

## In Cygwin (or alternatively in a Linux environment)

### 1. Install packages:

- make, gcc
- git, openssl, openssh
- perl (just type perl and click install on the entire group perl)
- [image|graphic|perl]magic (just type magic and click install on the entire groups perl and graphics)
- texlive and related packages (just type texlive and click install on the entire group publishing)
- then optionally as `cpan` dependencies may need to install:
  - unzip, tar, bzip2
  - patch, patchutils
  - curl, wget, lynx
  - crypt, libgcrypt (required by perl package JSON::XS)

Note: it is very important to install the packages make, gcc, perl, and texlive directly into Cygwin even though you might have them already installed in your system; the reason is mainly because of the paths; also in some cases, e.g., ImageMagick, cygwin can load a dll which has a different base for addressing

### 2. Run and configure cpan (perl package system):

1. simply run `cpan` command from the cygwin terminal
2. if it needs to be configured, it will show you a message in the terminal
  - a. follow the instructions
3. type `exit` in cpan terminal

### 3. Install missing perl packages:

- cpan -i File::Which IO::String Image::Size JSON::XS Parse::RecDescent

### 4. Download and install LaTeXML:

1. git clone <https://github.com/bruceMiller/LaTeXML.git>
2. cd LaTeXML
3. perl Makefile.PL
4. make
5. make test (*optional - you might get an error related to french babel, but i don't know how to fix it; plus, it is not important anyway*)
6. make install

## Lyx → LaTeX → XML → HTML / PNG

1. lyx --export latex document.lyx
2. latexml --destination=document.xml document.tex
3. latexml --destination=references.xml references.bib
4. latexmlpost --sitedirectory=. --format=html4 --bibliography=./references.xml  
--css=./customRules.css --destination=document.html document.xml

## The new solution -- Maven

In the current version, there is a hash script called `compile-html-help.sh` which is can be called from maven either directly (in Unix systems), or via Cygwin (in Windows systems).

In NetBeans right-click on pom.xml in project explorer pane and select Run Maven → Genarete HTML Help.

### The bash script:

```
#!/bin/bash

cwd=`pwd`

# LyX --> LaTeX
find . -type f -name \*.lyx -exec "$1" --export latex {} \;

# LaTeX & BibTeX --> XML
latexml --destination=references.bib.xml references.bib
find . -type f -name \*.tex -exec latexml --destination={}.xml {} \;

# XML --> HTML & PNG
find . -type f -name \*.tex.xml -exec latexmlpost --sitedirectory="$cwd" --format=html4
--bibliography="$cwd"/references.bib.xml --css="$cwd"/customRules.css
--destination={}.html {} \;

# remove all the generated files that are no longer needed
#rm references.bib.xml
find . -type f -name LaTeXML.cache -exec rm {} \;
find . -type f -name \*.tex -exec rm {} \;
find . -type f -name \*.tex.xml -exec rm {} \;

# rename the generated html files
for file in $(find . -type f -name \*.tex.xml.html); do mv ${file}
${file%.tex.xml.html}.html; done
```

## The old solution -- Ant

Note that the commands latexml and latexmlpost must be executed from cygwin!

```
<target name="compile-html-help">
  <property file="build.properties" description="local override"/>
  <property name="lyx.path" location="c:/Program Files (x86)/LyX 2.0/bin/"
description="path to lyx bin folder, can be overridden in build.properties"/>
  <property name="doc.root" location="src/resources/help"/>
  <property name="customCSS" location="src/resources/help/customRules.css" />

  <property environment="env"/>

  <apply executable="lyx" failonerror="true" resolveExecutable="false"
vmlauncher="false">
    <env key="Path" path="{env.Path};${lyx.path}"/>
    <arg line="--export latex"/>
    <srcfile/>
    <fileset dir="src/resources/help">
      <include name="**/*.lyx"/>
      <exclude name="**/*.*/>
      <depend targetdir="src/resources/help">
        <mapper type="glob" from="*.lyx" to="*.html"/>
      </depend>
    </fileset>
  </apply>

  <apply executable="latexml" failonerror="true" resolveExecutable="true"
vmlauncher="false" dest="${doc.root}">
    <fileset dir="${doc.root}">
      <include name="**/*.tex"/>
      <exclude name="**/*.*/>
    </fileset>
    <globmapper from="*.tex" to="*.xml"/>

    <srcfile/>
    <targetfile prefix="--destination="/>
  </apply>

  <exec executable="latexml" dir="${doc.root}" vmlauncher="false">
    <arg path="${doc.root}/references.bib"/>
    <arg prefix="--destination=" path="${doc.root}/references.xml"/>
  </exec>

  <apply executable="latexmlpost" failonerror="true" resolveExecutable="true"
vmlauncher="false" dest="${doc.root}">
    <arg value="--sitedirectory=${doc.root}"/>
    <arg value="--xsltparameter=TIMESTAMP:""/>
    <arg value="--format=html"/>
    <arg value="--bibliography=${doc.root}/references.xml"/>
    <arg value="--css=${customCSS}"/>
```

```

    <fileset dir="${doc.root}">
        <include name="**/*.xml"/>
        <exclude name="**/*.*/>
        <exclude name="**/references.xml"/>
    </fileset>
    <globmapper from="*.xml" to="*.html"/>

    <srcfile/>
    <targetfile prefix="--destination="/>
</apply>

<!--
<antcall target="clean html help"/>
-->
</target>

<target name="clean html help">
    <delete failonerror="false">
        <fileset id="myfileset" dir="src/resources/help">
            <include name="**/*.log"/>
            <include name="**/*.tex"/>
            <include name="**/*.xml"/>
            <include name="**/*.cache"/>
            <include name="**/*.eps"/>
        </fileset>
    </delete>
</target>

```