

La tipografia nel taschino

Presentazione del sistema $\text{\O}^{\text{P}}\text{s4}\text{G}\text{J}\text{T}$

Salvatore Schirone

Sommario

$\text{\O}^{\text{P}}\text{s4}\text{G}\text{J}\text{T}$ è un sistema \LaTeX portatile per *PenDrive* USB utilizzabile sul sistema operativo Windows (9x, ME, XP) e ottenibile gratuitamente e in formato aperto dalla rete. $\text{\O}^{\text{P}}\text{s4}\text{G}\text{J}\text{T}$ permette di avere sempre a portata di mano un sistema \LaTeX completo e funzionante, e quindi di ricompilare i nostri sorgenti su qualsiasi macchina, anche se sprovvista di un'installazione \LaTeX . In questo articolo presento per la prima volta $\text{\O}^{\text{P}}\text{s4}\text{G}\text{J}\text{T}$, descrivendone la struttura, l'installazione e la personalizzazione.

Abstract

$\text{\O}^{\text{P}}\text{s4}\text{G}\text{J}\text{T}$ is an *opensource* portable USB \LaTeX system for Windows (9x, ME, XP), freely available from internet. $\text{\O}^{\text{P}}\text{s4}\text{G}\text{J}\text{T}$ provides a fully working \LaTeX system always at hand, so that one can compile its own source `.tex` files on every computer. In the present article I will introduce $\text{\O}^{\text{P}}\text{s4}\text{G}\text{J}\text{T}$ for the first time, and I will describe its structure and how to install and customize it.

1 Presentazione di $\text{\O}^{\text{P}}\text{s4}\text{G}\text{J}\text{T}$

Chi usa il sistema \LaTeX per scrivere i propri documenti conosce bene alcuni problemi che si presentano nell'uso intensivo di questo programma:

- l'installazione di un sistema \LaTeX completo è relativamente complessa e occupa qualche centinaio di megabyte sul disco del calcolatore;
- anche per questo motivo, macchine diverse hanno spesso installazioni diverse, o ne sono completamente sprovviste, di conseguenza, risulta a volte impossibile ricompilare e gestire i propri documenti su architetture diverse dalla propria;
- il sistema \TeX Live, un sistema \LaTeX completo multiplatforma utilizzabile da cdrom è scomodo perché il supporto non riscrivibile ci costringe all'uso di un altro supporto (floppy, *PenDrive* USB ecc.) per i sorgenti e gli eventuali pacchetti aggiuntivi utilizzati, inoltre tale sistema non è aggiornabile se non a condizione di rimasterizzare una nuova versione (su un nuovo CD vergine).

Manca quindi, allo stato attuale, una versione di \LaTeX realmente e comodamente portatile, che permetta di avere a portata di mano sia i propri

documenti sia un sistema \LaTeX funzionante anche su macchine differenti. $\text{\O}^{\text{P}}\text{s4}\text{G}\text{J}\text{T}$ nasce esattamente per rispondere a questa esigenza.

La soluzione più comoda e semplice è quella di poter avere tutto ciò che ci occorre racchiuso all'interno di una pen-drive USB: il sistema \LaTeX già configurato, diversi suoi strumenti (tra cui visualizzatori, editor, ecc. . .) e i nostri documenti da compilare, con la possibilità di aggiornare la distribuzione, proprio grazie al fatto che il supporto USB è scrivibile.¹

L'installazione di un sistema \LaTeX su *PenDrive* USB potrebbe sembrare semplice e scontata da realizzare, tuttavia richiede un'analisi molto attenta in merito al software da installare sul supporto USB, così da confezionare una versione completamente portatile dell'intero sistema \LaTeX . Il suo sviluppo è stato difficile, soprattutto per via della struttura tanto complessa del software in questione, poiché risulta strettamente legato alla macchina su cui deve funzionare.

Il caso che, a mio avviso, si verifica più di frequente è quello di trovarsi occasionalmente a lavorare su di una macchina con sistema operativo Windows, dunque naturalmente priva di una qualunque distribuzione \LaTeX . A questo proposito mi sono dedicato alla realizzazione di una distribuzione portatile funzionante (per ora) solo su tale sistema operativo.

L'esperienza che ho maturato nell'ambito della preparazione di sistemi per programmi portatili in questi ultimi due anni, mi ha portato alla realizzazione di *oPenSuite*, ovvero di una versione specifica della mia *suite* per \LaTeX .

L'occasione mi si è presentata dopo aver scoperto l'esistenza di un *how-to* scritto da Alexander Grahn (GRAHN, 2006), poi tradotto per GJT da Stefano Caglio. Grahn spiegava come “portabilizzare” \LaTeX , rendendo disponibile un set di *script batch* per la configurazione della distribuzione $\text{MiK}\text{\TeX}$. Mi sono subito messo all'opera e il risultato è $\text{\O}^{\text{P}}\text{s4}\text{G}\text{J}\text{T}$ ²:

1. Abbiamo bisogno di una *PenDrive* di almeno 512 MB, cosa che oggi non è certo un problema, vista l'ampia diffusione e il costo contenuto di *PenDrive* con capacità pari o superiori ad 1GB.

2. La *suite* è prelevabile dal mio sito personale: <http://www.schirone.it> e il logo della *suite* è stato realizzato da Massimo Caschili.



2 La storia di oPs4GJr

È opportuno fare un breve *excursus* che inquadri i termini della questione, chiarendo innanzitutto cosa si intende per “programma portatile”, e cosa sono le *PenSuite* e nella fattispecie, *oPenSuite* (chi è interessato solamente all’utilizzo di oPs4GJr, può passare direttamente alla sezione successiva).

2.1 Come creare programmi portatili

I programmi per Windows sono quasi sempre distribuiti in forma di pacchetti *autoinstallanti* (ovvero distribuiti sotto forma di un unico file, spesso chiamato *Setup.exe*). Questo significa che durante la sua installazione, e spesso in modo poco trasparente per l’utente, il pacchetto provvede a copiare una serie di file in cartelle diverse del nostro *filesystem* e a scrivere o modificare chiavi nel file registro di configurazione di Windows. Un programma portatile, invece, di solito viene distribuito o in forma d’archivio tipo *zip*, oppure in forma d’archivio *autoestraente* (con estensione *.exe*). L’estrazione dei files dall’archivio fa sì che tutti i files necessari al programma (eseguibili, librerie *dll*, file di configurazione, risorse varie, ecc...) vengano salvati all’interno di una cartella. L’esecuzione del programma avviene lanciando l’eseguibile principale, che di solito porta il nome del programma, *<programma>.exe*. Un programma con queste caratteristiche è portatile e può essere avviato direttamente da una *PenDrive* USB.

I programmi *opensource*, di solito, vengono rilasciati contemporaneamente sia sotto forma d’archivio, che sotto forma di pacchetto autoinstallante. Tuttavia, quando disponiamo solo del pacchetto d’installazione di un programma, renderlo portatile non è un’operazione semplice; poi, se non si dispone dei *file sorgenti*,³ tale operazione si può rivelare addirittura impossibile. Utilizzando particolari software, in grado di estrarre i file del programma dal file *.exe* del pacchetto d’installazione, si possono fare dei tentativi. Un software di questo genere, attualmente tra i più efficaci, è *Universal Extractor*. Il programma permette di saltare l’operazione d’installazione sul PC del pacchetto software che vogliamo rendere portatile, e ne estrae tutti i file come se l’autoinstallante fosse un normale archivio compresso.

Il procedimento illustrato finora non basta a rendere portatile un pacchetto software, poiché bisogna capire se il programma, in fase d’installazione, scrive o modifica una qualche chiave nel registro di configurazione di Windows. Per scoprirlo si possono usare alcuni programmi di monitoraggio, tra

3. Sono i file racchiusi all’interno del pacchetto autoinstallante.

i quali spicca *RegMon*.⁴ Contemporaneamente, ovvero sempre durante la sua fase d’installazione, è necessario scoprire se il programma salva dei dati in cartelle particolari del *filesystem*, come ad esempio in:

```
C:\Documents and Settings\  
[user]\Dati applicazioni
```

e a tal fine ci vengono in aiuto altri programmi di monitoraggio, come ad esempio *FileMon*⁵.

Esistono alcuni piccoli eseguibili detti *launcher* che sono in grado di salvare nella stessa cartella del programma le rispettive chiavi di registro e i dati dell’utente. Uno tra questi è *Reg Rapper*.⁶

Attualmente, il progetto più completo e avanzato di *launcher* è in grado di far funzionare un programma e di salvare nella relativa cartella i dati e le sue chiavi di registro, così da rendere il programma stesso portatile al 100%. Si tratta di *X-Launcher*, presente su *WinPenPack.com*. Questo *launcher* è ormai in grado di “portabilizzare” una gran quantità di programmi; il suo utilizzo è reso difficile dalla sintassi particolare che occorre conoscere per impostarlo adeguatamente, dunque, prima di ogni altra cosa, bisogna analizzare bene il programma che si desidera rendere portatile.⁷

2.2 Nasce oPenSuite

Il fatto di poter disporre di svariati programmi portatili non è sufficiente da sé a realizzare una buona *distribuzione* portatile, poiché occorre anche un qualche *sistema di supporto* che consenta di gestire in modo pratico e comodo tali programmi: lanciarli direttamente dalla *PenDrive*, configurarli in base alle nostre esigenze, aggiornarli all’ultima versione disponibile, ecc... Queste operazioni vengono gestite da un cosiddetto *software launcher*. La gestione di tale programma e la composizione di una collezione di programmi portatili, che può essere ad esempio *base* o *plus*, è quello che noi chiamiamo *PenSuite*. Attualmente esistono diversi progetti attivi (italiani e non) che offrono gratuitamente una *suite* di programmi portatili per *PenDrive*⁸. A mio avviso, il miglior progetto italiano e supportato da un’ampia comunità di utenti e sviluppatori, è il già citato *WinPenPack.com*.⁹

4. <http://www.microsoft.com/technet/sysinternals/utilities/regmon.mspx>

5. <http://www.microsoft.com/technet/sysinternals/FileAndDisk/Filemon.mspx>

6. <http://www.portasoft.org/e107/page.php?4>.

7. Per questo paragrafo mi sono servito ampiamente delle ricerche fatte da Andrea Luparia: <http://www.lupo73.altervista.org>.

8. Tra i più rilevanti segnaliamo: *WinInizio PenSuite* (<http://pensuite.wininizio.it/ita/index.php>), *MozUp* (<http://www.mozillaitalia.it/mozup/>), *PortableApps.com* (<http://portableapps.com/>); per una recensione cfr. (<http://www.lupo73.altervista.org/pro.htm>).

9. <http://www.winpenpack.com>.

La quasi totalità delle *suite* esistenti si basano su *launcher* proprietari, gratuiti, ma non *opensource*, quindi difficilmente personalizzabili e ricompilabili dagli utenti. Solo recentemente l'autore del *launcher* di *WinPenPack* ne ha reso disponibile il sorgente, ma la sua ricompilazione è risultata alquanto problematica e la documentazione molto scarsa.¹⁰ Nella ricerca di un progetto totalmente *opensource* mi sono imbattuto in *FramaKey*,¹¹ un progetto francese che sta portando avanti l'idea di una *suite* composta rigorosamente da programmi *opensource* ed è proprio a questo eccellente progetto che mi sono ispirato nella realizzazione della mia *suite*. In un primo momento ho contribuito direttamente alla sua localizzazione italiana, poi mi sono distaccato con un mio *fork*, cioè una versione modificata, perché, malgrado girasse intorno una attiva comunità, lo sviluppo dei programmi era ed è molto indietro rispetto ad altri progetti anche italiani; inoltre la comunità è risultata prevalentemente francese e non ho trovato il conforto di altri collaboratori italiani. In questo ultimo anno, *oPenSuite* si è evoluto modificando molti aspetti della *FramaKey*, divenendo di fatto un progetto completamente indipendente. La versione attuale, ancora in fase beta, è la 0.6.

Frequentando diversi forum sui siti di supporto alle varie *suite*, ho conosciuto Andrea Luparia, che ha sviluppato una sua *suite* in concomitanza con il mio progetto. Attualmente mi avvalgo della sua collaborazione nella *portabilizzazione* dei vari programmi, e si sta valutando insieme l'ipotesi di giungere ad una fusione dei nostri due progetti.¹²

Il nome "*oPenSuite*" contiene in se stesso la definizione e la finalità del progetto: una *suite* di programmi portatili per supporti rimovibili, *PenDrive*, completamente *opensource*¹³ e particolarmente personalizzabile ed estendibile dall'utente. L'obiettivo di *oPenSuite* è di dare agli utenti la massima libertà di personalizzazione e adattamento. Per questo la *suite* viene rilasciata solo nella versione base con tre programmi essenziali: il *browser* Firefox, l'*editor* Notepad++ e l'utilità di archiviazione 7zip. Gli eventuali programmi aggiuntivi possono essere prelevati da altre fonti, dal momento che il mio compito principale è quello di aggiornare la *suite*

10. Si tratta di un ottimo *launcher*, *ASuite* (<http://www.salvadorsoftware.com/>)

11. <http://www.framakey.org>.

12. L'accordo è ancora in fase di discussione poiché vi sono alcune diversità d'opinione in merito ai software da distribuire: per me la *suite* dovrebbe essere completamente *opensource*, mentre Andrea Luparia, pur adottando un *launcher* *opensource*, preferisce inserire anche programmi con licenze proprietarie *freeware*.

13. I programmi *OpenSource*, comunemente chiamati anche *FreeSoftware*, con una accentuazione più filosofica che tecnica, non sono da confondersi con i programmi *freeware*. Questi ultimi, benché siano rilasciati gratuitamente per l'utilizzo privato, non possono essere modificati (non si dispone del codice sorgente), e possono essere ridistribuiti solo a particolari e restrittive condizioni poste dall'autore.

e di sviluppare/localizzare particolari programmi tra cui, appunto, MiKTEX.

oPenSuite, nei suoi contenuti testuali e multimediali, viene rilasciata sotto licenza *Creative Commons*, "*Attribuzione – Condividi allo stesso modo 2.5 Italia*", mentre i programmi da cui dipende mantengono le loro licenze originali e sono elencati qui di seguito:

AeTrayMenu¹⁴ Una piccola ma potente applicazione che permette di inserire nella *traybar* di Windows un menù flessibile e configurabile, dal quale è possibile lanciare le applicazioni portatili. Viene rilasciata sotto licenza *Creative Commons Public Domain*;

SplashStart¹⁵ Programma di avvio della *suite* con un semplice *splash screen*. Utilizza un file `.ini` per la configurazione ed è distribuito sotto licenza *GPL*;

TiddlyWiki¹⁶ Un *Wiki* contenuto in una unica pagina. Distribuito sotto licenza *BSD*;

NSIS¹⁷ Programma utilizzato per la compilazione dell'installer della *suite* e distribuito sotto la *Common Public License*.

È proprio grazie alla disponibilità dei sorgenti e all'ampia libertà garantita da queste licenze che è stato possibile realizzare `oPs4GJT`.

2.3 Il MiKTEX portatile

La versione portatile di MiKTEX, `ops-MiKTEX` è basata sugli *script batch*¹⁸ messi a disposizione da Grahan. Tali *script*¹⁹ permettono sostanzialmente di:

- inizializzare MiKTeX, Ghostscript e GSview sul dispositivo USB (`init.bat`);
- configurare ed aggiornare *MikTEX* (`mo.bat` e `mpm.bat`);
- ripristinare tutte le configurazioni del PC ospitante (`reset.bat`).

Per rendere portatile MiKTEX è stato necessario installare su di un PC la sua distribuzione più aggiornata (prelevabile dal sito ufficiale del progetto²⁰) in versione "leggera", accompagnato dai programmi accessori Ghostscript e GSview. Il procedimento completo è illustrato qui di seguito.

Una volta creata la cartella `ops-miktex`, si creano al suo interno due nuove sotto-cartelle: `texmf`

14. <http://www.xs4all.nl/~broekroo/aetraymenu>.

15. <http://cdbrowserkit.sourceforge.net/SplashStart/SplashStart.html>

16. <http://www.tiddlywiki.com>, nella traduzione di Maurizio Pollio, <http://pollio.maurizio.googlepages.com/MiniGuidaTiddlyWiki.html>.

17. <http://nsis.sourceforge.net>.

18. Uno *script batch* è un vero e proprio programma costituito da linee di comando passate all'interprete shell di DOS (= *script*), in una determinata sequenza (= *batch*).

19. Per approfondimenti visitare la pagina: <http://www.guit.sssup.it/installazione/drive.php>.

20. <http://www.miktex.org>.

e `localtexmf.texmf` è la cartella dove andremo a copiare tutto ciò che è contenuto nella cartella principale della nostra distribuzione MiKTeX, che potrebbe essere, ad esempio

`C:\Program Files\MiKTeX 2.5`

La cartella `localtexmf` servirà a contenere ulteriori pacchetti aggiuntivi, creati da terze parti, come ad esempio `arstexnica` di `qJr`, che ho utilizzato per comporre questo articolo. Tutti i file batch devono essere nuovamente copiati nella `root` della cartella `ops-miktex`, assieme all'utility propria di Windows `reg.exe` (versione 3.0), reperibile nella cartella di sistema `c:\WINDOWS\system32`. A ciò occorre aggiungere le cartelle relative alle installazioni di Ghostscript e GSview: `gs` e `Ghostgum`. Una impostazione fondamentale riguarda lo script `versions.bat`, che deve contenere le esatte versioni dei programmi MiKTeX, Ghostscript e GSview, nelle rispettive variabili: `MiKTeX_VERSION`, `GS_VERSION` e `GSVIEW_VERSION`.

Per agevolare l'esecuzione dei tre script fondamentali, è stata approntata una interfaccia grafica, con `wxbasic`²¹, un compilatore basic che sfrutta le librerie `wx`. L'idea è stata presa da un articolo di Jerónimo Leal (LEAL, 2006). Sono bastate poche linee di codice per ottenere un discreto pannello di controllo (cfr. Figura 2). Il sorgente e il compilatore sono inseriti nella cartella `src` della `suite`, a disposizione di chi lo volesse modificare e ricompilare, così da espanderne le funzioni o semplicemente migliorare l'aspetto grafico.

Infine, per rendere il tutto più funzionale e pratico, è stato inserito nella `suite` l'editor `Texmaker`, anch'esso avviabile dal **Pannello di Controllo**. Grazie alla sua semplicità e leggerezza non ha richiesto nessun intervento di *portabilizzazione*. L'ultima fase del lavoro ha riguardato la compressione e l'archiviazione della cartella del progetto, `ops-miktex`.

3 Usare `oP4qJr`

3.1 Installazione e uso

È molto semplice installare e utilizzare `oP4qJr` e, come prima cosa, occorre prelevare il pacchetto `ops4guitInstaller_base-0.1.0.0-it.exe` dal sito dell'autore. L'operazione potrebbe richiedere alcuni minuti, a seconda del tipo di connessione che si sta utilizzando, poiché si tratta di un archivio di ben 84.69MB. Il "doppio click" sull'eseguibile avvia la procedura d'installazione e si viene subito informati del fatto che il pacchetto è distribuito nei termini della licenza *Creative Commons* e che per poter proseguire, occorre accettare tali termini. Successivamente viene chiesto di indicare il supporto USB sul quale si desidera installare la

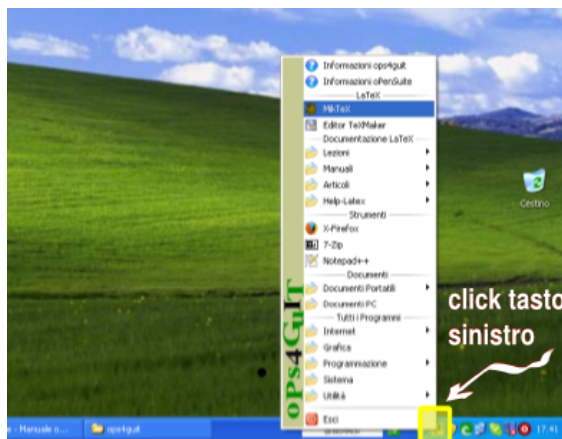


FIGURA 1: Menu tasto sinistro

suite (è possibile installarla anche in una cartella del nostro disco locale). Lo script dell'installer *NSIS* completerà la procedura d'installazione e, giunto al termine, ci chiederà se vorremo avviare l'applicazione.

Successivamente, grazie ad un sistema di *auto-run*, ogni qualvolta inseriremo la pen-drive nella porta usb del computer ci apparirà uno *splash screen* che ci pregherà di attendere qualche secondo. Trascorso questo tempo, nella *traybar* del *Desktop* di Windows troveremo l'icona `oP4qJr`, dalla quale potremo lanciare le varie applicazioni ed applicare alla *suite* le nostre personalizzazioni.

Con il tasto sinistro del mouse, potremo accedere al menu contestuale principale (cfr. Figura 1), che consente di avviare le varie applicazioni, esplorare la cartella dei documenti portatili o accedere anche ai documenti presenti sul computer ospitante.

Le due voci del del menù **Informazioni** consentono di consultare rispettivamente il manuale generale di *oPenSuite* e quello specifico di `oP4qJr`, all'interno del browser predefinito (*X-Firefox*). Entrambi i manuali possono essere contestualmente modificati e aggiornati grazie a *Tiddlywiki*. Sele-

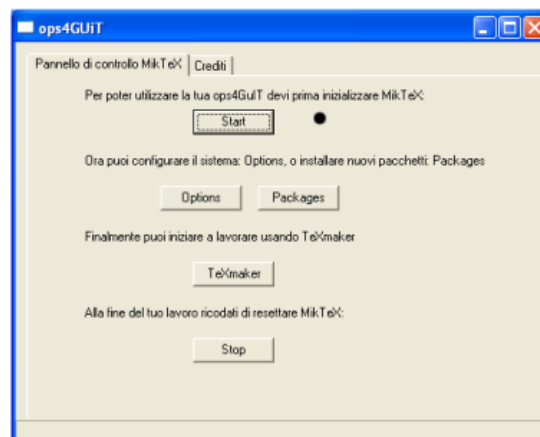


FIGURA 2: Pannello di controllo `oP4qJr`

21. <http://wxbasic.sourceforge.net/>

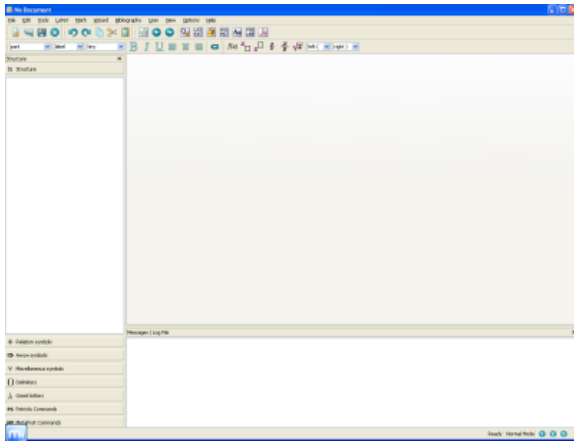


FIGURA 3: Texmaker

zionando **Esci**, termina l'esecuzione di **oPs4GUIT**. In aggiunta, dallo stesso menù potremo disporre di tutti gli strumenti messi a disposizione dalla *suite* e avviare tutti gli eventuali programmi che avremo installato durante la nostra fase di personalizzazione. In particolare, potremo consultare rapidamente alcuni tutorial e manuali distribuiti dal **GUIT**.

La sezione **LaTeX** conduce alla configurazione vera e propria della nostra distribuzione **TEX** portatile e, selezionando la voce **MiKTEX**, ne appare il pannello di configurazione (cfr. Figura n. 2). Il pulsante **Start** avvia, all'interno di una finestra DOS, lo script di configurazione generale di **MiKTEX**. Al termine dell'operazione, se il processo avrà avuto successo, avremo a disposizione il nostro sistema **MiKTEX** pronto per l'uso. Attraverso il pulsante **Opzioni** potremo effettuare diverse operazioni di configurazione, oppure installare altre estensioni per **MiKTEX** (*packages*). Il pulsante **Texmaker** avvierà l'omonimo editor (cfr. Figura n. 3), che potrà essere configurato ulteriormente attraverso le sue impostazioni di configurazione. *Texmaker* può anche essere avviato dal menù principale della *suite*. Quando avremo terminato il lavoro, dovremo ricordarci di arrestare il sistema tipografico premendo il pulsante **Stop**. Ciò avvierà lo script `reset.bat` che riporterà tutte le impostazioni del registro di sistema allo stato iniziale, così da non lasciare alcuna traccia fastidiosa dei programmi appena utilizzati sul computer ospitante e senza compromettere le impostazioni di una eventuale distribuzione **MiKTEX** preesistente nel sistema.

3.2 Personalizzazione

un click Con il tasto destro del mouse (cfr. Figura n. 4) si accede ad un altro menù dal quale possiamo gestire tutti i parametri di configurazione della *suite*, accedere al prompt di DOS e ad altre informazioni supplementari.

Selezionando la voce **Edita menu**, si avvia l'editor *Notepad++* con il file

Framakey/FramaLauncher/FramaLauncher.ini

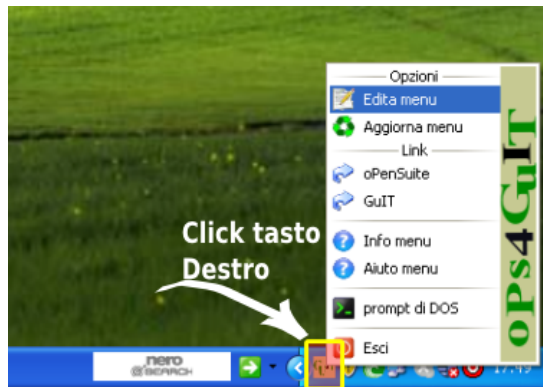


FIGURA 4: Menu tasto destro

precaricato al suo interno. Agendo su questo file si possono aggiungere e/o modificare le *voci* di menù (cfr. Figura n. 5). La sintassi da seguire è abbastanza intuitiva e le varie sezioni sono ampiamente commentate, dunque è sufficiente dare uno sguardo agli esempi già pronti per poi tentare di scrivere le proprie *voci*. Per fare un esempio, se si desiderasse utilizzare anche l'editor *Vim*, per scrivere i propri sorgenti **LATEX**, e lo si volesse far 'convivere' con *Texmaker*, dovremmo, prima di tutto, prelevare l'archivio dell'applicazione *GVim Portable* dall'indirizzo <http://portablegvim.sourceforge.net> e poi estrarne i file, ponendoli nella cartella *Apps* della nostra *suite*. Successivamente ci accerteremo dell'esatto percorso in cui si venisse a trovare l'eseguibile principale dell'applicazione,

`Apps/GVimPortable/GVimPortable.exe`

indi creeremo la relativa voce del menù come spiegato nei passi seguenti.

1. Avvieremo l'editor di configurazione e con il tasto destro del mouse selezioneremo la voce **Edita menu**.
2. Ci sposteremo alla sezione `[Menu.Left]` e, in coda alla parte di codice il cui commento si riferisce alla sottosezione **LaTeX**, aggiungeremo il seguente blocco di codice:

```
Type: item; Caption: "GVimPortable";
```

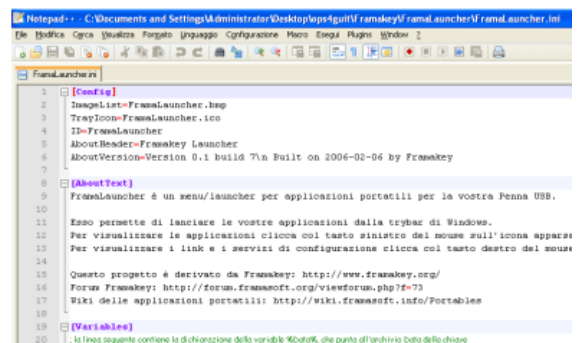


FIGURA 5: Configurazione del menù

```
Action: run;
FileName: "%Apps%\GVimPortable\
GVimPortable.exe";
Glyph: 26
```

La sintassi è particolarmente intuitiva, infatti si può notare esser composta da una serie di coppie del tipo `<elemento>: <valore>`;²²:

Type:item annuncia che ci si appresta ad aggiungere o modificare una voce di menù;

Caption:<nome> specifica il nome della voce che apparirà scritto nel menù;

Action:run si riferisce all'azione da eseguire, in questo caso si tratta di avviare (*run*) una applicazione;

FileName:<eseguibile> specifica il percorso e il nome dell'eseguibile associato all'applicazione che si desidera avviare;

Glyph:<n> è il numero identificativo dell'icona associata al programma.

Per inserire l'icona di *Vim* nel suddetto menù, occorre agire sul file

```
Framakey\Fra...er\FramaLauncher.bmp
```

Bisogna aprirlo con un editor grafico e copiarvi al suo interno l'icona prescelta in formato **bmp** (e su sfondo rosa) entro uno dei riquadri liberi di coda. La posizione della nuova icona sarà associata ad un numero che identificherà proprio il valore di **Glyph**.

3. Salveremo il file e chiuderemo l'editor.
4. Per rendere efficaci le modifiche appena apportate, aggiorneremo il sistema con un semplice click destro del mouse su **Aggiorna menu**.
5. Infine, nel caso in cui avessimo commesso un errore di sintassi, riceveremmo un avviso immediato e dovremmo riaprire l'editor per correggerlo.

A questo punto, nel menù della *suite*, dovremmo trovare la nuova voce per poter avviare *Vim*. Se volessimo personalizzare questo editor, così da poterlo utilizzare al meglio con L^AT_EX, potremmo prelevare il *plugin* adatto da <http://vim-latex.sourceforge.net/index.php> e seguire le istruzioni per la sua installazione. In particolare, durante la procedura d'installazione del *plugin*, ogni qualvolta venga fatto riferimento alla cartella `~/vimfiles`, per la nostra *suite* dovremo riferirci alla cartella

```
Apps/GVimPortable/App/vim/vim70
```

Il menù a scomparsa, visualizzabile col tasto destro del mouse, contiene un elenco di collegamenti utili a siti internet, che possono essere ampliati o modificati a piacimento dall'utente. La voce **Info**

²². Il punto e virgola, che segue ciascuna voce, è necessario.

menu mostra una finestra con informazioni relative ad *AeTrayMenu*, ovvero il *launcher* della *tray-bar*, mentre **Aiuto menu** ne fornisce l'help in linea. **Prompt di Dos** esegue il comando `cmd.exe`, che mette a disposizione dell'utente la console di MS-DOS. **Esci** ha la stessa funzione del suo omonimo comando, attivabile col tasto sinistro del mouse.

4 Sviluppo futuro e ringraziamenti

Lo sviluppo del *core* del sistema dipenderà essenzialmente dall'evoluzione che avrà – e se l'avrà, *oPenSuite*. La *todo*list,²³ comunque, è la seguente:

- implementare altri strumenti complementari a L^AT_EX;
- inserire anche *Emacs* come *editor* alternativo a *Texmaker*;
- aggiungere altri manuali del G_UI_T;
- sviluppare il manuale wiki della *suite*;
- migliorare la grafica e il pannello di controllo.

Tutti gli utenti sono invitati a partecipare al progetto così da sostenerne lo sviluppo futuro. Ciò può essere fatto in vari modi, ad esempio scrivendo all'autore per segnalare eventuali errori, chiedere l'aggiunta di nuove funzioni, oppure aggiornando il manuale L^AT_EX attraverso *TiddlyWiki*. Più direttamente, è possibile partecipare al progetto anche tentando di personalizzare e sviluppare autonomamente la *suite* e mettendo a disposizione di tutti gli utenti il proprio lavoro, tramite il forum di discussione raggiungibile dal sito dell'autore.

Si ringraziano in particolare:

- Massimo Caschili per il logo $\text{P}\text{S}\text{4G}\text{I}\text{T}$;
- Andrea Luparia per la fattiva collaborazione a *oPenSuite*;
- Stefano Caglio per la traduzione di Grahn;
- gli amici del G_UI_T per tutto il resto.

Riferimenti bibliografici

GRAHN, A. (2006). «Miktex+ghostscript+gsview+usb-drive-howto». <http://www.ctan.org/tex-archive/documentation/MiKTeX+Ghostscript+GSview+USB-drive-HOWTO.txt>.

LEAL, J. (2006). «Esperienze didattiche con L^AT_EX». *ArsTeXnica*, (2), pp. 71–74. URL <http://www.guit.sssup.it/arstexnica.php>.

▷ Salvatore Schirone
<http://www.schirone.it>
info@schirone.it

²³. In gergo informatico, dall'inglese, è l'elenco dei compiti di cui ci si deve ancora occupare.