

# PDFPreview

## Purpose

This project use mupdf library to generate preview and printout of pdf documents.

The reason for this is I need to port my old application to Linux and I use external Sumatra pdf viewer for preview and printing of pdf documents. Sumatra does not exists on linux, so...

Also sumatra use libmupdf 1.6 and I want 1.8.

The purpose of the document is to explain how to build libmupdf.dll (32 and 64 bit) for windows and Linux, how to link it in FPC (lazarus or codetyphon) application.

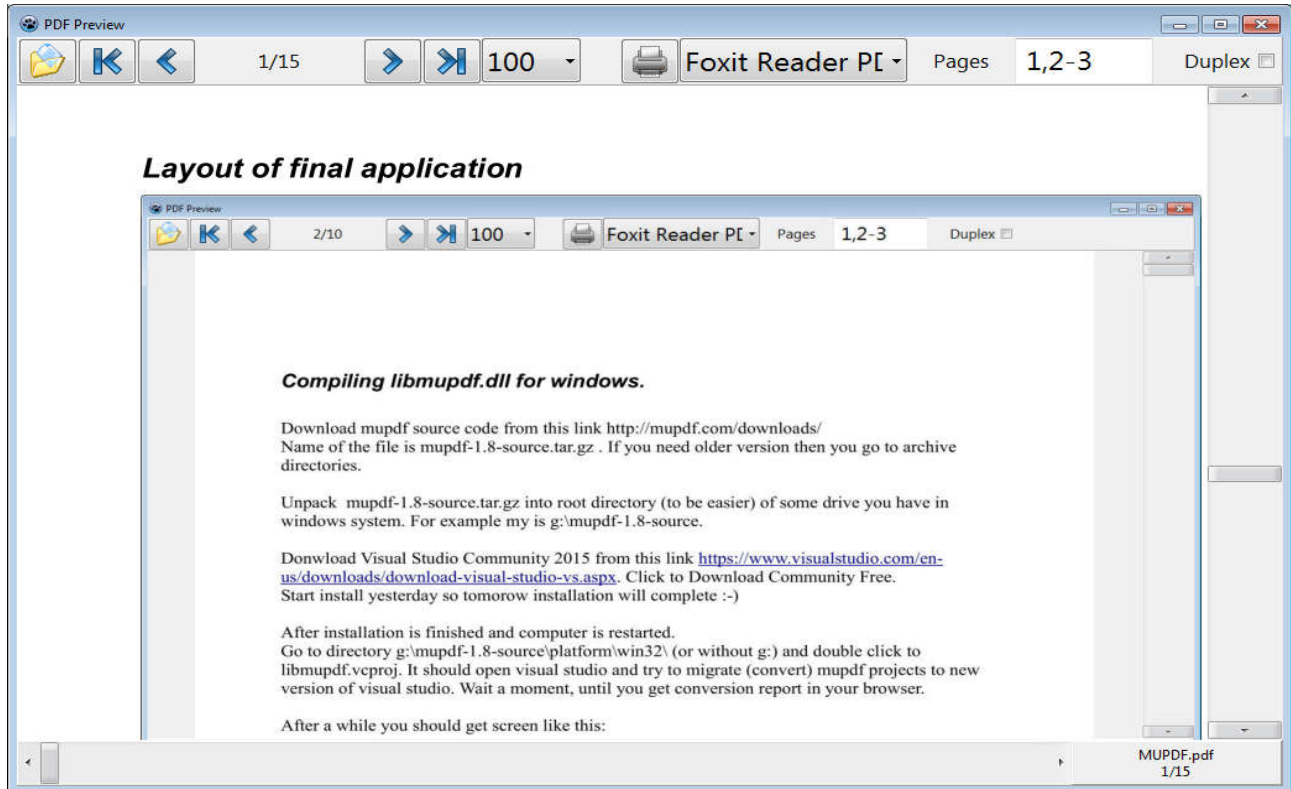
Example project show the use of library with common preview functions. You are encourage to extend this project.

Initial idea for this FPC project is done by Blestan Tabakov. His version use one of first version of libmupdf library.

Problems:

There are lot of changes in cdecl function calls between versions. Big change is between version 1.6 and 1.7. Almost every call in version 1.7 is changed in way that first param is CONTEXT variable. Some functions are changed in way that they have different params or param types.

## Layout of final application



## ***History***

My first approach was to use ver 1.6 of libmupdf. I did not find any precompiled version of it. According to Blestan, Sumatra project always have libmupdf.dll if you use installer and install sumatra. So I did, but guys from Sumatra did not export all function needed by Blestan example. So it was dead end for me. Then I used Sumatra source code and recompile libmupdf.dll but included missing function. This was good. Then I tried to precompile for Linux. But Fedora19 did not have 1.6 and latest version was 1.5. So I decided to go again from start. Use plain mupdf source (latest was 1.8) and recompile for win32 and win64. Almost all calls are changed between 1.6 and 1.8. Some of the functions are renamed.

## Compiling libmupdf.dll for windows (Dynamic Library - dll)

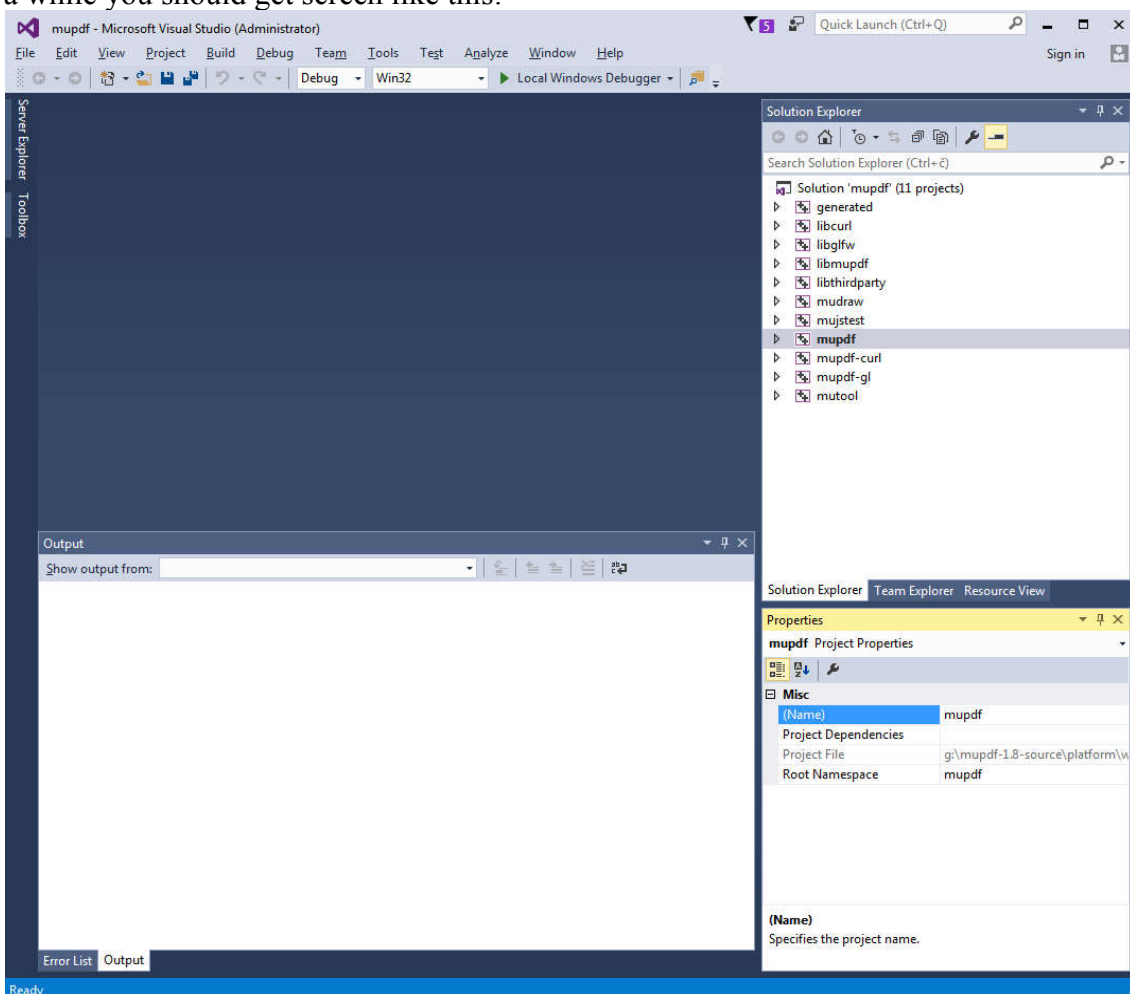
Download mupdf source code from this link <http://mupdf.com/downloads/>  
Name of the file is mupdf-1.8-source.tar.gz . If you need older version then you go to archive directories.

Unpack mupdf-1.8-source.tar.gz into root directory (to be easier) of some drive you have in windows system. For example my is g:\mupdf-1.8-source.

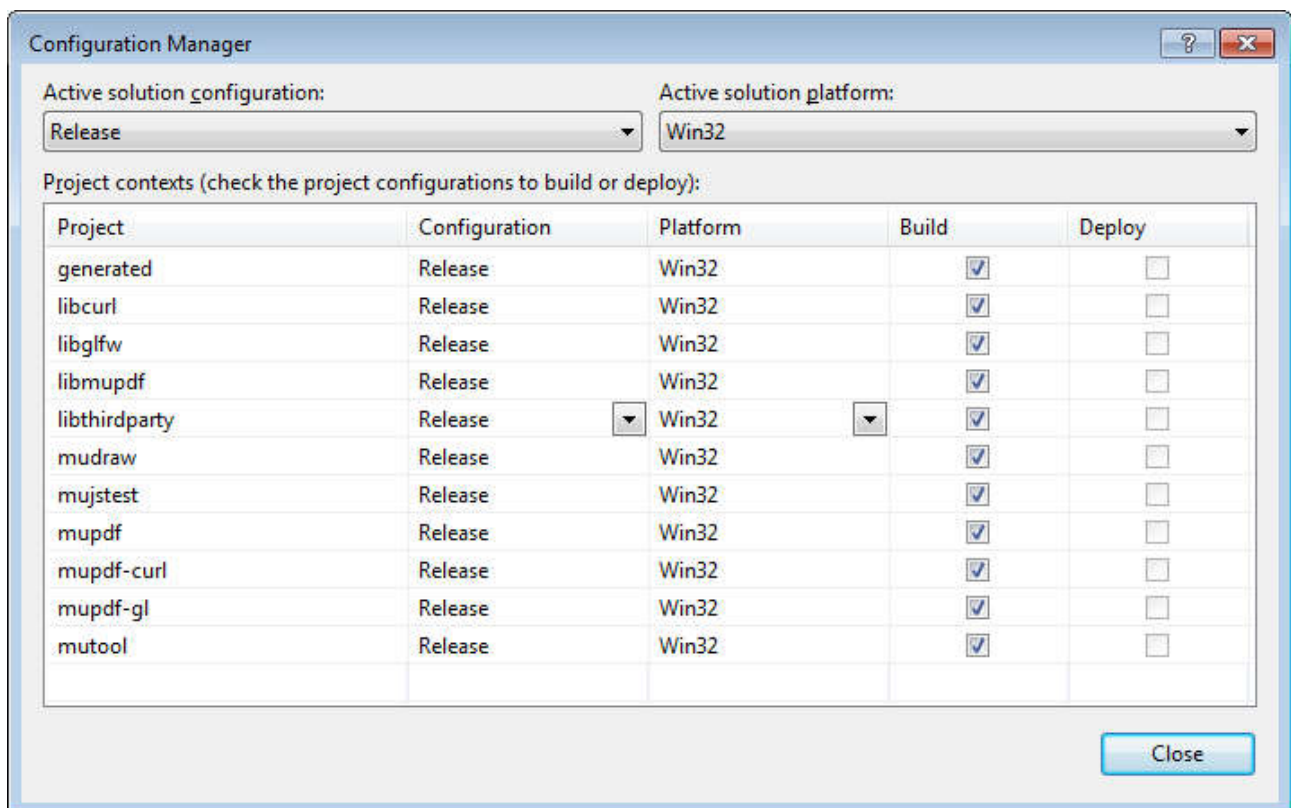
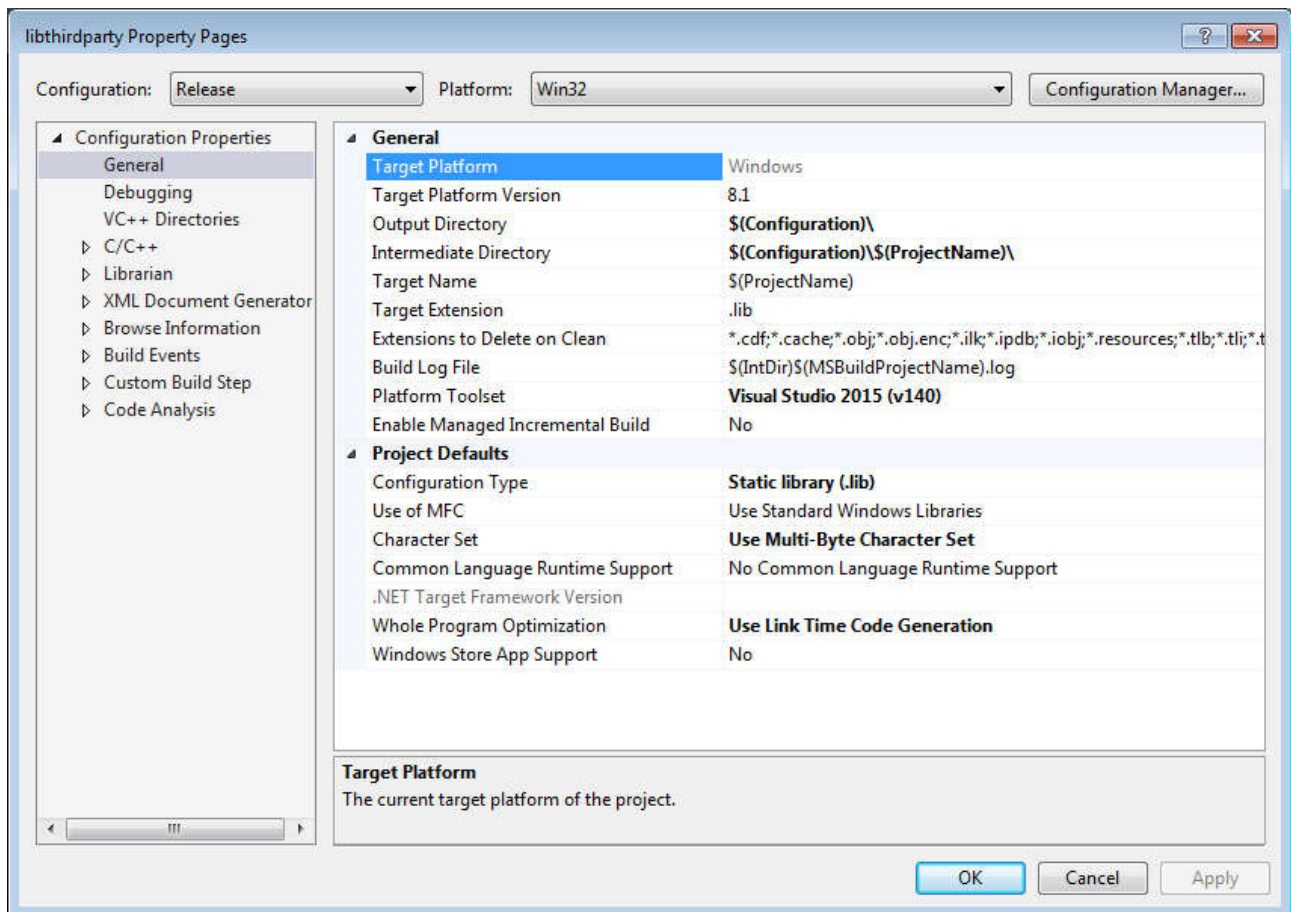
Download Visual Studio Community 2015 from this link <https://www.visualstudio.com/en-us/downloads/download-visual-studio-vs.aspx>. Click to Download Community Free.  
Start install yesterday so tomorrow installation will complete :-)

After installation is finished and computer is restarted.  
Go to directory g:\mupdf-1.8-source\platform\win32\ (or without g:) and double click to libmupdf.vcproj. It should open visual studio and try to migrate (convert) mupdf projects to new version of visual studio. Wait a moment, until you get conversion report in your browser.

After a while you should get screen like this:



Right click to libmupdf third party and chose properties, after that choose Configuration Manager.



Chose Active Soutlion = Release (original is Debug)

Close both windows, right click again to libthirdparty and choose Build. Wait until this lib is built, and you get message like this

```

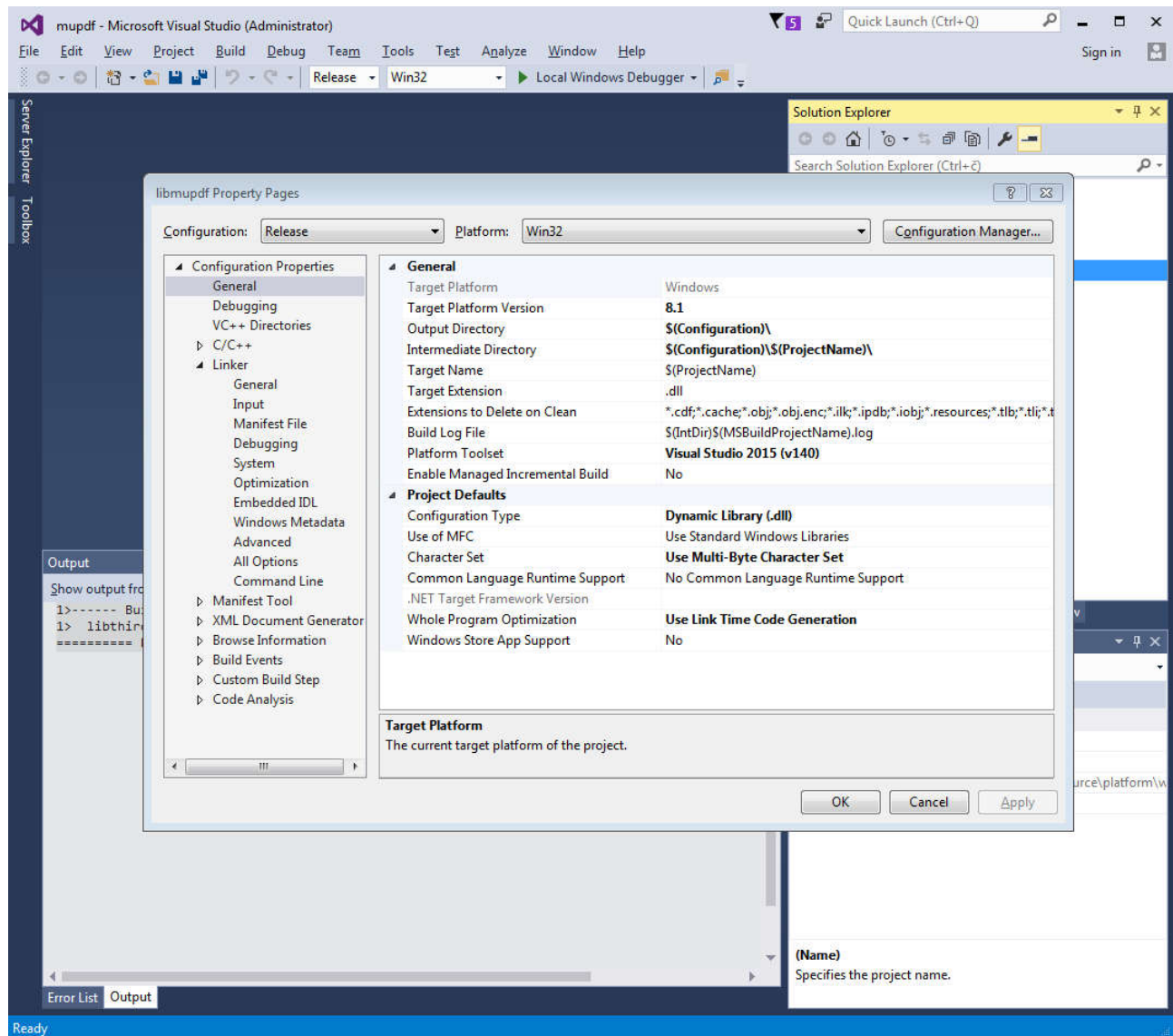
1>----- Build started: Project: libthirdparty, Configuration: Release Win32 -----
1> libthirdparty.vcxproj -> g:\mupdf-1.8-source\platform\win32\Release\libthirdparty.lib
===== Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped =====

```

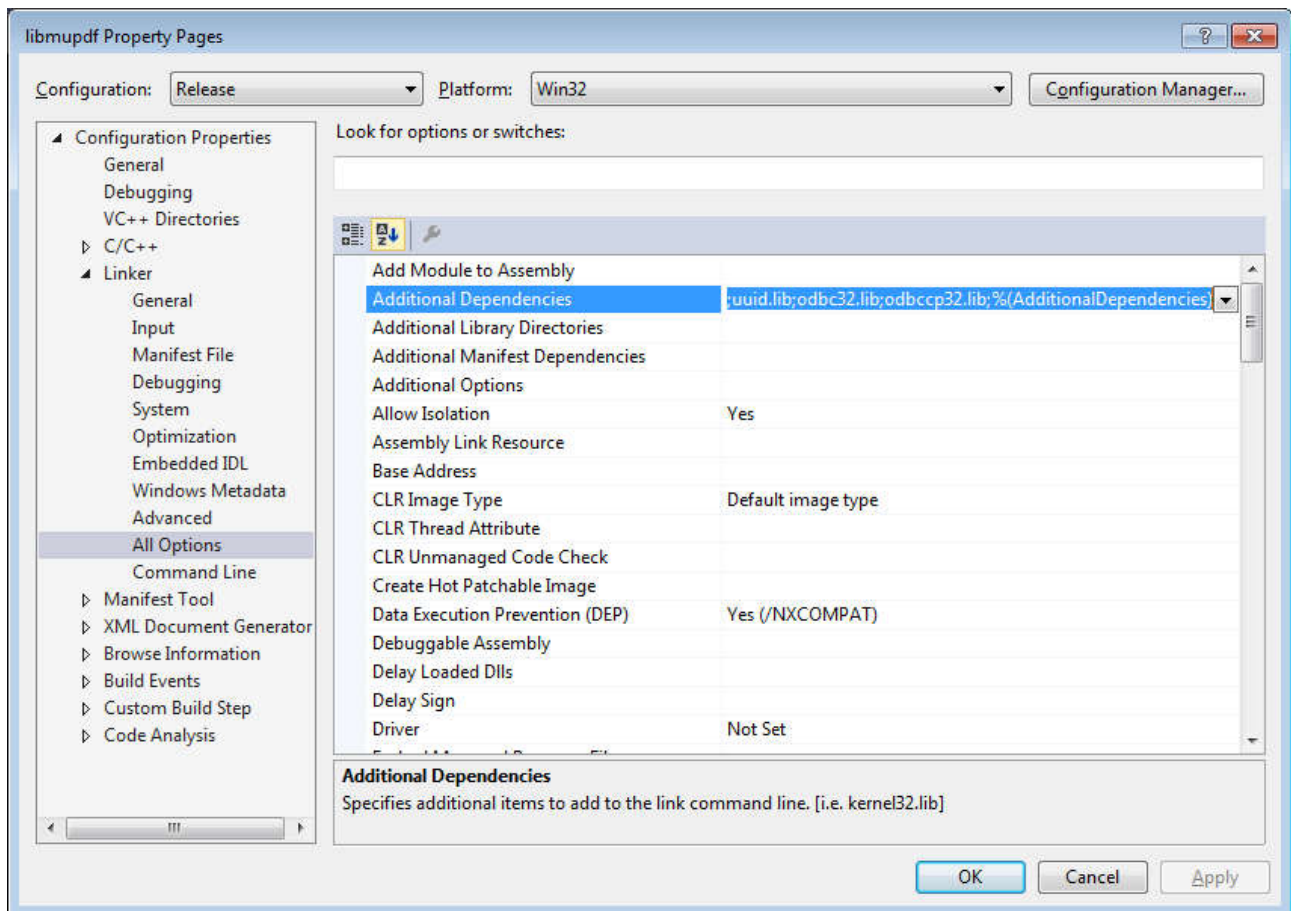
So, first lib is created and it is located in  
g:\mupdf-1.8\source\platform\win32\Release\libthirdparty.lib

Now we must make libmupdf.dll.

Right Click to libmupdf and chose Properties. This library depends on libthirdparty.lib so we have to make a link to it. But first we need to change project configuration type to dll.



After that open Linker options:



Add new library into Additional Dependencies so the value look like:

kernel32.lib;user32.lib;gdi32.lib;winspool.lib;comdlg32.lib;advapi32.lib;shell32.lib;ole32.lib;oleaut32.lib;uuid.lib;odbc32.lib;odbccp32.lib;**g:\mupdf-1.8-source\platform\win32\Release\libthirdparty.lib**;%(AdditionalDependencies)

Now we need list of function which dll have to export and that are needed for our project.

File is /DEF:**g:\mupdf-1.8-source\libmupdf.def** %(AdditionalOptions)

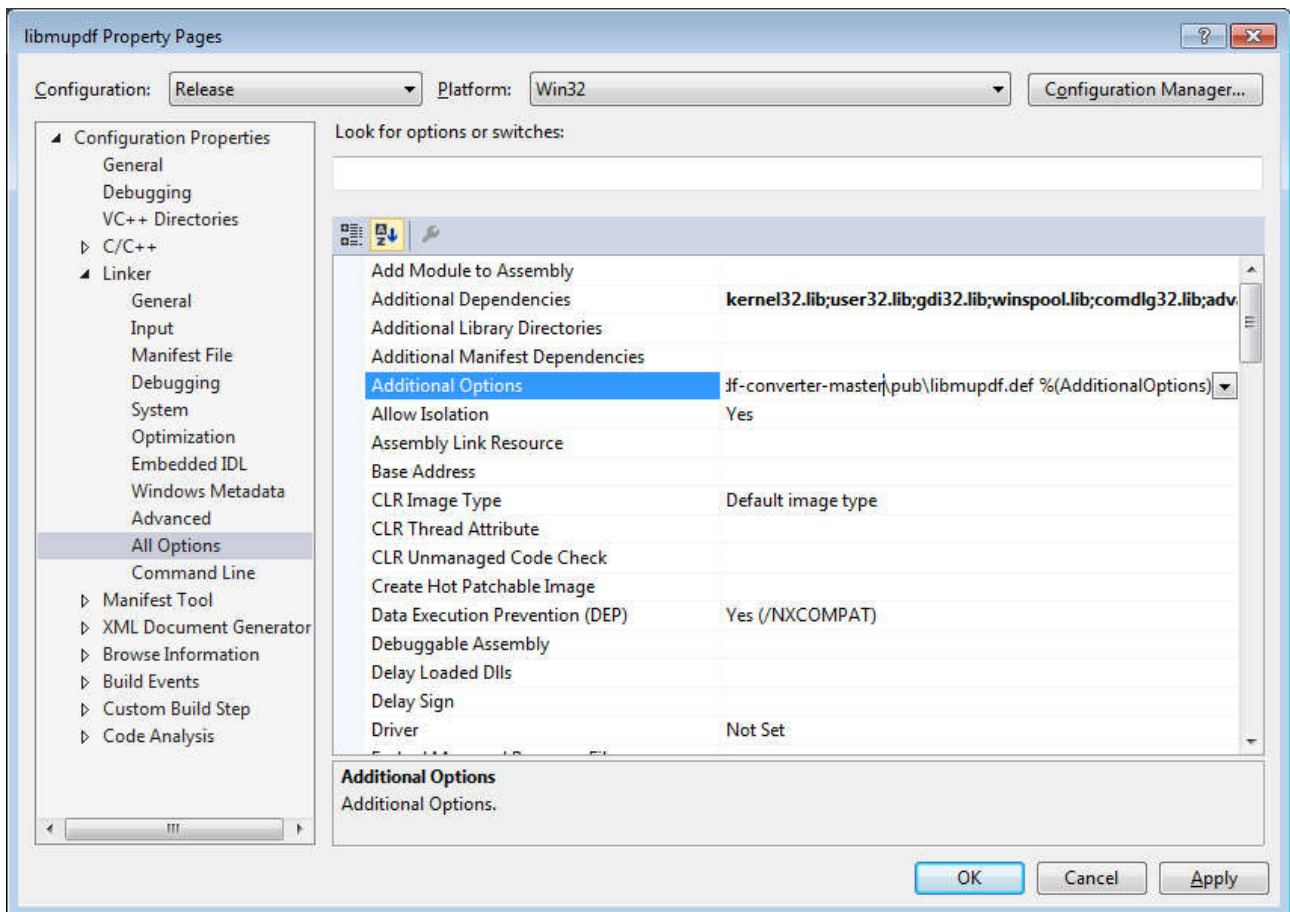
Only functions that are listed here will be visible in libmupdf.dll by some other consumer program.

Example of libmupdf.def file:

; Fitz exports

```
fz_new_context_imp
pdf_document_handler
```





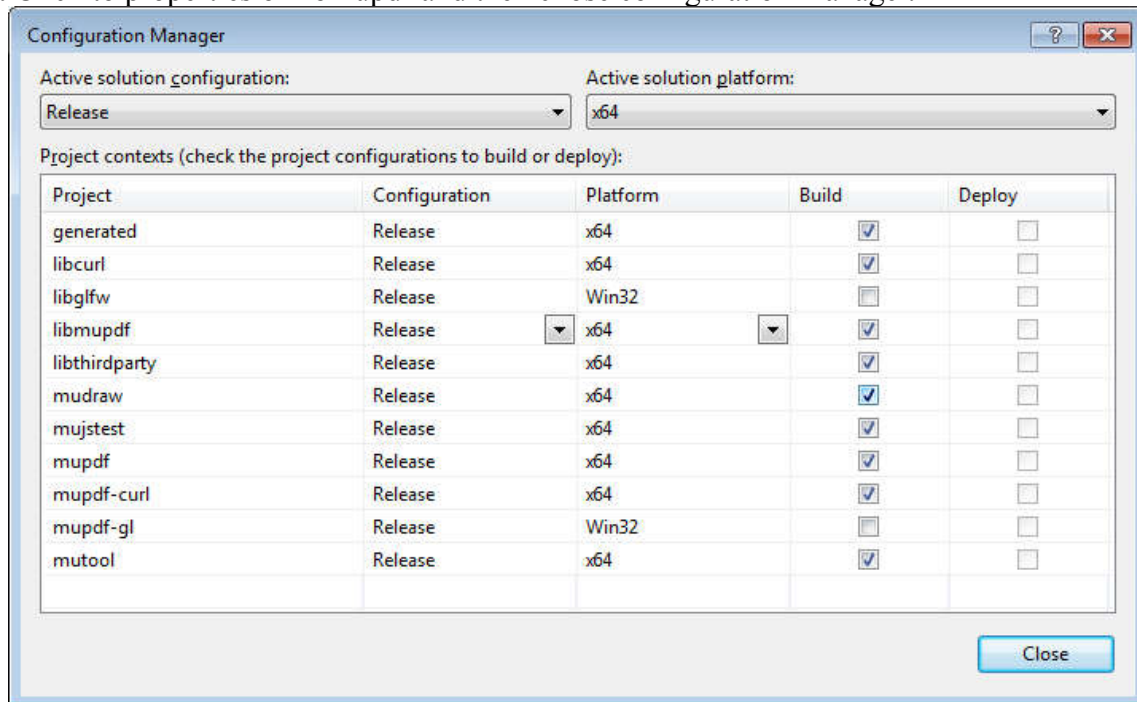
Close all other windows, right click to libmupdf and choose Build.  
After some time you should see something like this:

```
2> Generating Code...
2> Creating library Release\libmupdf.lib and object Release\libmupdf.exp
2> Generating code
2> All 2655 functions were compiled because no usable IPDB/IOBJ from previous compilation
was found.
2> Finished generating code
2> libmupdf.vcxproj -> g:\mupdf-1.8-source\platform\win32\Release\libmupdf.dll
2> libmupdf.vcxproj -> Release\libmupdf.pdb (Full PDB)
===== Build: 2 succeeded, 0 failed, 0 up-to-date, 0 skipped =====
```

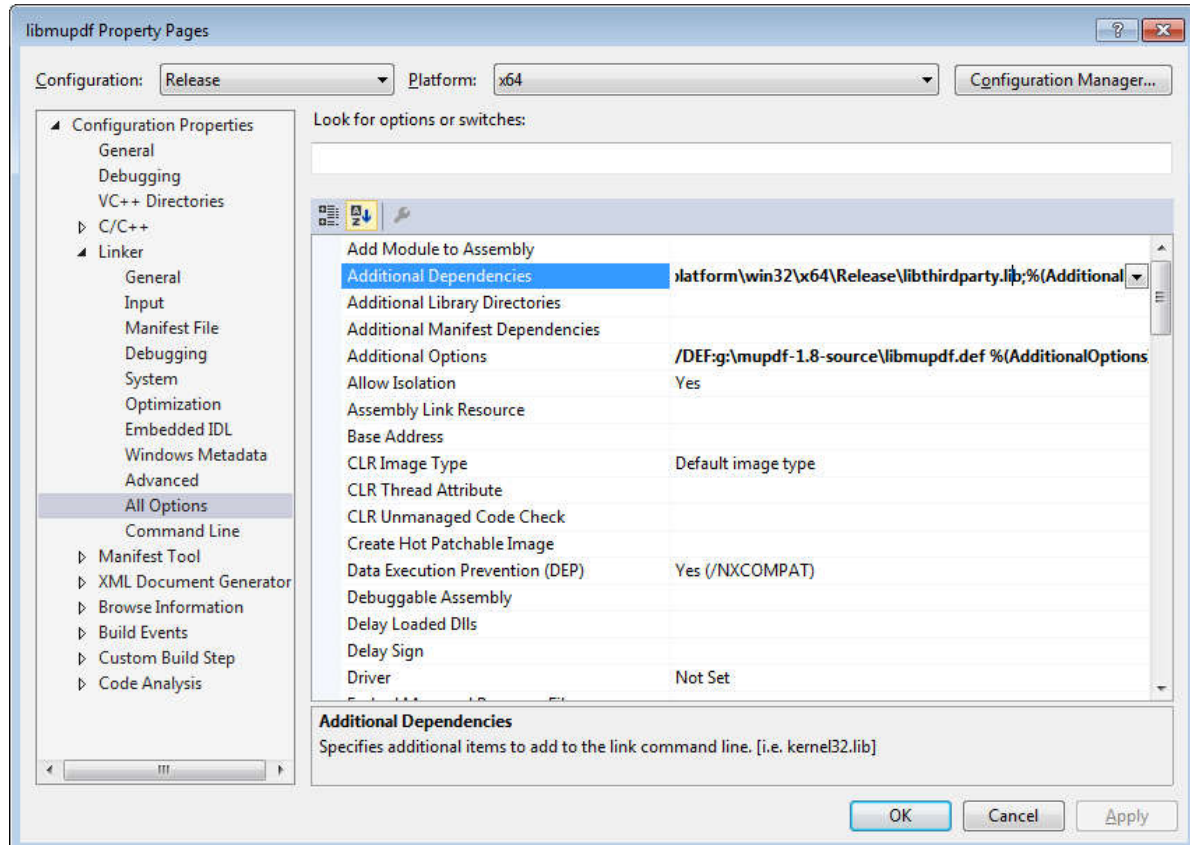
The final file is located in g:\mupdf-1.8-source\platform\win32\Release\libmupdf.dll  
Now take this file and rename it too libmupdf18-32.dll

## Lets build win64 dll.

Right Click to properties of libmupdf and then chose configuration manager.



Change to Release and x64.



kernel32.lib;user32.lib;gdi32.lib;winspool.lib;comdlg32.lib;advapi32.lib;shell32.lib;ole32.lib;oleaut



Change everything like you did for 32 version. Be carefull. Release of library in now in different directory:

32.lib;uuid.lib;odbc32.lib;odbccp32.lib;g:\mupdf-1.8-source\platform\win32\x64\Release\libthirdparty.lib;%(AdditionalDependencies)

Change Additional options like we did in 32 version.

/DEF:g:\mupdf-1.8-source\libmupdf.def %(AdditionalOptions)

Windows and static linking did not workout for me. I always got the error while linking. Something like Wrong COFF magic number. This error is the same no matter I compile with mingw or visual studio 2015.

Compilation of mupdf 1.7 did not workout for me because Visual Studio 2010 i required, or Platform Tools 2010 for Visual Studio 2015. I was not able to find Platform Tools.

## ***Content of file libmupdf.def (function names)***

LIBRARY libmupdf

EXPORTS

; Fitz exports variables

pdf\_document\_handler  
xps\_document\_handler  
cbz\_document\_handler  
img\_document\_handler  
tiff\_document\_handler  
html\_document\_handler  
epub\_document\_handler

; Fitz exports functions

fz\_new\_context\_imp  
fz\_register\_document\_handler  
fz\_register\_document\_handlers  
fz\_new\_document\_handler\_context  
fz\_drop\_document\_handler\_context








; fz\_free\_context is changed to fz\_drop\_context  
; fz\_close\_document is changed to fz\_drop\_document  
; fz\_free\_device is changed to fz\_drop\_device  
; fz\_free\_page is changed to fz\_drop\_page

fz\_open\_document  
fz\_drop\_document  
fz\_drop\_context  
fz\_drop\_device  
fz\_count\_pages  
fz\_load\_page  
fz\_drop\_page  
fz\_load\_page  
fz\_bound\_page  
fz\_run\_page  
fz\_new\_draw\_device  
fz\_rotate  
fz\_pre\_scale  
fz\_lookup\_device\_colorspace  
fz\_pixmap\_bbox  
fz\_pixmap\_width  
fz\_pixmap\_height  
fz\_new\_pixmap  
fz\_new\_pixmap\_with\_bbox  
fz\_new\_pixmap\_with\_data  
fz\_new\_pixmap\_with\_bbox\_and\_data  
fz\_keep\_pixmap

fz\_drop\_pixmap  
fz\_pixmap\_colorspace  
fz\_pixmap\_components  
fz\_pixmap\_samples  
fz\_clear\_pixmap\_with\_value  
fz\_clear\_pixmap  
fz\_new\_trace\_device  
fz\_new\_bbox\_device  
fz\_new\_draw\_device\_with\_bbox  
fz\_needs\_password  
fz\_authenticate\_password  
fz\_transform\_rect

Activities PDFPreviewLin64-18 Wed 17:11 hr 🔊 🔌 🔌

PDF Preview

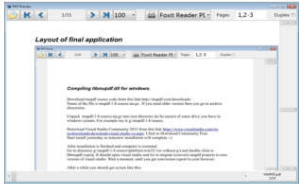
   1/23   Page  FS-4020DN  Pages 1,2-3 ☐ Duplex

### PDFPreview

**Purpose**

This project uses mupdf library to generate preview and printout of pdf documents.  
The purpose of this document is to explain how to build libmupdf.dll (32 and 64 bit) for windows, how to link it in FPC, Delphi or other platform application.  
Example project shows the use of library with common preview functions. You are encourage to extend this project. Project should work with linux who has it is not tested.  
Initial idea for this FPC project is done by Blietum Tabular. His version use one of first version of libmupdf library.  
Problems:  
There are lot of changes in color function calls between versions.  
Not every version of libmupdf.dll is available.

**Layout of final application**



**History**

My first approach was to use ver 1.6 of libmupdf. I did not found any precompiled version of it. According to Blietum, Sumatra project always have libmupdf.dll if you use installer and install sumatra. So I did, but guys from Sumatra did not export all function needed by Blietum example. So it was dead end for me. Then I used Sumatra source code and recompiled libmupdf.dll but included missing function. This was good. Then I tried to precompile for Linux. But Fedora19 did not have

MUPDF.pdf  
1/23

## ***Following is my notest about Fedora 22***

Install Fedora22 (I use 64 bit) on virtual machine. You will need latest vmware (12) virtual machine. After installation you will need to uninstall open-vmware-tools and install regular vmware tools. This is because you want to share drives between windows host and linux virtual machine. I installed codetyphon 5.40. Download it and unpack in user home directory (user can not be root). Run install.sh script, and choose option 0. This option will install all prerequisites needed by typhon64. Then you choose option 8 (Remove and Build All)

<http://www.linuxfromscratch.org/blfs/view/svn/pst/mupdf.html>

build win32

make OS=MINGW build=release

make build=release install

[https://fedoraproject.org/wiki/Common\\_F22\\_bugs#No\\_network\\_connection\\_in\\_VM\\_when\\_both\\_host\\_and\\_guest\\_installed\\_from\\_a\\_Live\\_image](https://fedoraproject.org/wiki/Common_F22_bugs#No_network_connection_in_VM_when_both_host_and_guest_installed_from_a_Live_image)

sudo virsh net-destroy default

sudo virsh net-undefine default

## Installing Fedora23 – 32 on VMWare12.

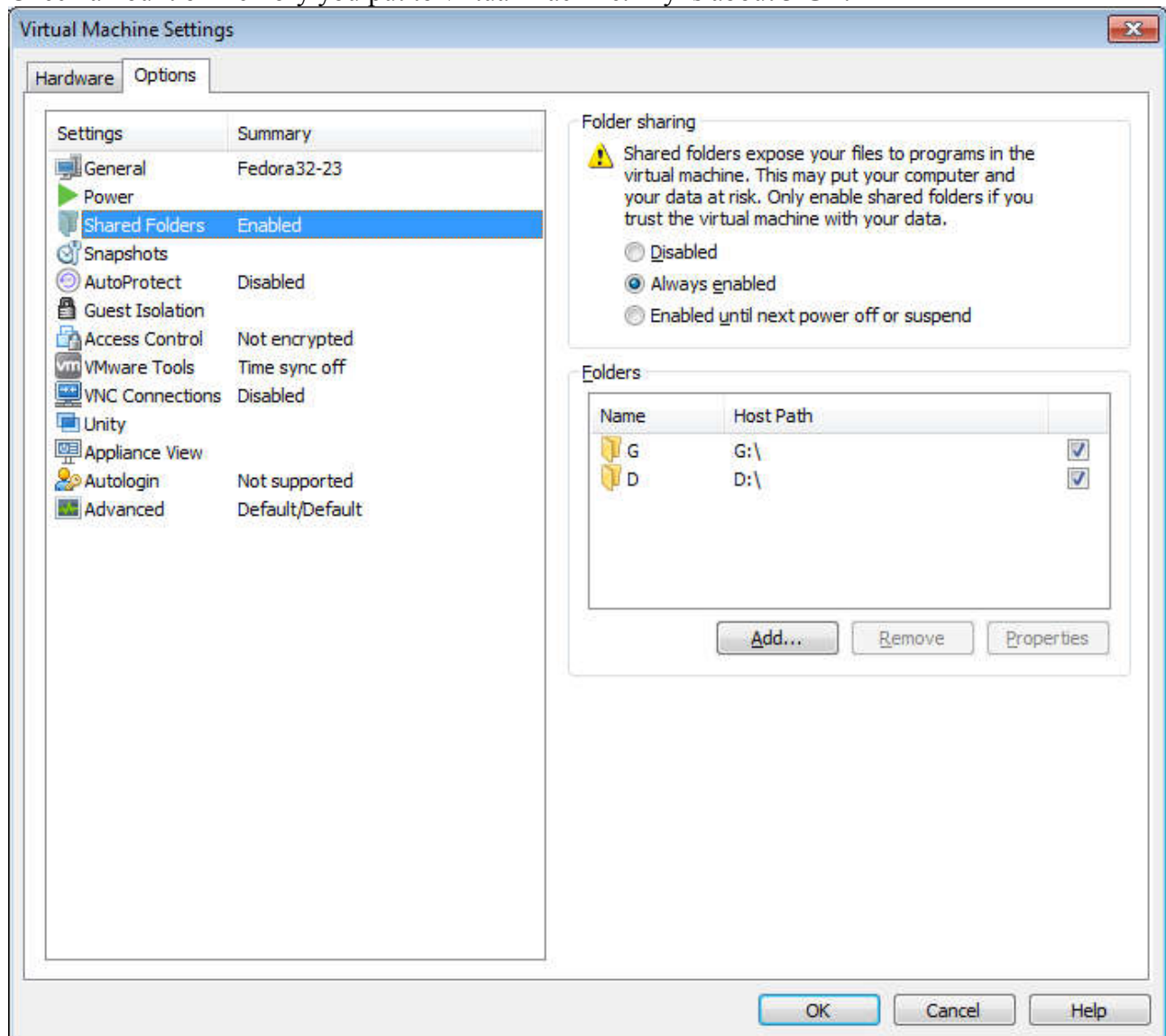
### Installation procedure:

Choose installation image and install.

After installation

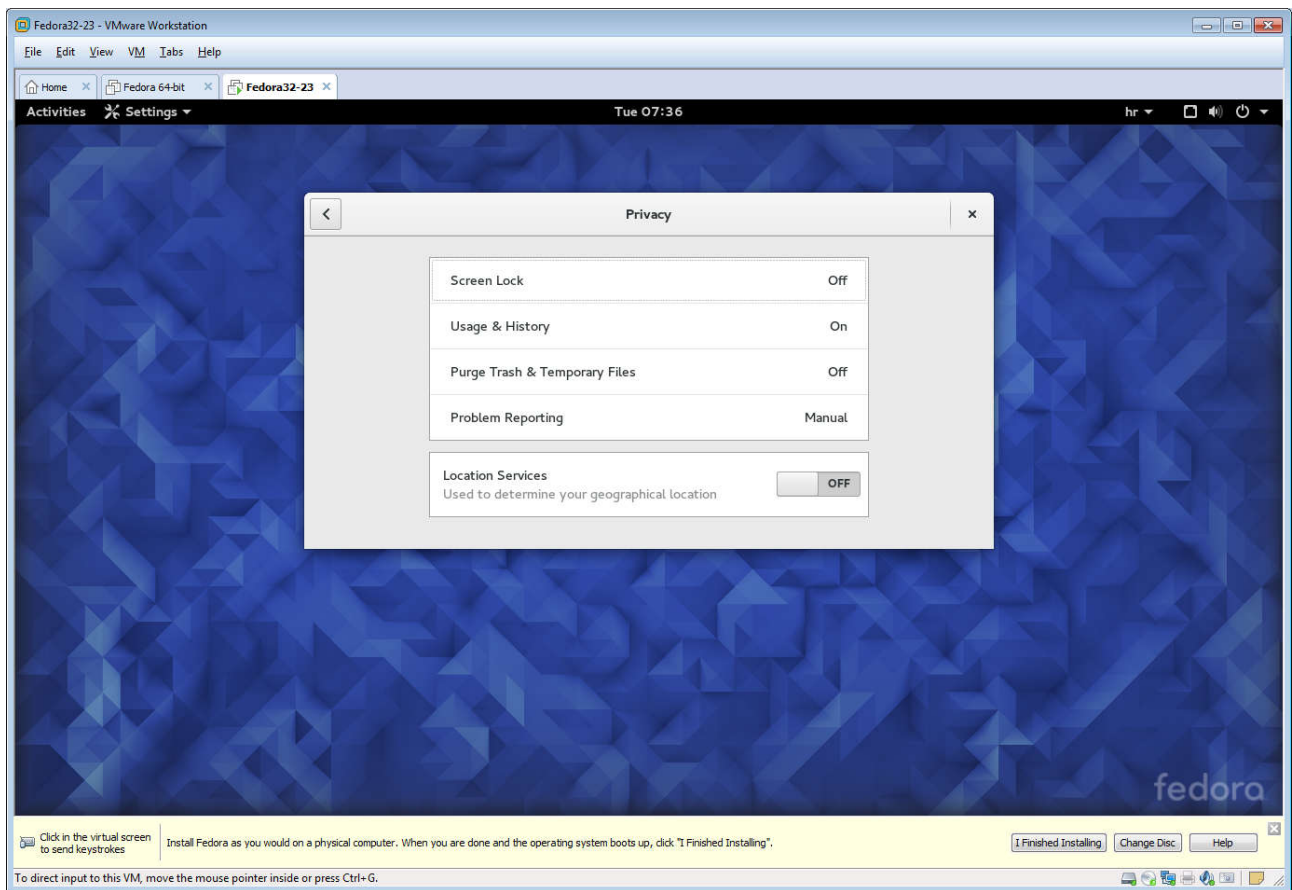
Put shared disk on: G and D in vmware Settings/Options/Shared Folders

Check amount of memory you put to virtual machine. My is about 3 GB.

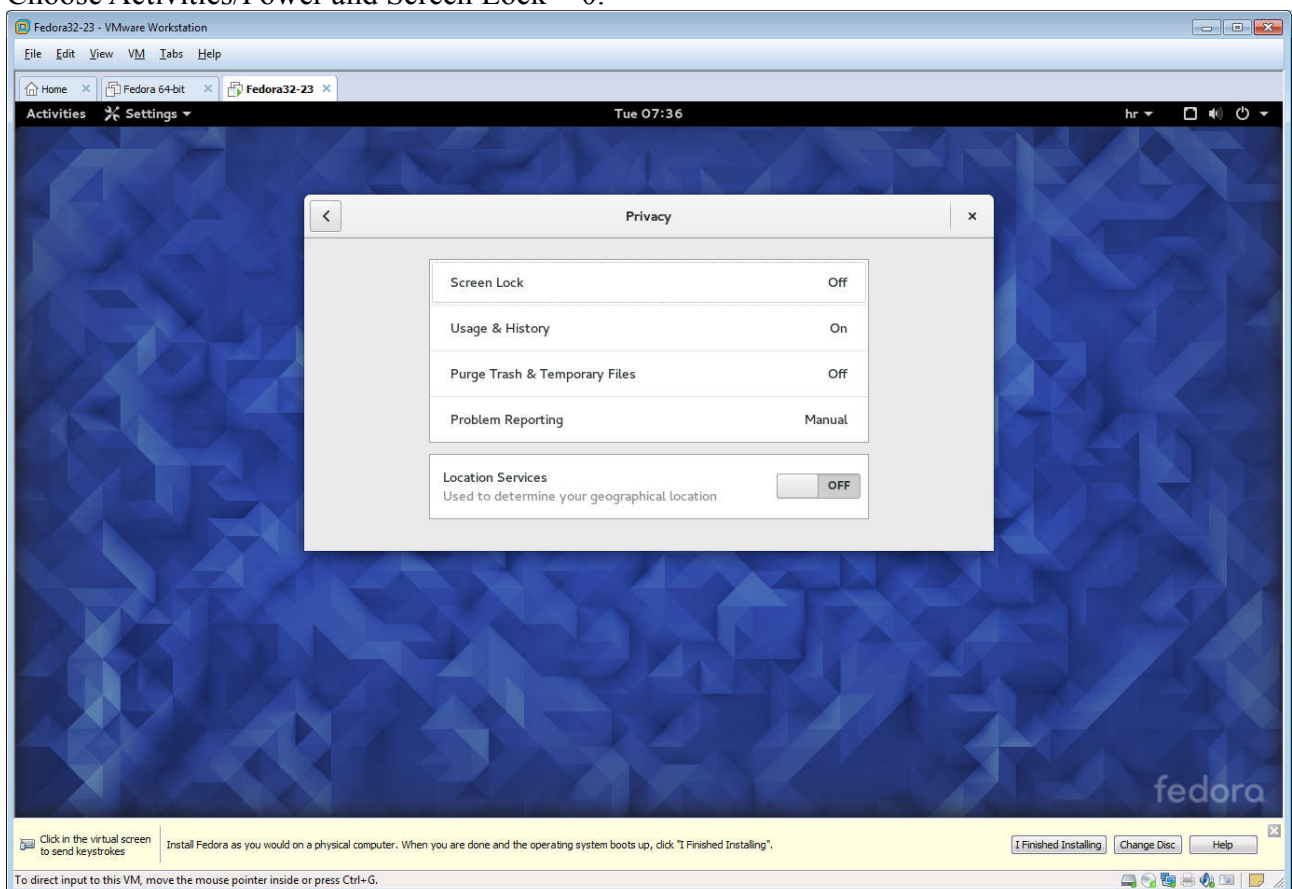


If Fedora chose Activity and type Privacy. Choose Screen Lock = Off

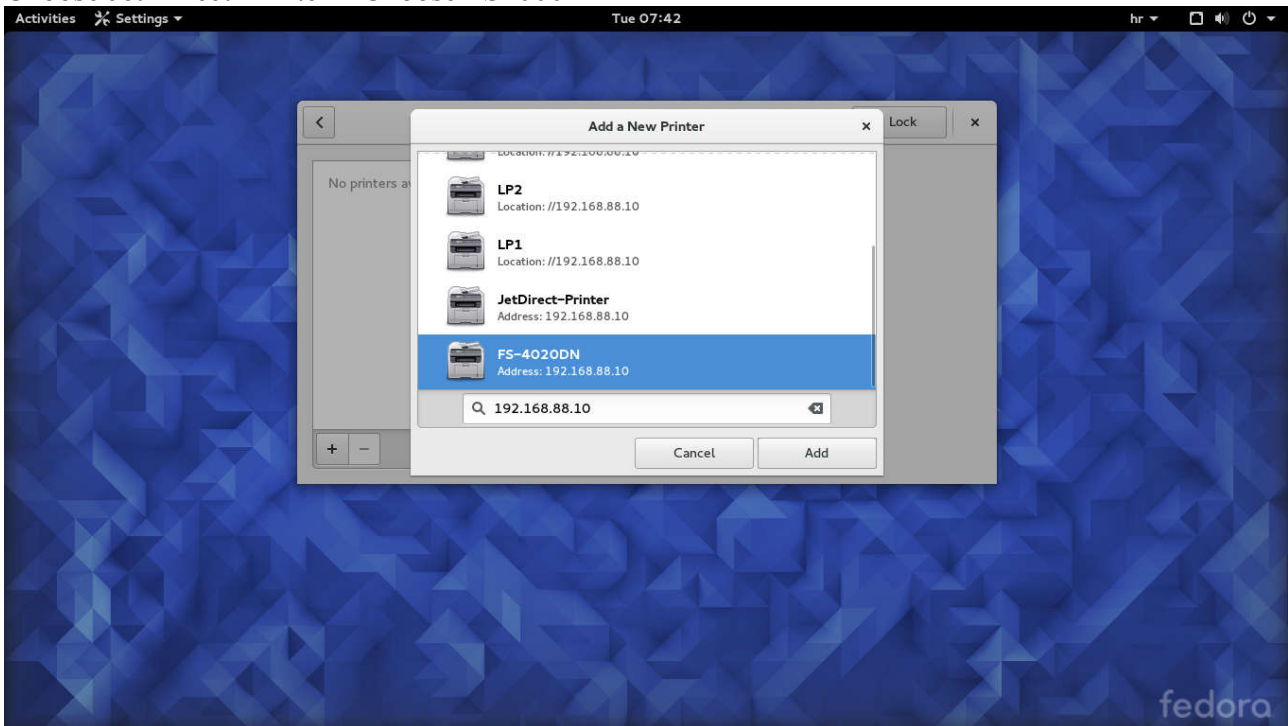




Choose Activities/Power and Screen Lock = 0.

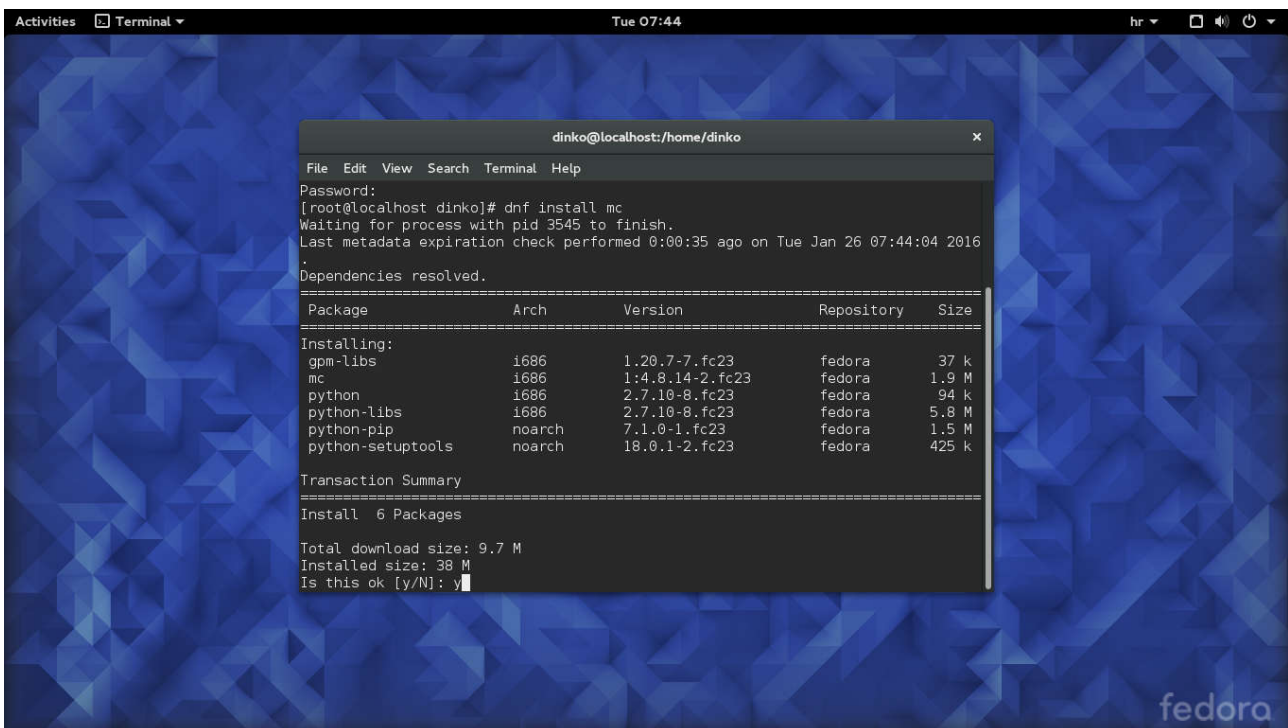


Choose Activity and type Printers. Click Unlock and type password.  
Install printer by typing ip address 192.168.88.10  
Choose Jet Direct-Printer – Choose FS4000



Choose FS-4020DN as driver

Install mc  
Open Terminal. Activity type terminal.  
Type:  
su and password  
dnf install mc



Enabling vmware shares:  
 uninstall open-vmware-tools  
 Type:  
 su and password  
 dnf remove open-vm-tools

```

dinko@localhost:/home/dinko
File Edit View Search Terminal Help
Running transaction
Erasing      : open-vm-tools-desktop-10.0.0-7.fc23.i686      1/7
Erasing      : open-vm-tools-10.0.0-7.fc23.i686              2/7
Erasing      : xml-security-c-1.6.1-10.fc23.i686             3/7
Erasing      : xalan-c-1.11.0-7.fc23.i686                    4/7
Erasing      : xerces-c-3.1.2-3.fc23.i686                    5/7
Erasing      : libdnet-1.12-16.fc23.i686                      6/7
Erasing      : gtkmm24-2.24.4-7.fc23.i686                    7/7
Verifying    : open-vm-tools-10.0.0-7.fc23.i686              1/7
Verifying    : open-vm-tools-desktop-10.0.0-7.fc23.i686     2/7
Verifying    : xerces-c-3.1.2-3.fc23.i686                    3/7
Verifying    : gtkmm24-2.24.4-7.fc23.i686                    4/7
Verifying    : libdnet-1.12-16.fc23.i686                      5/7
Verifying    : xml-security-c-1.6.1-10.fc23.i686             6/7
Verifying    : xalan-c-1.11.0-7.fc23.i686                    7/7

Removed:
gtkmm24.i686 2.24.4-7.fc23      libdnet.i686 1.12-16.fc23
open-vm-tools.i686 10.0.0-7.fc23 open-vm-tools-desktop.i686 10.0.0-7.fc23
xalan-c.i686 1.11.0-7.fc23      xerces-c.i686 3.1.2-3.fc23
xml-security-c.i686 1.6.1-10.fc23

Complete!
[root@localhost dinko]#
  
```

Go to vmware VM menu of host application and chose VM/Install VMWare Tools. Wait until tools are mounted in Fedora 23 file system.

Goto Fedora22 and open mc in terminal. Goto /run/media/dinko directory.

If not succesfull (not exists) then restart Fedora23 and try again the same procedure.

Inside directory should be VMWare Tools disk with VMWare-Tools 10.0.5-3228523 tar gzip file.

Unzip this file in your home directory by using mc. Just enter into tar gzip by pressing Enter.

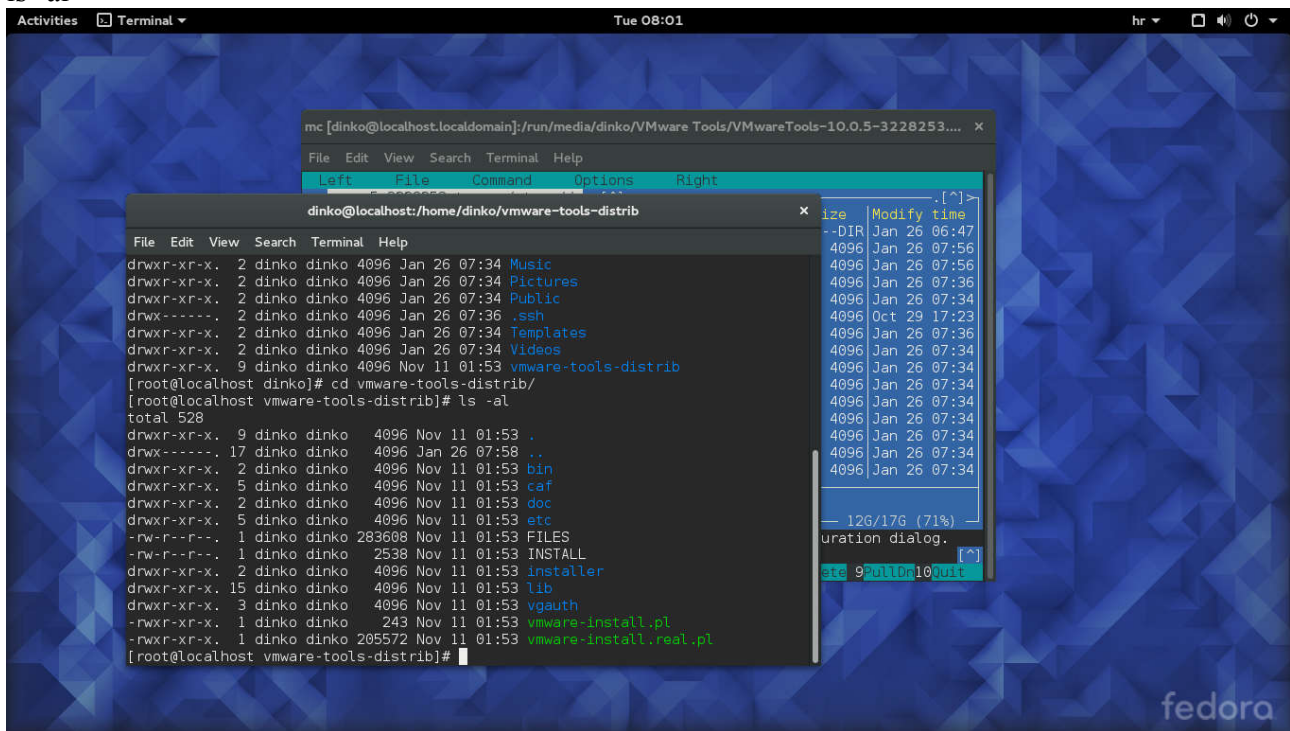
```

mc [dinko@localhost.localdomain]:/run/media/dinko/VMware Tools
File Edit View Search Terminal Help
Left: File Command Options Right
.n Name Size Modify time .n Name Size Modify time
/.. UP--DIR Jan 26 07:53 /.. UP--DIR Jan 26 06:47
VMwareTools.tar.gz 76577K Nov 11 01:54 /.cache 4096 Jan 26 07:56
*manifest.txt 1994 Nov 11 01:53 /.config 4096 Jan 26 07:56
*run_upgrader.sh 1850 Nov 11 01:32 /.gnupg 4096 Jan 26 07:36
*vmware-t-ader-32 687524 Nov 11 01:34 /.local 4096 Jan 26 07:34
*vmware-t-ader-64 757944 Nov 11 01:37 /.mozilla 4096 Oct 29 17:23
/ssh 4096 Jan 26 07:36
/Desktop 4096 Jan 26 07:34
/Documents 4096 Jan 26 07:34
/Downloads 4096 Jan 26 07:34
/Music 4096 Jan 26 07:34
/Pictures 4096 Jan 26 07:34
/Public 4096 Jan 26 07:34
/Templates 4096 Jan 26 07:34
/Videos 4096 Jan 26 07:34

UP--DIR 0/71M (0%) UP--DIR 126/17G (71%)
Hint: Want your plain shell? Press C-o, and get back to MC with C-o again.
[dinko@localhost VMware Tools]$
1?help 2?menu 3?view 4?edit 5?copy 6?renMov 7?mkdir 8?delete 9?ulldir 10?quit
  
```

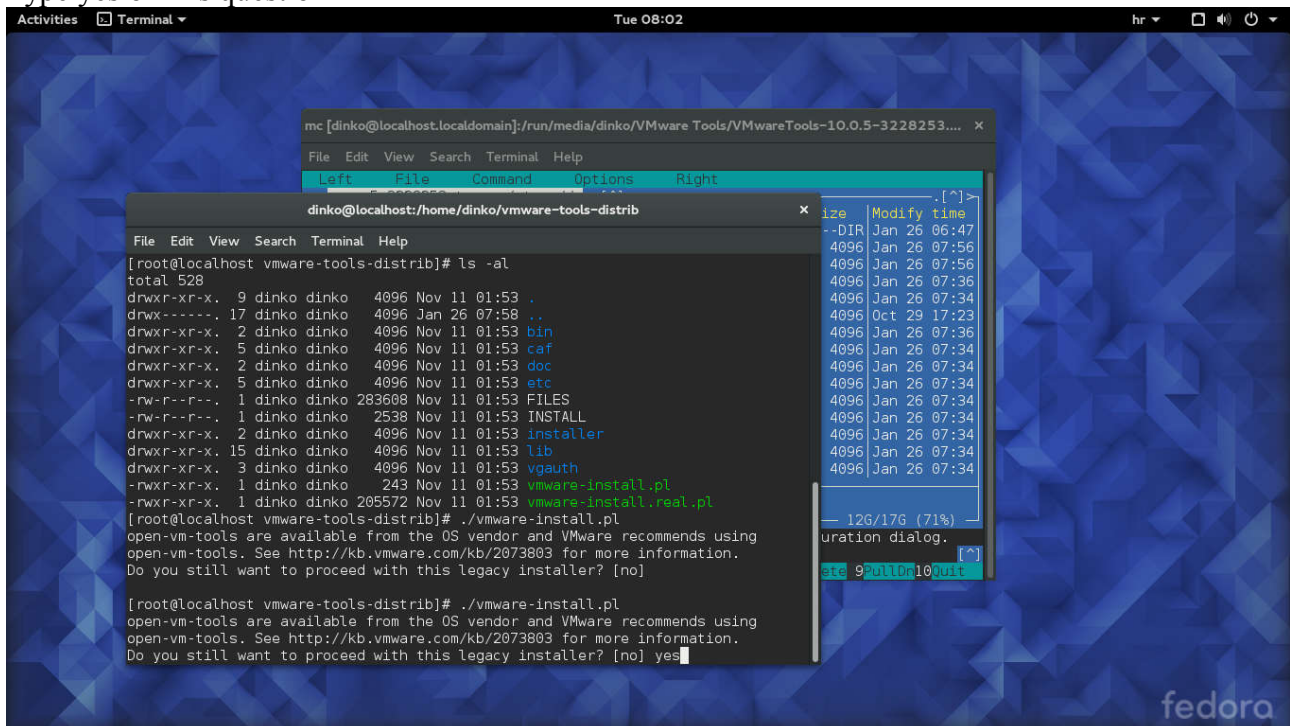


Open another terminal  
type:  
su and password  
cd vm and press tab then enter  
ls -al



```
mc [dinko@localhost.localdomain]:/run/media/dinko/VMware Tools/VMwareTools-10.0.5-3228253.... x
File Edit View Search Terminal Help
Left File Command Options Right
dinko@localhost:/home/dinko/vmware-tools-distrib x
File Edit View Search Terminal Help
drwxr-xr-x. 2 dinko dinko 4096 Jan 26 07:34 Music
drwxr-xr-x. 2 dinko dinko 4096 Jan 26 07:34 Pictures
drwxr-xr-x. 2 dinko dinko 4096 Jan 26 07:34 Public
drwx----- 2 dinko dinko 4096 Jan 26 07:36 .ssh
drwxr-xr-x. 2 dinko dinko 4096 Jan 26 07:34 Templates
drwxr-xr-x. 2 dinko dinko 4096 Jan 26 07:34 Videos
drwxr-xr-x. 9 dinko dinko 4096 Nov 11 01:53 vmware-tools-distrib
[root@localhost dinko]# cd vmware-tools-distrib/
[root@localhost vmware-tools-distrib]# ls -al
total 528
drwxr-xr-x. 9 dinko dinko 4096 Nov 11 01:53 .
drwx----- 17 dinko dinko 4096 Jan 26 07:58 ..
drwxr-xr-x. 2 dinko dinko 4096 Nov 11 01:53 bin
drwxr-xr-x. 5 dinko dinko 4096 Nov 11 01:53 caf
drwxr-xr-x. 2 dinko dinko 4096 Nov 11 01:53 doc
drwxr-xr-x. 5 dinko dinko 4096 Nov 11 01:53 etc
-rw-r--r-- 1 dinko dinko 283608 Nov 11 01:53 FILES
-rw-r--r-- 1 dinko dinko 2538 Nov 11 01:53 INSTALL
drwxr-xr-x. 2 dinko dinko 4096 Nov 11 01:53 installer
drwxr-xr-x. 15 dinko dinko 4096 Nov 11 01:53 lib
drwxr-xr-x. 3 dinko dinko 4096 Nov 11 01:53 vgauth
-rwxr-xr-x. 1 dinko dinko 243 Nov 11 01:53 vmware-install.pl
-rwxr-xr-x. 1 dinko dinko 205572 Nov 11 01:53 vmware-install.real.pl
[root@localhost vmware-tools-distrib]#
```

type to start installation  
./vmware-install.pl  
Type yes on first question

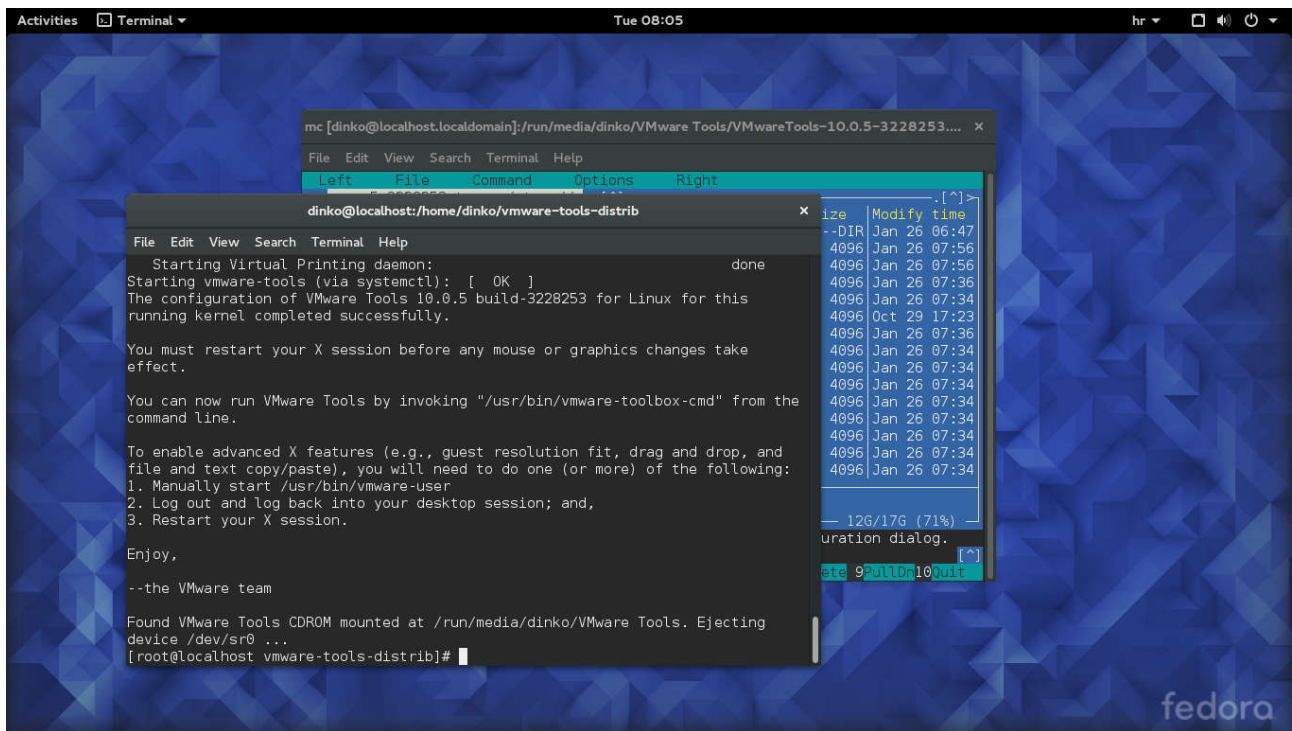


```
mc [dinko@localhost.localdomain]:/run/media/dinko/VMware Tools/VMwareTools-10.0.5-3228253.... x
File Edit View Search Terminal Help
Left File Command Options Right
dinko@localhost:/home/dinko/vmware-tools-distrib x
File Edit View Search Terminal Help
[root@localhost vmware-tools-distrib]# ls -al
total 528
drwxr-xr-x. 9 dinko dinko 4096 Nov 11 01:53 .
drwx----- 17 dinko dinko 4096 Jan 26 07:58 ..
drwxr-xr-x. 2 dinko dinko 4096 Nov 11 01:53 bin
drwxr-xr-x. 5 dinko dinko 4096 Nov 11 01:53 caf
drwxr-xr-x. 2 dinko dinko 4096 Nov 11 01:53 doc
drwxr-xr-x. 5 dinko dinko 4096 Nov 11 01:53 etc
-rw-r--r-- 1 dinko dinko 283608 Nov 11 01:53 FILES
-rw-r--r-- 1 dinko dinko 2538 Nov 11 01:53 INSTALL
drwxr-xr-x. 2 dinko dinko 4096 Nov 11 01:53 installer
drwxr-xr-x. 15 dinko dinko 4096 Nov 11 01:53 lib
drwxr-xr-x. 3 dinko dinko 4096 Nov 11 01:53 vgauth
-rwxr-xr-x. 1 dinko dinko 243 Nov 11 01:53 vmware-install.pl
-rwxr-xr-x. 1 dinko dinko 205572 Nov 11 01:53 vmware-install.real.pl
[root@localhost vmware-tools-distrib]# ./vmware-install.pl
open-vm-tools are available from the OS vendor and VMware recommends using
open-vm-tools. See http://kb.vmware.com/kb/2073803 for more information.
Do you still want to proceed with this legacy installer? [no] yes
[root@localhost vmware-tools-distrib]#
```

Confirm every question with default by pressing enter. Wait until installation is finished

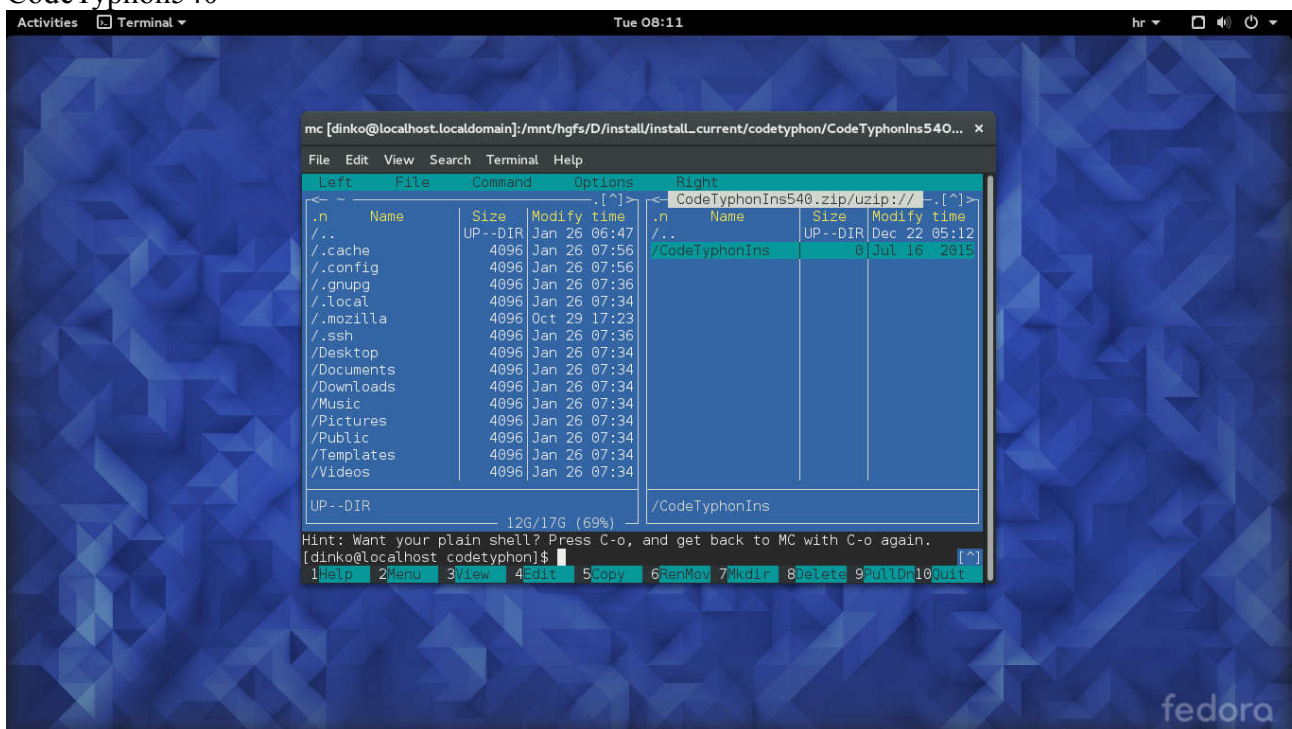






## Restart Fedora 23

Now you should have shared directories from host. Open terminal and find `/mnt/hgfs/G` and `/mnt/hgfs/D` directory. If you find it then you are ready to install codetyphon. I install CodeTyphon540



Find your codetyphon installation on host disks and unzip CodeTyphonIns folder in your home directory. Check this link

1)-Copy, Unzip CodeTyphonIns.zip



## 2)-Install "nano" text editor

```
su and password

dnf install nano
```

## 3)-Give full "sudo" permissions to current user.

```
su

sudo nano /etc/sudoers
```

add in the end of /etc/sudoers the line

```
username          ALL=(ALL) NOPASSWD: ALL
```

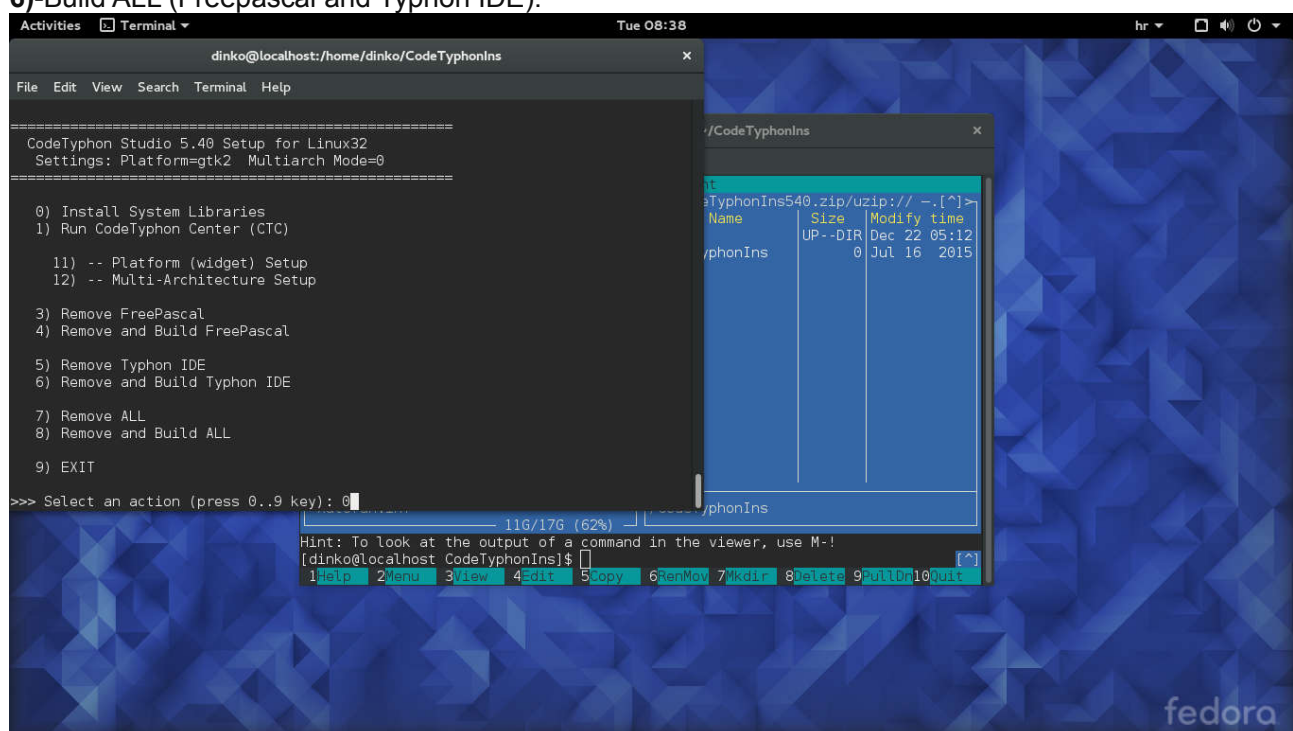
and comment out the following line in /etc/sudoers:

```
#Defaults requiretty - this part I did not found in Fedora 23
```

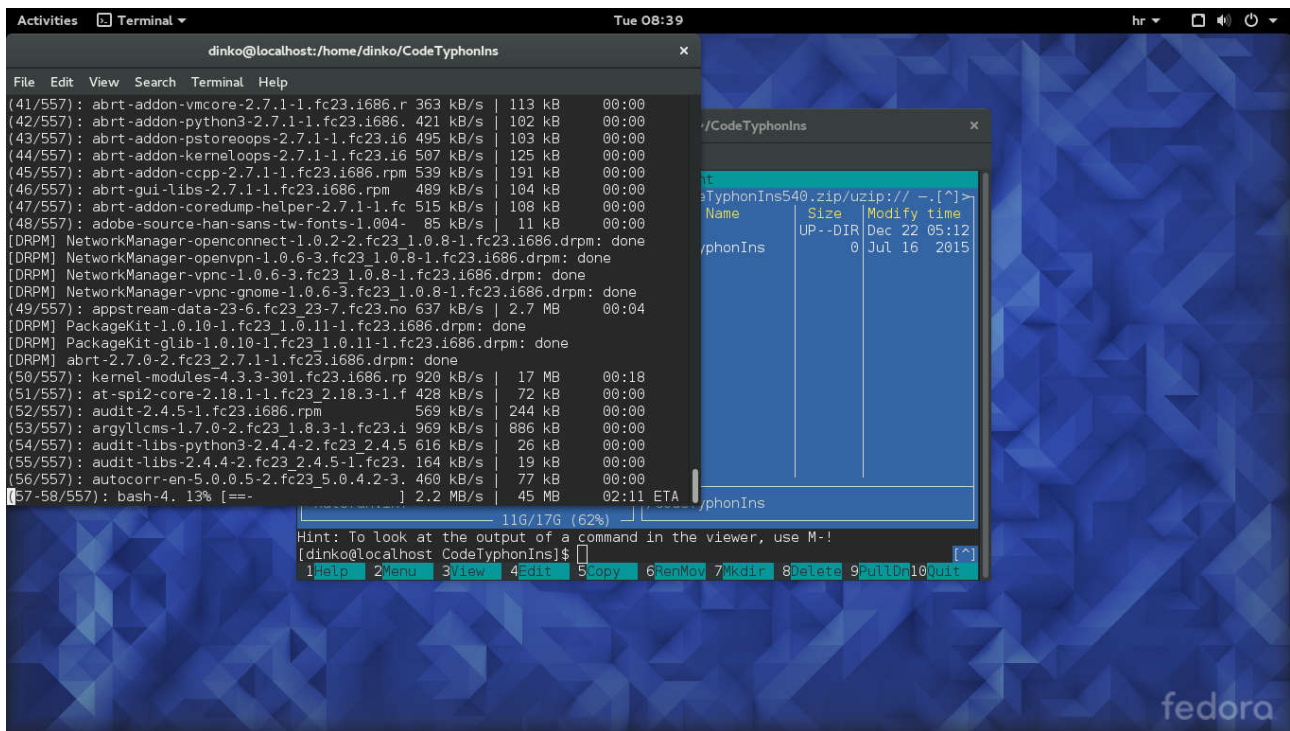
## 4)-Install CodeTyphon

## 5)-Install "System Libraries" from CT setup script. Option 0

## 6)-Build ALL (FreePascal and Typhon IDE).



Installation sometimes fail. Check amount of memory you put to virtual machine. My is about 3 GB.



After installation is finished type  
 dnf install qt4pas-devel-2.5-7.fc23.i686

Do not install dnf install mupdf-devel-1.7a-4.fc23.i686  
 Do not install dnf install mupdf-1.7a-4.fc23.i686

## ***Duplex Printing of Windows***

This was really big problem. I was unable to set duplex printing on windows for 2 days. Now I have solution, but this solution change global DUPLEX print variable on printer driver (and after print it puts it back to original settings). I think that it will be better solution that I installed one driver with SIMPLEX setting and other one with DUPLEX settings, but this is not good solution for me because I have many terminals and I have to go to each and every one and set printers.

Windows solution is in file WinPrinterUtils.pas

Linux duplex printing is not set at all.

Reference:

<https://support.microsoft.com/en-us/kb/167345>

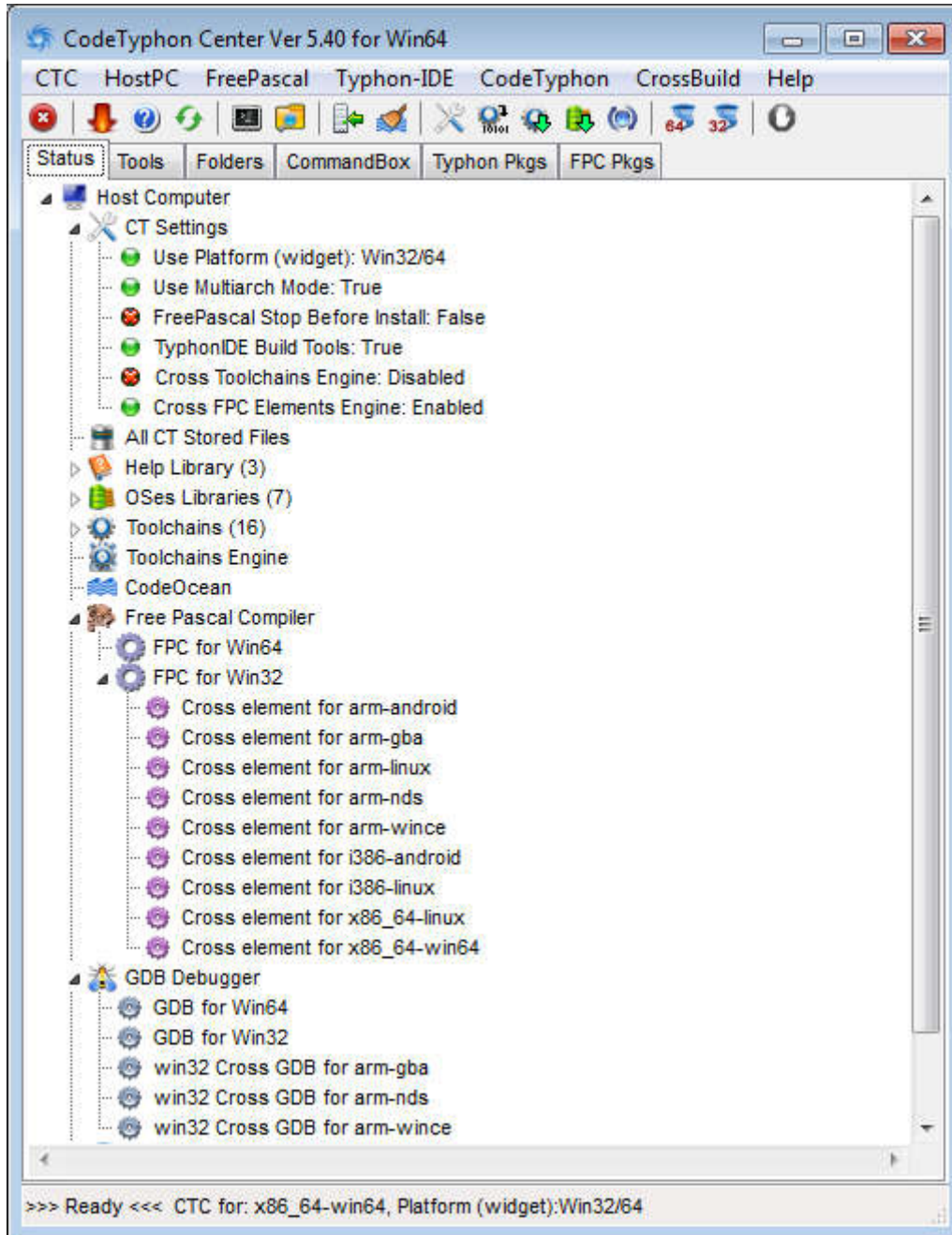
Also I found some printersettings.zip on lazarus forum.

<http://forum.lazarus.freepascal.org/index.php/topic,15444.msg83670.html#msg83670>

# Setting CodeTyphon for cross compiling

This is easy with CodeTyphon, but it is long process.

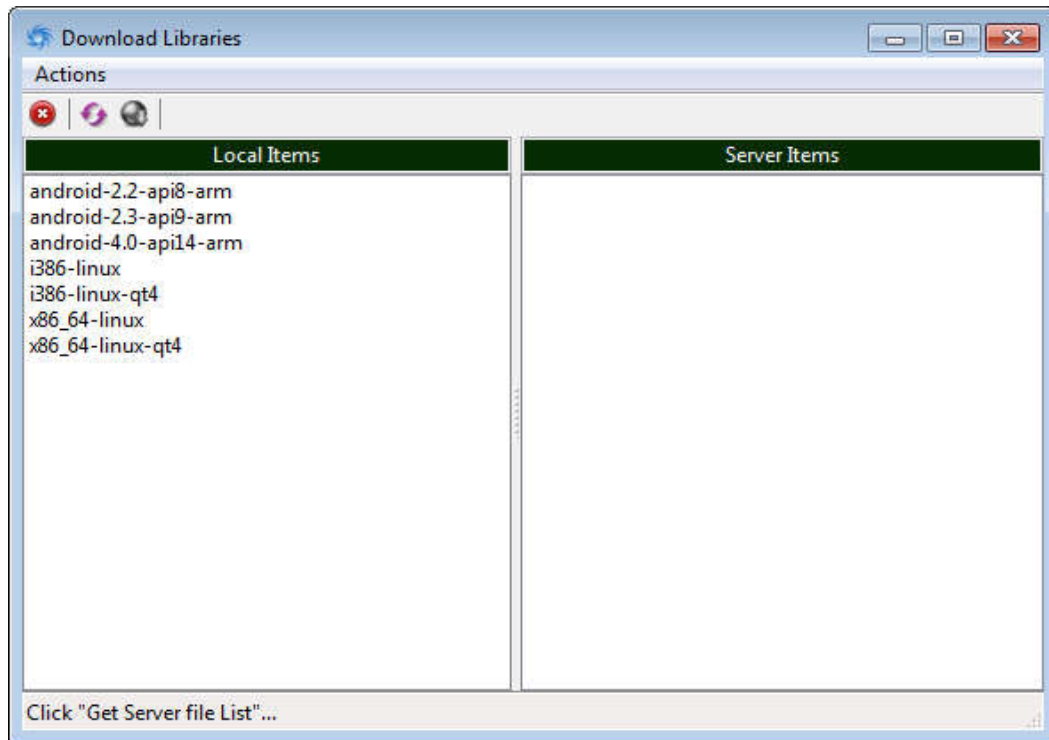
Install CodeTyphon for windows as usual. I have 64 bit machine, so after installation i start CodeTyphon64.



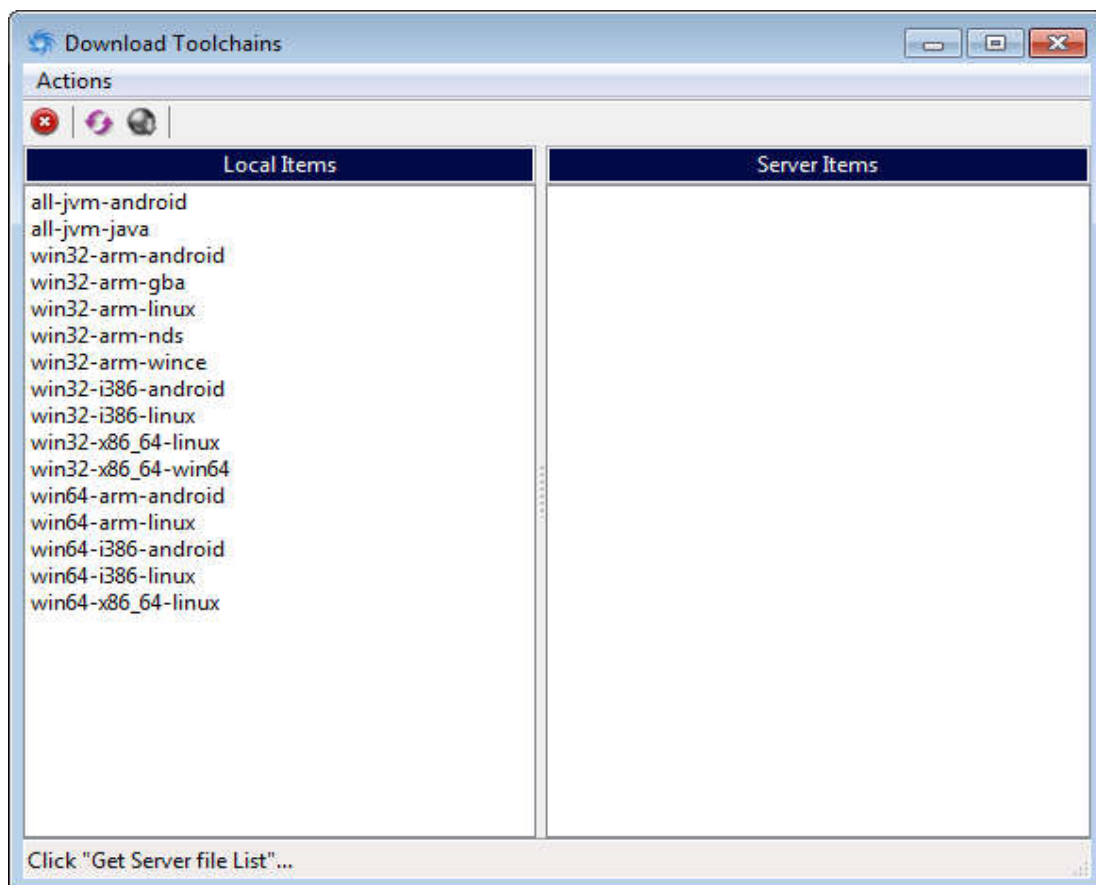
Next step is to setup Typhon32 cross compilers.

Goto Menu/Cross Build/FreePascal 32bits/FPC32 Build Cross Element All and wait for program to finish (1 to 2 hours)

Next is to download necessary libraries. Goto to Menu/Cross Build/Download Libraries. Then refresh button and choose i386-linux, i386-linuxqt4, x86\_64-linux, x86\_64-linux-qt4. After program finish you get your libraries in directories under c:\codetyphon\binLibraries\.



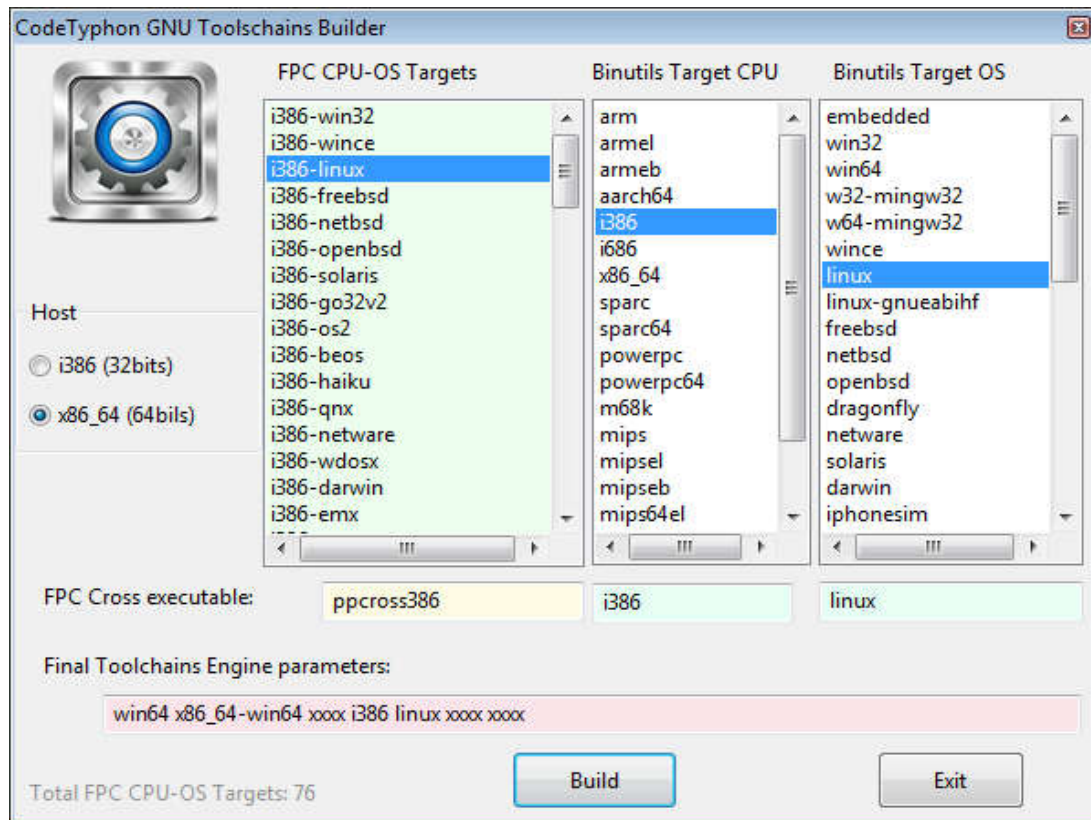
Now we need toolchain. Goto to Menu/Cross Build/Download Toolchains



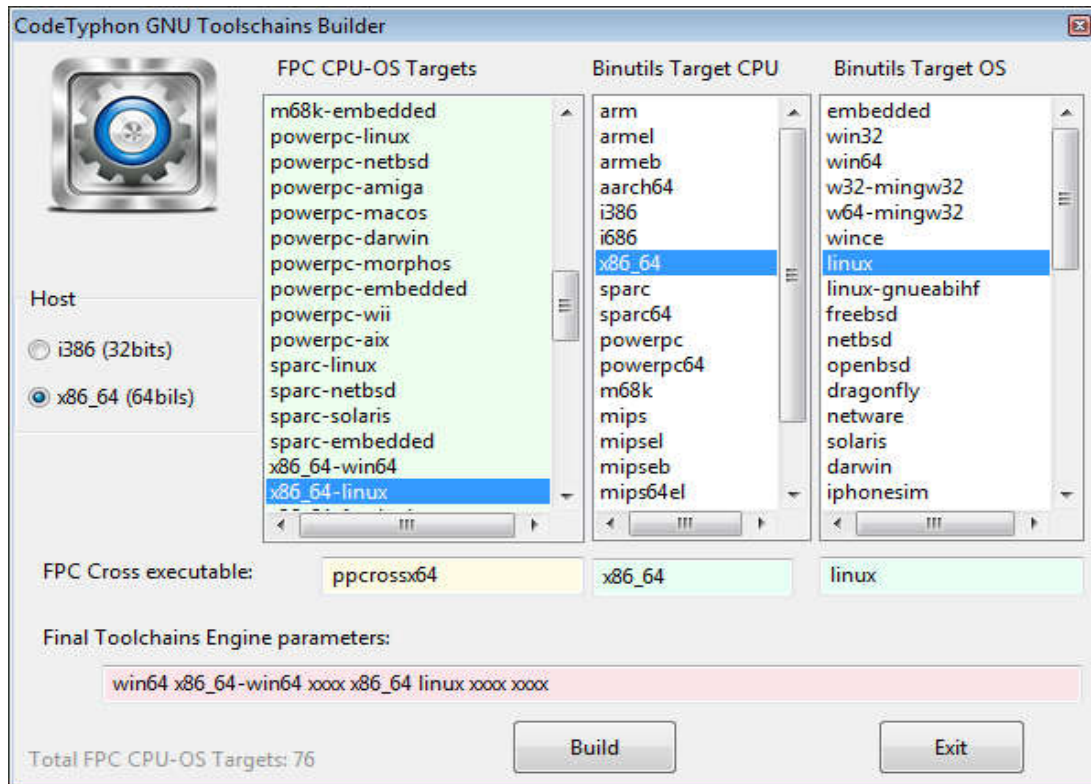
After that you need to build toolchains. Goto to Menu/Cross Build/Toolchain Builder



and choose your build configuration



We need "i386-linux and i386 and linux" and "x86\_64-linux and x86\_64 and linux"

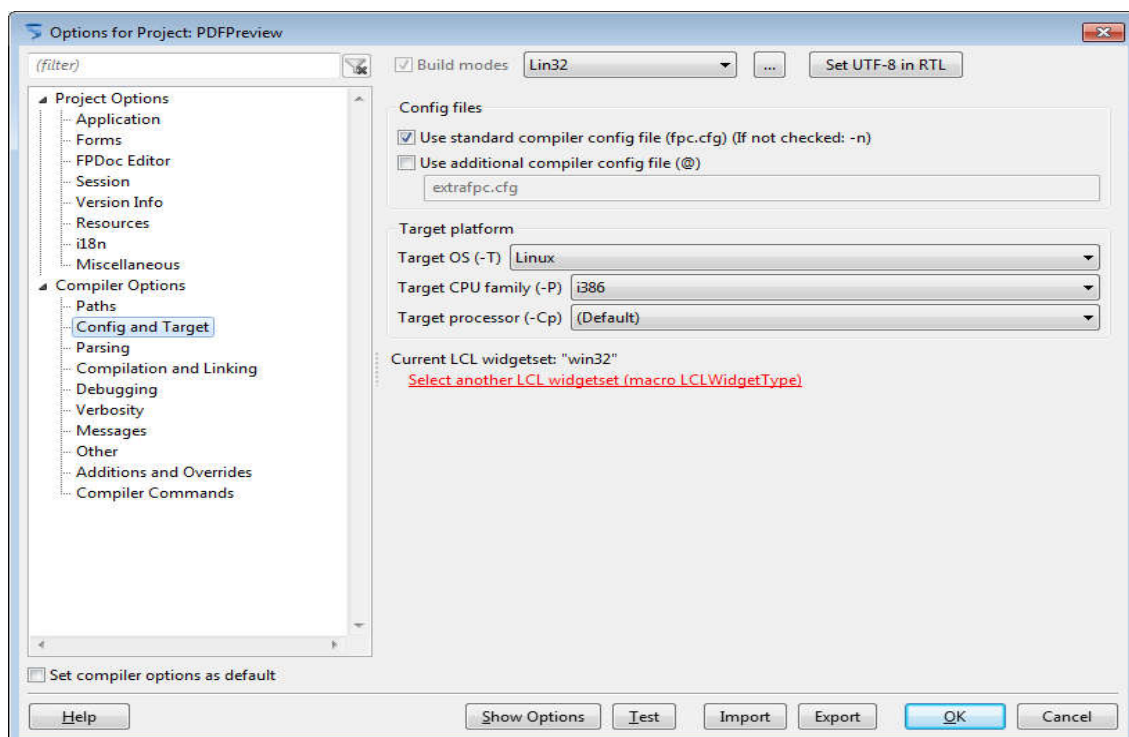
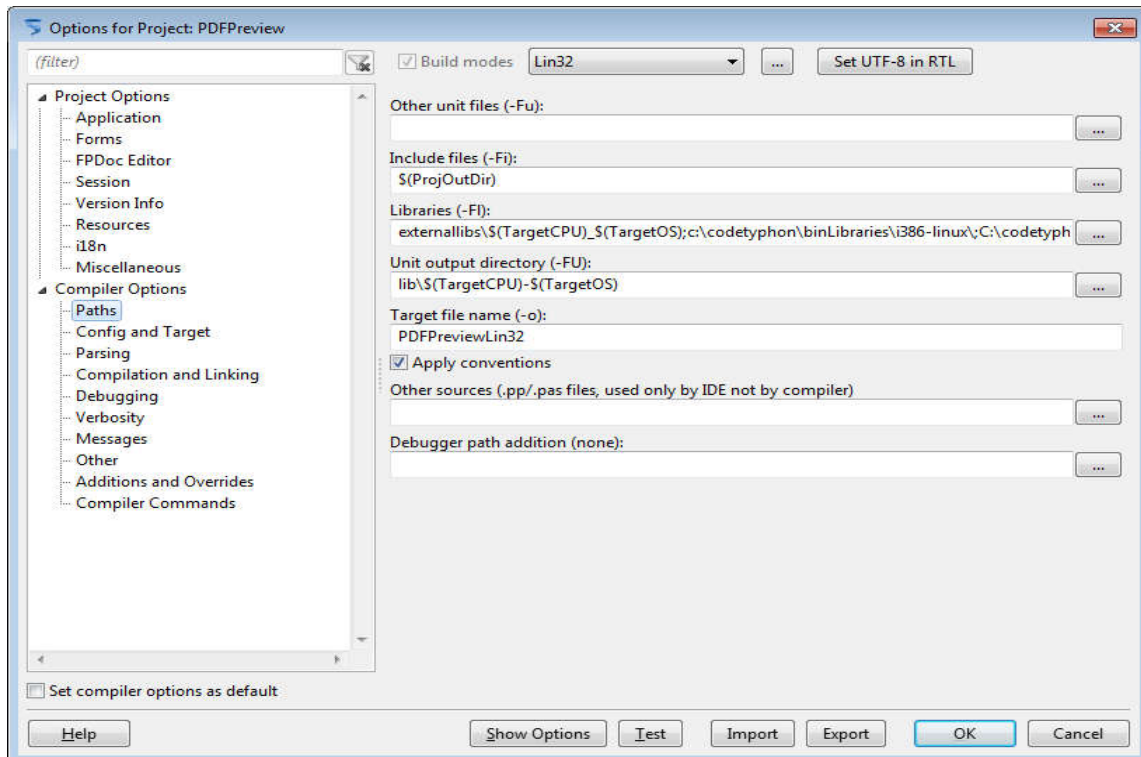


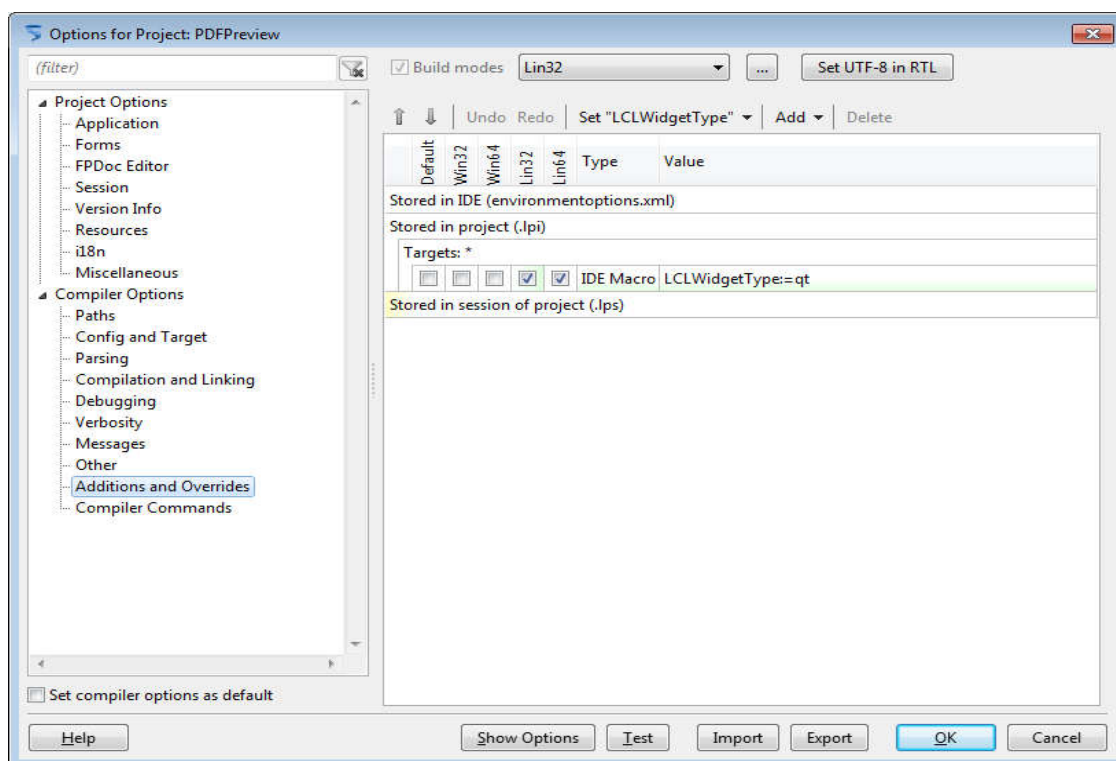
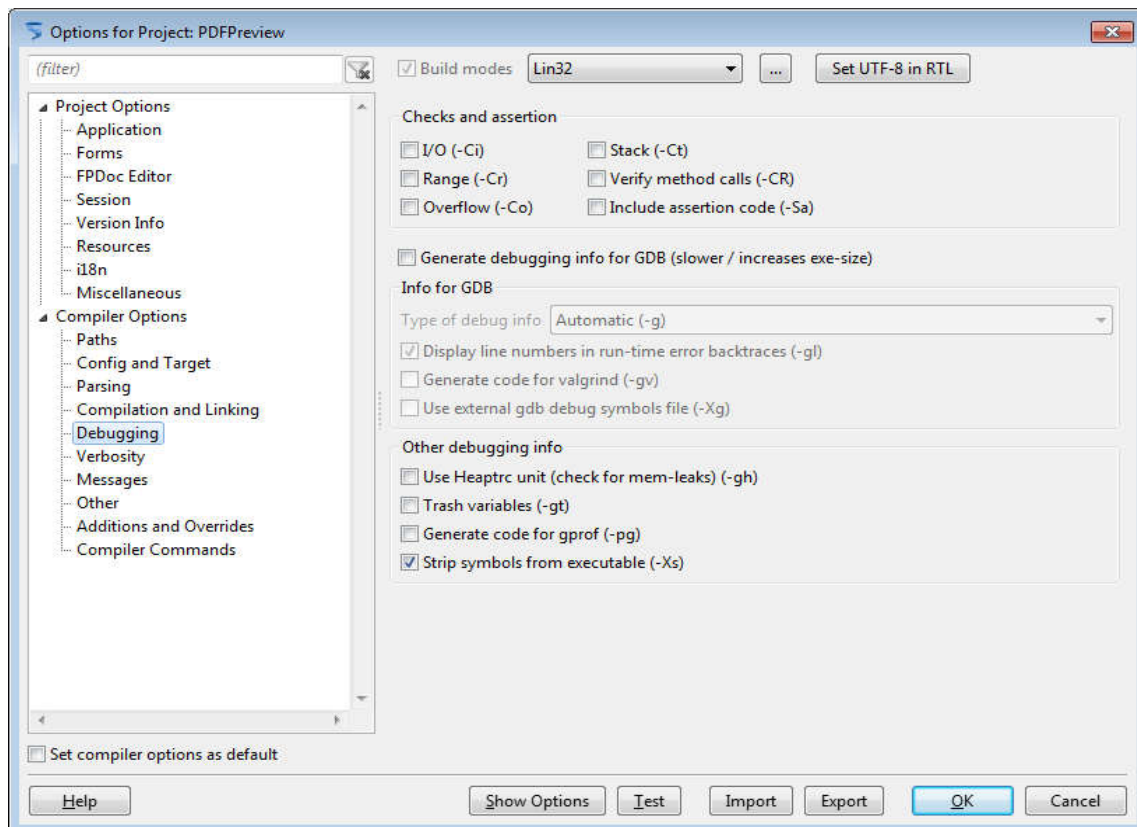
Click to build selected configuration one by one.

That is it. Now you can cross compile project, but for every project you have to create build configuration. Go to Project properties and duplicate default configuration and set operating system,



processor type, put location to libraries, add location to mupdf extra libraries, choose widget (for linux is gtk2 or qt4). Check the pictures for linux 32. Turn off debugging to produce smaller executables.





Note linking order is important. So this library path is important  
 externallibs\$(TargetCPU)\_\$(TargetOS);c:\codetyphon\binLibraries\i386-  
 linux\C:\codetyphon\binLibraries\i386-linux-qt4\

Also compiler comands like

{\$linklib m}, {\$linklib libcurl.a} and

`{ $L g:\dev\mupdf\PDFPreview\externallibs/i386_linux/libgcc_s-5.3.1-20151207.so.1 }`

have different meaning.

Meanings:

`{ $linklib m }` means libm standard linux library

`{ $linklib libcurl.a }` means find library in libcurl.a in current library path (externallibs\\$(TargetCPU)\_\$(TargetOS) defined by library path in Path options or Project Properties

`{ $L g:\dev\mupdf\PDFPreview\externallibs/i386_linux/libgcc_s-5.3.1-20151207.so.1 }` means that compiler must link static library function from exact path location.

Good reference for linking is this url:

<http://www.math.uni-leipzig.de/pool/tuts/FreePascal/prog/node6.html>

## ***About***

Author: Dinko Miljak  
email: [dinmil@gmail.com](mailto:dinmil@gmail.com)  
Verison: 0.0.0.1  
Date: 2016-01-22