

Chattare con... un editor di testo

Vediamo come chattare su IRC con GNU/Emacs...



Come un buon *sistema operativo* anche GNU/Emacs permette di restare in contatto con il mondo tramite dei client di chat o messaggistica istantanea.

Il nostro amato Emacs viene già fornito, dalla versione 22, con un ottimo client IRC: **erc**. Assumeremo che la versione di Emacs installata sul vostro computer sia la 23, di conseguenza non avremo bisogno di installare **erc**, tuttavia se la versione da voi utilizzata è precedente alla 23 (e in particolare precedente anche alla 22), allora potrete trovare delle precise istruzioni per l'installazione sull'*Emacswiki*.

Utilizzo rapido di erc

Erc può essere già utilizzato senza richiedere alcuna configurazione, è sufficiente richiamare la funzione **erc**:

M-x erc <RET>

Ci verranno chieste alcune informazioni:

- **IRC server**: il server IRC al quale desideriamo connetterci (ad esempio *irc.freenode.net*);
- **IRC port**: la porta del server sulla quale vogliamo avviare la connessione (ad esempio *6667*);
- **Username**: il vostro nickname;
- **Password**: la password che volete usare per essere autenticati sul server (non è sempre necessaria).

Una volta terminato l'inserimento dei dati verrà avviata la connessione, e successivamente l'autenticazione, al server IRC prescelto. La connessione così effettuata (dato che non è ancora presente alcuna configurazione nel nostro *.emacs*) non prevede alcun autojoin (perché nel *.emacs* non abbiamo indicato alcun canale), quindi dovremo provvedere manualmente ad effettuare un join ai canali desiderati o tramite i comandi previsti dal protocollo IRC (ad esempio */join #emacs*) o tramite:

M-x erc-join-channel <RET>

Configurazione basilare di erc

Quest'utilizzo di **erc** è scomodo, soprattutto per chi utilizza spesso **erc**, quindi è arrivato il momento di creare gli automatismi che velocizzeranno le nostre attività. Sebbene questa fase possa essere svolta tramite *Customize*, noi opteremo per la scrittura di poche e semplici righe di Emacs Lisp nel nostro *~/.emacs*. Iniziamo dunque la configurazione di **erc**:

C-x C-f ~/.emacs <RET>

Iniziamo informando Emacs riguardo al nostro server di chat preferito (supporremo come esempio di volerci connettere sempre al server *irc.freenode.net* e in particolare ai canali *#emacs* e *#lisp*):

```
(setq erc-server "irc.freenode.net")
```

```
(setq erc-port "6667")
```

I nostri dati:

```
(setq erc-nick "emacs_user")
```

```
(setq erc-password "my_very_secure_password")
```

A questo punto è necessario specificare in quali canali entrare automaticamente (l'autojoin è attivato di default):

```
(setq erc-autojoin-channels-alist  
'(("freenode.net" "#emacs" "lisp")))
```

La configurazione basilare è terminata; richiamando **erc** ci verrà chiesto solo di confermare i dati già inseriti nel nostro *.emacs*.

Qualora desiderassimo avviare la connessione ad **erc** ad ogni avvio di Emacs sarà sufficiente aggiungere nel *.emacs* la seguente riga:

```
(erc :server "irc.freenode.net" :port  
"6667" :nick "emacs-user" :password  
"my_very_secure_password")
```

Configurazione avanzata di erc

Iniziamo dicendo che non è una buona pratica quella di inserire le nostre password in chiaro nel *.emacs* (anche considerando che è consuetudine fra gli utenti di Emacs lo scambio di questi file sia perché da altri utenti si può sempre imparare qualcosa, sia perché è interessante osservare le abitudini emacsiane di altre persone). Una possibile soluzione è quella di mettere (*setq erc-password "my_very_secure_password"*) in un altro file, ad esempio *~/.emacs.d/ercpasswd* e richiamare questo file con:

```
(load "~/.emacs.d/ercpasswd")
```

È poi molto comune la necessità di specificare più server nel *.emacs* con diversi canali da *joinare* e con differenti dati di accesso... Vediamo un esempio:

```
(setq erc-autojoin-channels-alist '  
("freenode.net" "#emacs" "#salug!")  
("gimp.net" "#gimp")
```

```
)
```

```
(erc :server "irc.freenode.net" :port  
"6667" :nick "emacs-user" :password  
freenode_pwd)
```

```
(erc :server "irc.gimp.net" :port "6667"  
:nick "gimp-user" :password gimp_pwd)
```

In questo esempio abbiamo supposto di volerci connettere automaticamente a *irc.freenode.net* con utente *emacs_user* e password specificata tramite la variabile *freenode_pwd* letta da un file esterno nascosto e di volerci connettere, sempre automaticamente, a *irc.gimp.org* con credenziali differenti. Quando sarà stabilita la connessione con i due server verrà effettuato l'autojoin

Chattare con... un editor di testo

Vediamo come chattare su IRC con GNU/Emacs...



ai canali #emacs e #salug! (su *irc.freenode.net*) e al canale #gimp (su *irc.gimp.net*).

Altra possibile personalizzazione riguarda le notifiche che vogliamo ricevere nella *modeline* o per quali eventi di erc aprire nuove finestre o nuovi frame. La riga seguente mi permette di non ricevere notifiche nella *modeline* per molti eventi ai quali non sono interessato:

```
(setq erc-track-exclude-types
(quote ("KICK" "NICK" "PART"
"QUIT" "MODE" "333" "353")))

```

Con le seguenti due righe chiedo a erc di aprire un nuovo frame (nel linguaggio comune e' cio' che noi chiameremmo finestra... mentre quando si parla di window nella terminologia di Emacs si intende cio' che invece noi chiameremmo frame :-P) per ogni query effettuata o ricevuta:

```
(setq erc-auto-query (quote frame))

```

```
(setq erc-query-display (quote
frame))

```

Erc e i moduli

Erc si compone di molti moduli che possono essere caricati per aggiungere nuove funzioni. Alcuni di questi moduli sono abilitati di default (box0), altri sono presenti in erc, ma non attivati (box1).

A questi si affiancano altri moduli, la cui installazione solitamente non viene effettuata tramite la modifica della variabile *erc-modules*, ma seguendo le istruzioni fornite dal loro ideatore. Esempi di questi moduli sono **ErcChess** (per giocare a scacchi con i nostri interlocutori), **ErcSpeak** (che dona la voce a Emacs consentendogli di pronunciare i nostri messaggi di chat), **ErcSpeedbar** (che fornisce l'integrazione con la *Speedbar* per accedere rapidamente ad alcune informazioni su canali e utenti), **ErcDCC** (implementa il *Direct Client-to-Client* per rendere possibile lo scambio di file tramite IRC) e tanti altri.

Questa e' una sezione della configurazione di erc nel mio *.emacs*, quella che definisce i moduli da caricare:

```
(setq erc-modules (quote (autoaway
autojoin button completion fill
ircontrols list match menu move-to-
prompt netsplit networks noncommands
readonly ring smiley stamp track
xdcc)))

```

- **ErcAutoJoin**: necessario per effettuare l'autojoin ai canali (l'abbiamo inconsapevolmente utilizzato negli esempi precedenti).

- **ErcButton**: rende le parole come i nick o i nomi dei canali cliccabili.

- **ErcChannelTracking**: permette di tenere traccia dell'attivita' nei canali tramite la *modeline*. In questo modo potremo continuare a svolgere il nostro lavoro ed essere informati di eventuali messaggi sul canale IRC al quale siamo interessati.

- **ErcCompletion**: completare i nick con un tab e' vitale!

- **ErcControlCharacters**: per evidenziare o rimuovere i caratteri di controllo.

- **ErcFilling**: tramite questo modulo le righe troppo lunghe vengono spezzate.

- **ErcHistory**: per abilitare una history.

- **ErcList**: per ottenere una lista dei canali ordinata alfabeticamente o per dimensione in un altro buffer apposito.

- **ErcMatch**: se siamo interessati a particolari parole chiave, ad esempio vogliamo essere informati ogni qualvolta qualcuno parla di *vi* sul canale di emacs allora questo modulo fa al caso nostro. Verifica le corrispondenze fra i messaggi inviati e le parole configurate ed eventualmente le evidenzia. E' utile anche nel caso in cui su IRC ci chiamano con un abbreviativo o un altro nome oltre a quello con cui siamo connessi in quanto utilizzando **ErcMatch** possiamo comunque essere notificati della pronuncia del nostro nome.

- **ErcNetsplit**: un netsplit puo' verificarsi nel momento in cui un server viene disconnesso e si spostano tutti gli utenti connessi su un altro server (ad esempio si disconnette il server *brown.freenode.net* e gli utenti vengono spostati su *barjavel.freenode.net*). Quando accade un netsplit sembra solitamente che tutti gli utenti si disconnettono contemporaneamente e le notifiche inviate dal client IRC riempiono facilmente la finestra. **ErcNetsplit** e' in grado di rilevare i netsplit e nascondere la pioggia di messaggi di uscita tenendo quindi pulita la finestra.

- **ErcNonCommands**: i comandi del tipo /COMMAND vengono solitamente inviati al server IRC e indicano delle operazioni da compiere. Quelli pero' che non richiedono di essere inviati al server vengono detti *non-commands* e vengono gestiti da questo modulo.

- **ErcReadOnly**: per rendere in sola lettura (quindi non modificabili) le righe che riceviamo dal server.

box 0

- **ErcAutoAway**: utilizzato per impostare lo stato di away dopo un determinato numero di secondi.

- **ErcBbdb**: fornisce l'integrazione con *BigBrotherDataBase (BBDB)*.

- **ErcLogging**: rende possibile il salvataggio dei log delle chat.

- **ErcMorseCode**: lo si utilizza quando si vuole comunicare tramite codice Morse.

- **ErcNickserv**: fornisce funzioni per l'autenticazione automatica al Nickserv.

- **ErcNotify**: possibilita' di ricevere notifica per determinati eventi.

- **ErcReplace**: filtra i messaggi per effettuare sostituzioni nel testo.

- **ErcScrollToBottom**: il modulo in questione permette di mantenere il prompt nella parte bassa della finestra e mostrare gli aggiornamenti sempre nella parte bassa. Puo' risultare utile quando dopo una pausa si ritorna al computer e si vuole avere subito la situazione sotto controllo.

- **ErcSmileys**: lo si utilizza per visualizzare delle simpatiche faccine al posto del classico ":-)"

- **ErcSound**: le notifiche sonore diventano possibili con questo modulo.

- **ErcSpelling**: si occupa dell'integrazione di *flyspell* in erc che viene usata per eseguire lo spell check al volo.

- **ErcTruncation**: tronca i buffer quando diventano troppo lunghi.

box 1