

Relazione di progetto

Lora

Laurent Claessens

15 gennaio 2016

Indice

| | | |
|----------|---|----------|
| 1 | Scopo et descrizione generale del progetto | 1 |
| 1.1 | Cosa fa ? | 1 |
| 1.2 | L'algoritmo di backup | 1 |
| 1.3 | Purge | 2 |
| 1.4 | Multi-thread | 3 |
| 1.5 | Perché è così grosso ? | 3 |
| 2 | Gerarchia polimorfa : <code>GenericTask</code> | 3 |
| 3 | Gerarchia polimorfa : <code>MainLoop</code> | 3 |

1 Scopo et descrizione generale del progetto

1.1 Cosa fa ?

Lora è un programma che fa due cose.

- Fa un backup della cartella `$HOME`
- Cerca le cartelle che sono *git repository*, verifica se hanno bisogno di un `git add`, `git commit` oppure di aggiungere dei file dentro `.gitignore`. Un interfaccia grafica aiuta a questo tipo di manutenzione.

1.2 L'algoritmo di backup

Il programma si invoca con¹

¹Vedere il manuale dell'utente per maggiori informazioni.

./lora

In questo esempio, supponiamo fare il backup di `<home>` (che sarà usualmente la cartella `$HOME`) dentro la cartella `<backup>` (che carà su un disco esterno²). Di più abbiamo bisogno di una cartelle `<purge>` accanto a `<backup>`.

Lora fa un *loop* su tutti i file di `$HOME`.

1. Se `<home>/foo/bar.txt` è diverso³ da `<backup>/foo/bar.txt`.

- Sposta `<backup>/foo/bar.txt` verso `<purge>/foo/bar.txt` Quindi non ci sarà mai una perdita di dati : Lora è più di un programma di sincronizzazione. Lora fa la sincronizzazione di `<home>` con `<backup>`, ma guarda i vecchi files in `<purge>`.
- Copia `<home>/foo/bar.txt` su `<backup>/foo/bar.txt`

Se `<home>/foo/bar.txt` ma `<backup>/foo/bar.txt` non esista.

- Copia.

Vantaggio di Lora su altri programmi di backup⁴ : i dati si ritrovano con una semplice copia di `<backup>` si `<home>`. L'utente non ha bisogno si Lora per ritrovare i dati.

In particolare, l'utente può ritrovare i sui dati anche da una chiave USB minimale senza interfaccia grafica.

1.3 L'algoritmo di purge

Quando il backup è finito, Lora fa un *loop* su la cartella `<bakcup>`.

Se `<backup>/foo/bar.txt` esista ma `<home>/foo/bar.txt` non esista, Lora pensa che l'utente abbia eliminato questo file.

- Sposta `<backup>/foo/bar.txt` verso `<purge>/foo/bar.txt`.

Nota bene. The purge directory here is not exactly the same as the previous one. In fact each time you launch Lora, a new directory is created :

`<purge>/<date>/<hour-minutes>`

²Che dovrebbe essere cifrato, però è un'altra storia.

³L'attributo *size* o *last write time* è diverso.

⁴Soprattutto quelli che implementano un backup incrementale.

Inside that directory, Lora creates two directories : **modified** and **removed** which contain the files that were respectively seen to be modified and removed.

A general concept of Lora is that your data is more precious than your disk space and than everything⁵. If you understand well the backup/purge concept, you can imagine the extreme disk space waste when you just rename, say your music directory.

1.4 Multi-thread

Il backup stesso occupa 2 threads.

- Il primo è un *loop* sulla cartella `$HOME` che alla ricerca di files o cartelle di cui aggiogare il backup. Questo thread aggiunge delle *tasks* (vedere 2) a una lista.
- Il secondo thread legge la lista e esegue le copie da fare.

Questi threads non hanno interfaccia grafica, e questo è una scelta di design : devo poter fare un backup anche (e soprattutto) quando il computer va male. Per esempio voglio poter fare il backup da una chiave USB minimale⁶.

Nello stesso tempo, un elenco dei *git repository* che hanno bisogno di pulizia si aggiorna in un interfaccia grafica.

1.5 Perché è così grosso ?

Avevo cominciato il lavoro all'inizio di Ottobre 2015 perché avevo bisogno di un programma di backup per uso personale. La parte di backup era già quasi finita quando abbiamo a parlare del progetto in classe.

Quindi per il progetto stesso, ho aggiunto la parte “git” e l'interfaccia grafica.

2 Gerarchia polimorfa : `GenericTask`

3 Gerarchia polimorfa : `MainLoop`

⁵Better to crash than to manage a borderline situation.

⁶Dal punto di vista del codice, il backup è abbastanza ben separato delle dipendenze su Qt. Dovrei scrivere un giorno una versione che si può veramente eseguire fuori dell'interfaccia grafica.