentation with Apache maven

In order to transform the documentation source files to different formats, such as HTML or PDF, you will need to install the Apache Maven program. You can get a copy here or by installing it through your package manager if you are using Linux. Just execute the command mvn clean package on Windows or on Linux from the `/dhis2-docbook-docs` directory. Maven will start to download the necessary components to transform the documents into HMTL,PDF and RTF. Once the process has completed (be patient the first time, as there are a number of components that must be downloaded), all of the target document types will be generated in the `/dhis2docs/target/docbkx` directory in respective sub-directories.

1.10.2. Building with xmlto

xmlto is a useful utility available on Linux platforms for transforming DocBook documents into many different formats. More information on the package can be found here. If you do not want to use Apache Maven for some reason, you can install xmlto through your package manager. Once you have installed xmlto you can just execute

`xmlto` [htmlfile_to_transform]
where the *`file_to_transform`* parameter is the name of the file you wish to transform. There are many other formats available, such as PDF, PS, JavaHelp and others.

1.11. Committing your changes back to GitHub

Once you have finished editing your document, you will need to commit your changes back to GitHub. Open up a command prompt on Windows or a shell on Linux, and navigate to the folder where you have placed your documentation. If you have added any new files or folders to your local repository, you will need to add them to the source tree with the git add command, followed by the folder or file name(s) that you have added. After that you need to commit your changes. Be sure to include a descriptive comment with your commit.

`git commit` [-m] ["Created Amharic translation of documentation"]
Finally, you should push the changes back to the repository with git push origin master

If you have any questions, or cannot find that you can get started, just send an email with your problem to `<dhis2-documenters@lists.launchpad.net>`.