

# ConText

Vediamo di mettere un *Readme* usando questa volta, invece del file .md il formato .tex per avere un pdf. ConT<sub>E</sub>Xt è un altro engine per creare Pdf partendo da un sorgente in Tex. È uno degli ultimi sviluppi per quanto riguarda il mondo Tex e per me vale la pena di provare.

## Installazione

L'installazione non l'ho trovata immediata. C'è un Wiki che si trova su <https://wiki.contextgarden.net/Installation> dove viene indicato un link per win64. Questo link, però, non riesco a scaricarlo in quanto lo zip sembra bloccato da Brave. Ho allora dovuto usare in un WLS:

```
wget http://lmtx.pragma-ade.com/install-lmtx/context-win64.zip
```

Ho anche scaricato predentemente uno zip che, da quanto ho capito, mi sembra essere il sorgente di tutto il progetto, quindi niente di eseguibile. Questo zip l'ho scaricato da <http://www.pragma-ade.com/download-1.htm> e l'ho scompattato in

```
D:\scratch\latex\ConTex\source-distribution
```

Lo zip context-win64.zip per windows l'ho scompattato in:

D:\scratch\latex\ConTex. In context-win64.zip si trova un installer che non fa altro che scaricare tutti gli eseguibili mancanti per win64, assieme a tutti i vari font, e li mette nella sottodirectory tex. L'eseguibile mtrun.exe che crea il Pdf è ora in

```
D:\scratch\latex\ConTex\tex\texmf-win64\bin
```

Ho creato la directory **\_my-project** dove ho messo i miei primi esperimenti con ConTex come questo file e altri esempi introduttivi presi dal wiki o dal file install.pdf, così come la patch di SciTe. La prima impressione non è affatto male e credo che continuerò a provarlo. XeLatex (TinyTex) e soprattutto Latex (MikTex) non mi entusiasmano più di tanto.

## Uso

Per generare il file Readme-Context.pdf dal file Readme-Context.tex uso:

```
context Readme-Context.tex
```

Nota bene che setpath.bat funziona solo quando uso cmd.exe. In powershell, invece, il path l'ho settato con il seguente comando:

```
$env:path = "D:\scratch\latex\ConTex\tex\texmf-win64\bin;" + $env:path
```

## Scite

Per vedere come alla fine ho usato SciTe guarda la pagina 3.

Il programma SciTe è molto semplice da installare (non ha l'installer) e si trova su: <https://www.scintilla.org/SciTEDownload.html>. Ho installato la versione a 64 bit. Per lanciarlo in combinazione con CON<sub>T</sub>E<sub>X</sub>T uso la command line con questa sequenza:

```
cd D:\scratch\latex\ConTex
setpath.bat
cd .\scite\bin
SciTe-WithLpeg.exe
```

Per l'installazione del lexer di CON<sub>T</sub>E<sub>X</sub>T in SciTe vado a copiare la directory context in D:\scratch\latex\ConTex\tex\texmf-context\context\data\scite nella directory dove risiede SciTe. Poi bisogna editare il file SciTEGlobal.properties aggiungendo la linea alla fine del file:

```
import context/scite-context-user
```

Per quanto riguarda il Lexer di CON<sub>T</sub>E<sub>X</sub>T, usando SciTe scaricato nel formato binario dal sito ufficiale non riesco a farlo andare in quanto il modulo di Lua lpeg non viene riconosciuto. Questo è dovuto al fatto che il binario di SciTe distribuito dal sito Scintilla non contiene Lpeg e penso che difficilmente lo conterrà nel futuro.

SciTe con CON<sub>T</sub>E<sub>X</sub>T risulta impostato per usare il pdf viewer <https://www.sumatrapdf-reader.org/downloadafter> che è un exe standalone. L'ho messo nella stessa directory dove è CON<sub>T</sub>E<sub>X</sub>T. Così com'è, però, SciTe della distribuzione ufficiale non è utilizzabile.

## Compilare Scite

Per avere un SciTe che funzioni con CON<sub>T</sub>E<sub>X</sub>T ho ricompilato, sotto windows, SciTe dai sorgenti. Ho usato Msys2 (<https://www.msys2.org/>) e per i pacchetti vedi:

[https://packages.msys2.org/package/mingw-w64-ucrt-x86\\_64-gtk2?repo=ucrt64](https://packages.msys2.org/package/mingw-w64-ucrt-x86_64-gtk2?repo=ucrt64)

Nel terminale C:\msys64\msys2.exe ho installato tutti i pacchetti per sviluppare in C/C++. Ho usato la categoria mingw-w64-ucrt che supporta Window10/11 (vedi [https://github.com/libjxl/libjxl/blob/main/doc/developing\\_in\\_windows\\_msys.md](https://github.com/libjxl/libjxl/blob/main/doc/developing_in_windows_msys.md) per una spiegazione delle varie opzioni). Questi sono i comandi che ho usato per installare il compilatore:

```
pacman -S mingw-w64-ucrt-x86_64-gcc
pacman -S mingw-w64-ucrt-x86_64-gtk2
pacman -S mingw-w64-ucrt-x86_64-make
pacman -S --needed base-devel mingw-w64-ucrt-x86_64-toolchain
pacman -S git mingw-w64-ucrt-x86_64-cmake mingw-w64-ucrt-x86_64-ninja
pacman -S mingw-w64-ucrt-x86_64-gtest mingw-w64-ucrt-x86_64-giflib
pacman -S mingw-w64-ucrt-x86_64-libpng mingw-w64-ucrt-x86_64-libjpeg-turbo
cd /d/scratch/latex/ConTex/editor-scite/sourcecode-scite531/scintilla/win32
mingw32-make
```

SciTe si può compilare, sotto windows, solo nella versione win32, quindi i pacchetti Gtk si possono trascurare.

Siccome in Msys2 ho diversi ambienti di compilazione installati è risultato che nella bash non avevo più nessun gcc e mingw32-make. Per questo ho settato nello script iniziale della bash .bash\_profile questo path:

```
export PATH="$PATH:/c/msys64/ucrt64/bin"
```

Per la compilazione basta seguire il file Readme incluso nei sorgenti di SciTe, che poi è lanciare il programma mingw32-make.

## Lpeg in SciTe

Da quanto ho capito, Lpeg è un modulo di Lua che non è integrato nei sorgenti nativi di SciTe. La spiegazione si trova in un commento nel file `scite-context-lexer.lua`. L'autore ha mantenuto una dll sulla falsa riga di ScintillaLua per avere un Lexer di `CONTEXT`. Ad un certo punto si è stufato di mantenere la dll con tutti i suoi cambiamenti di interfaccia ed ha pensato bene di integrare Lpeg in Scite direttamente. A me interessa il codice sorgente di questo nuovo SciTe, ma non sono riuscito a trovarlo. Ho fatto dei tentativi di integrare Lpeg in Scite usando il compilatore mingw-w64 e scaricando Lpeg da [www.inf.puc-rio.br/~roberto/lpeg](http://www.inf.puc-rio.br/~roberto/lpeg). Inizialmente senza risultato in quanto non conosco bene i meccanismi di integrazione di Lua in SciTe. Il meccanismo, però, è molto semplice. Basta compilare la libreria lpeg in SciTe e mettere la libreria Lpeg a disposizione come, per esempio, viene fatto per la libreria Lua di string. Nella repository dove si trova questo file ho messo la patch usata. Alla fine uso il file SciTe statico da me compilato:

`ConTex\scite\bin\SciTe-WithLpeg.exe`

che ingloba tutte i componenti necessari, vale a dire scintilla, lexilla, la `stdlibc++` di mingw64 e Lpeg. Il programma funziona molto bene con il Lexer di `CONTEXT`, mostra un'ottima formattazione dei file in formato `.tex`, esegue la generazione del file Pdf e mostra subito il file così creato (es. con `CTRL + F7`).

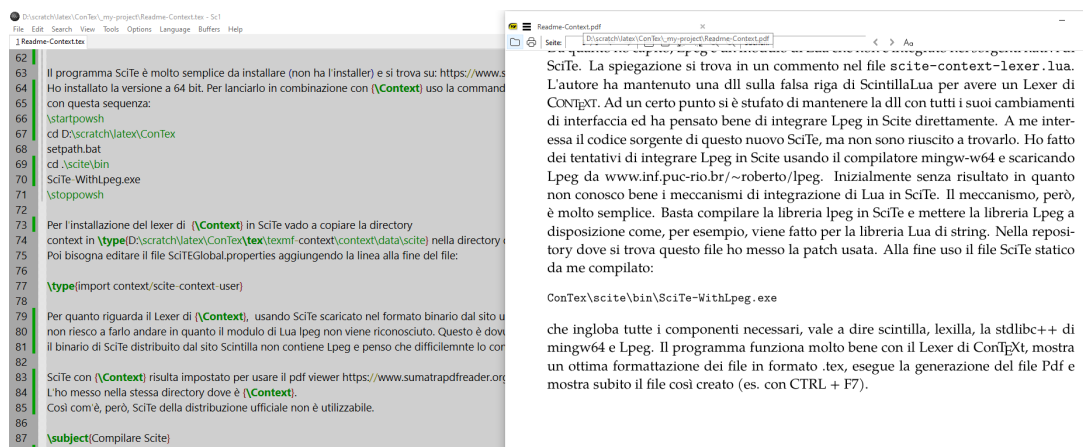


Figure 1 SciTe e Pdf.