

Linux e Audio Real time

LINUX E AUDIO REAL TIME

ilcorsaronero@gmail.com

ottobre 2007

Indice del documento

Indice del documento.....	2
Introduzione.....	4
Un documento per capire e provare soluzioni libere.....	4
Il documento si può scaricare liberamente dal sito dell'autore.....	4
Rilasciato sotto GNU General Public License Version 3.....	4
Requisiti teorici.....	5
Quali sono le problematiche della registrazione e/o riproduzione audio?.....	5
La soluzione del buffer.....	5
Il problema del buffer.....	6
Serve avere un buffer piccolo?.....	6
Come ridurre al minimo la lunghezza di questo buffer senza perdere la caratteristica real time delle applicazioni?.....	7
A chi chiedere le risorse necessarie?.....	7
In pratica?.....	7
Compilare o non compilare?.....	8
Requisiti Hardware.....	10
Hardware.....	10
Il dvd di Ubuntu Studio.....	10
Un personal computer compatibile.....	10
Una buona scheda audio compatibile.....	11
Cavo jack e adattatore jack mini jack per gli strumenti amplificati.....	11
Uno o più microfoni di qualità più mixer per tutti gli altri strumenti.....	11
Cuffie chiuse “da dj” per registrare.....	12
Due casse a due vie per riprodurre.....	12
Requisiti Software.....	13
Programmi.....	13
Jack.....	13
Qjackctl (Jack Control).....	13
Jack-Rack.....	13
Caps, Tap-plugins, Swh-plugins, Dssi-example-plugins	13
Jamin.....	14
Timemachine.....	14
Gtick.....	14
Lingot.....	14
Audacity.....	14

Linux e Audio Real time

K3b.....	14
in Pratica.....	15
Configurare Ubuntu Studio.....	15
Lingot.....	15
Configurare la scheda sonora.....	16
Configurare qjackctl e lanciare jack.....	16
Jamin e le prime connessioni con Jack-Control.....	19
Jack-rack e gli effetti.....	22
Gtick.....	23
Timemachine per registrare.....	23
Audacity.....	24
K3b.....	24
Problemi comuni.....	25
Ho installato ubuntu con il kernel realtime ma jack si rifiuta di funzionare in modalità realtime. L'avvio fallisce con questo messaggio di errore.....	25
Il sistema funziona, ma è scomodo dover avviare uno per uno tutti questi programmi ogni volta. Si può semplificare la procedura?.....	25
Commenti e conclusioni.....	27
Vantaggi.....	27
Svantaggi.....	27
Conclusioni: è possibile registrare un demo con linux?.....	27

Introduzione

Un'introduzione pratica al mondo delle applicazioni audio real time nel sistema operativo libero gnu/linux

- **Un documento per capire e provare soluzioni libere**

Il documento è indirizzato a chi, con il suo personal computer, vuole sperimentare e venire a conoscenza delle migliori soluzioni audio real time disponibili oggi per il sistema operativo libero gnu/linux

- **Il documento si può scaricare liberamente dal sito dell'autore**

Una versione del documento aggiornata è sempre disponibile per il download o per la consultazione sul mio sito personale <http://www.ntoni.altervista.org/>

- **Rilasciato sotto GNU General Public License Version 3**

A questo documento si applica la GNU GENERAL PUBLIC LICENSE Version 3. Siete liberi di copiare, di modificare e poi di copiare di nuovo tutti i contenuti che vi interessano

Requisiti teorici

Quali sono le problematiche della registrazione e/o riproduzione audio?

Quando ascoltiamo o suoniamo una canzone, è indispensabile che il flusso audio continui senza interruzioni

- Può capitare, se la nostra applicazione non viene eseguita correttamente, che d'improvviso il computer si fermi a caricare, fermando così la riproduzione o la registrazione audio
- Un'applicazione di videoscrittura può fermarsi a caricare senza farci perdere parole o frasi scritte
- Un'applicazione che sta registrando un flusso audio, invece, non può fermarsi a caricare, perché se si fermasse a caricare, perderemmo parte di quello che stiamo registrando

La soluzione del buffer

Tra i programmi audio e la scheda sonora viene frapposto un buffer o meglio un serbatoio di informazioni audio

- Il buffer permette che questi programmi audio non interrompano o scattino neanche quando l'utilizzo complessivo della cpu è elevato
- Il buffer infatti è un serbatoio di informazioni audio, che può essere svuotato o ricaricato quando più si preferisce, e quindi nei momenti di utilizzo meno intensivo della cpu
- Le applicazioni per ascoltare la musica possono così leggere da un serbatoio, che ha sempre informazioni audio da fornire
- Le applicazioni per registrare la musica hanno invece la possibilità di scrivere informazioni audio nel buffer anche quando la cpu è impegnata

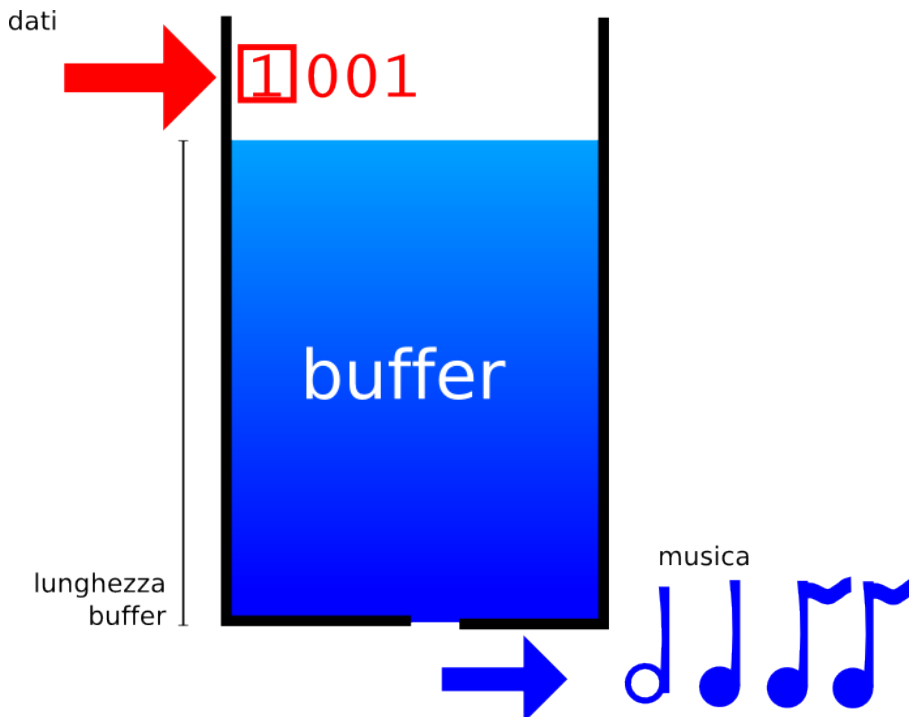


Illustration 1: Schema del buffer sonoro

Il problema del buffer

Il buffer è utile ma toglie reattività al sistema

- Il buffer è una vera e propria coda che rallenta la reattività delle applicazioni audio
- Tanto più quanto più questo buffer è elevato, tanto meno il sistema audio è reattivo
- Infatti, prima che i dati appena inseriti nel serbatoio escano dallo stesso, bisogna aspettare che tutti le informazioni audio precedentemente accumulate vengano suonate

Serve avere un buffer piccolo?

Sì, un buffer piccolo minimizza il tempo di risposta delle applicazioni

- Nei videogiochi, con un buffer piccolo si possono sentire i passi dei nemici alle nostre spalle e voltarci prima che questi facciano frag a noi
- Nei programmi per mixare la musica, un buffer piccolo permette di scratchare ed effettuare mp3 al volo, infatti con una bassa latenza (sinonimo di buffer piccolo) il tempo che intercorre tra la pressione del tasto play e l'ascolto della nuova canzone si riduce al minimo
- Nelle applicazioni che processano suoni in tempo reale una bassa latenza è poi addirittura indispensabile, per registrare e per sentire in tempo reale quello che suoniamo perché quando suoniamo una nota dobbiamo sentire subito la risposta in cuffia

Come ridurre al minimo la lunghezza di questo buffer senza perdere la caratteristica real time delle applicazioni?

Diamo a quelle parti dei programmi che hanno bisogno di risorse audio le massime prestazioni disponibili

- Garantendo a questi programmi elevate prestazioni costanti, possiamo senza pericolo ridurre la lunghezza del buffer sonoro. Il programma che fornisce e preleva le informazioni audio viene infatti eseguito con continuità e ha a disposizione le risorse necessarie.
- Questi programmi che scrivono o leggono audio dovranno però avere il permesso di rubare risorse agli altri programmi in esecuzione quando ne hanno bisogno
- E queste risorse dovranno poi essere restituite al resto del sistema nei tempi morti, altrimenti avremmo un sistema bloccato

A chi chiedere le risorse necessarie?

Non basta un processore più potente. E' necessario uno scheduler, programma che si occupa di assegnare le risorse disponibili ai numerosi programmi in esecuzione sui nostri sistemi, modificato

- Normalmente un buono scheduler deve concedere ad ogni processo e ad ogni programma le risorse di cui ha bisogno con continuità nel tempo in maniera equa
- Noi invece abbiamo bisogno di uno scheduler speciale che tratti in maniera diversa quelle parti programmi che devono accedere all'audio. Deve garantire loro le massime prestazioni disponibili, anche quando il carico di lavoro della cpu è elevato

In pratica?

Abbiamo bisogno di uno scheduler nuovo, ok ma lo scheduler fa parte del kernel! Dobbiamo forse cambiare kernel?

- No, il kernel linux 2.6.20 o superiore include già uno scheduler adatto alle nostre esigenze!
- Tuttavia, poiché durante il normale uso del computer è preferibile uno scheduler più democratico, e meno selettivo, questo speciale componente di solito non viene installato
- Dobbiamo quindi procurarci una copia del kernel adattata alle nostre esigenze

Compilare o non compilare?

Compilare cioè preparare un kernel apposito oppure cercare una distribuzione di linux che installi questo componente di già?

- Compilare un kernel è un'operazione lunga e tediosa e non indicata agli utenti alle prime armi
- E' preferibile rivolgersi a distribuzioni di linux dedicate a queste particolari esigenze

Linux e Audio Real time

- Tra le varie distribuzioni dedicate disponibili la nostra scelta cade su ubuntu-studio, che riesce a raggiungere latenze molto basse in un comodo desktop, quello di ubuntu, solido e molto apprezzato dalla comunità open



Illustration 2: Installazione di ubuntu studio

Requisiti Hardware

Hardware

Cosa serve per registrare con Ubuntu Studio?

■ Il dvd di Ubuntu Studio

Potete scaricare una copia di ubuntu studio direttamente dal sito ufficiale <http://ubuntustudio.org/>. Scompattate su un disco dvd il file scaricato ed avrete una copia pari pari del dvd di ubuntu studio. Se non sapete come effettuare questa operazione di scompattamento, potete scaricare gratuitamente il programma *infrarecorder* dal sito <http://infrarecorder.sourceforge.net/>, e scegliere *Masterizza Immagine Dvd* dal menù Azioni.

Se avete già una copia di ubuntu funzionante e volete aggiornare il vostro ubuntu a ubuntu studio, vi basta installare il pacchetto di base di ubuntu studio. Su ubuntu gutsy le righe di terminale che seguono effettuano l'operazione richiesta. Notare che vi verrà chiesta la password di sistema per procedere.

```
sudo apt-get install ubuntu-studio-desktop
```

■ Un personal computer compatibile

Quasi tutti i personal computer nuovi o vecchi che siano funzionano, più o meno bene, con linux. Capita però che certe potenzialità del proprio computer siano difficili da sfruttare in ambiente open. Per questo è consigliabile scegliere solo componenti pienamente supportati dalla comunità open. Dell per esempio vende personal computer già perfettamente configurati per funzionare con ubuntu linux. Maggiori informazioni sono disponibili all'indirizzo internet <http://www.dell.com/open>

■ Una buona scheda audio compatibile

E' possibile consultare una lista delle schede che supportate sul sito internet del progetto *alsa*. L'indirizzo di riferimento è <http://www.alsa-project.org/main/index.php/Matrix:Main>. Le schede supportate sono moltissime e includono una grande varietà di schede sonore integrate. Molto probabilmente il vostro computer è già dotato di una scheda sonora di questo tipo. Il mio consiglio tuttavia è quello di aggiornare il vostro computer, comperando una scheda sonora nuova di classe superiore. Le schede sonore *Creative Soundblaster live!* e *Creative Audigy* svolgono egregiamente compiti di registrazione e di riproduzione a livello amatoriale. In più sono disponibili ad un prezzo veramente modico. Per chi invece avesse maggiori esigenze di mobilità e di flessibilità, un'ottima scheda audio è la *M-Audio Audiophile USB*, che permette di registrare contemporaneamente fino a 4 distinte sorgenti audio.

- **Cavo jack e adattatore jack mini jack per gli strumenti amplificati**
- **Uno o più microfoni di qualità più mixer per tutti gli altri strumenti**

Per registrare il basso o la chitarra elettrica tutto quello che serve è un cavo jack da 3 o più metri e un adattatore da jack a mini jack, che serve per collegare il cavo jack all'ingresso di linea del computer. Per registrare la voce e più in generale gli strumenti non amplificati è necessario invece un buon microfono. Forse avete un microfono integrato nel vostro computer o avete comperato un piccolo microfono a clip da computer. Questa classe di microfoni non è assolutamente adatta alla registrazione. E' necessario un buon microfono a cavo jack o a cavo cannon e di un mixer, che porti il debole segnale microfonico ad un livello adatto alla registrazione. Poi si può collegare il mixer al computer con un cavo rca mini jack. Se non avete un mixer e non intendete acquistarne uno, esistono schede sonore che integrano già un piccolo mixer, come per esempio la già citata M-Audio USB alla quale si può collegare direttamente un microfono con cavo cannon o jack.

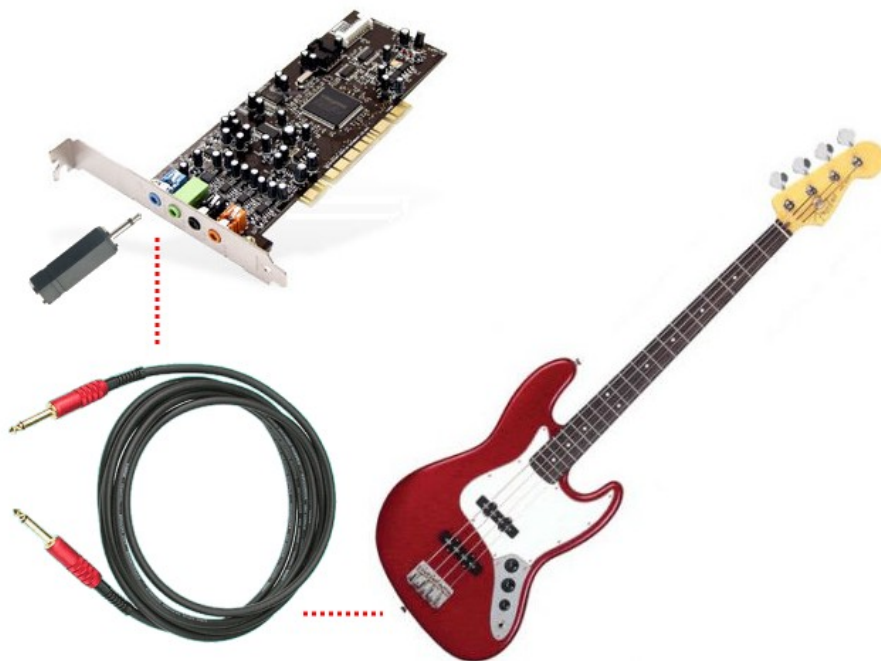


Illustration 3: Esempio di connessione tra strumento e scheda sonora

- **Cuffie chiuse “da dj” per registrare**
- **Due casse a due vie per riprodurre**

E' indispensabile, quando si registra, avere un monitor di registrazione, cioè avere la possibilità, mentre si suona, di sentire quello che si sta registrando. Un buon paio di cuffie sono la soluzione ideale. Ci sono in commercio una grande varietà di cuffie diverse. Le più adatte al nostro scopo sono le cuffie chiuse “da dj”, perché offrono un'elevata intensità sonora e un buono isolamento dai rumori dell'ambiente circostante. Dopo la registrazione si passa di solito all'ascolto della stessa su un buon paio di casse. Ora, musica e registrazioni si ascoltano meglio quando si hanno due (e solo due) casse a due vie. Cioè due casse dotate singolarmente di un cono piccolo per le note più alte e di un cono grande per le note più basse. Configurazioni composte da subwoofer centrale e satelliti laterali sono poco indicate per queste applicazioni. Queste configurazioni sono pensate per la visione di filmati e per la riproduzione di effetti speciali, ma non si comportano bene con la musica, perché separano note spostandole da un canale all'altro, senza rispettare fedelmente le informazioni audio della sorgente.

Requisiti Software

Programmi

Tutti i programmi qui citati si possono scaricare gratuitamente nei rispettivi siti internet, oppure direttamente dai repository di ubuntu studio

■ Jack

Jack è un vero e proprio cavo virtuale. Con jack si può collegare l'entrata della scheda sonora al programma che usiamo per la registrazione e registrare, oppure l'uscita del programma che usiamo per ascoltare la musica alle cuffie ed ascoltare. Ma, al pari del cavo reale, con jack si possono realizzare anche connessioni più complesse. Per esempio, si può prendere l'ingresso della scheda sonora, collegarlo ad un programma che equalizza il suono e poi attaccare questo programma equalizzatore ad un altro programma che distorce il suono e così via. Abbiamo così la possibilità di combinare tra di loro i nostri programmi come se fossero tante pedaliere o tanti rack di effetti virtuali.

Questo cavo virtuale è un cavo di qualità: è progettato specificamente per la bassa latenza e le applicazioni real time. Jack infatti non introduce alcuna latenza extra quando viene avviato. Se la latenza era 2ms prima di lanciare il server jack, sarà 2ms anche dopo aver lanciato il server jack. Poiché questo programma si occupa poi in maniera esclusiva del controllo degli ingressi e delle uscite della scheda sonora, la sua presenza ottimizza il funzionamento dei programmi garantendo loro le risorse di cui hanno costantemente bisogno

■ Qjackctl (Jack Control)

Qjackctl serve per controllare jack. Con questo programma, chiamato a volte anche “Jack-Control”, si può decidere la latenza, modificare i collegamenti audio, e controllare il buon funzionamento del sistema audio

■ Jack-Rack

Jack-Rack è un vero e proprio rack di effetti virtuali. Questo programma carica tutti gli effetti che abbiamo in libreria, quali riverbero, distorsori ed equalizzatori, e ci permette poi comodamente di controllarli, accendendoli, spegnendoli oppure modificandone le impostazioni

■ Caps, Tap-plugins, Swh-plugins, Dssi-example-plugins

Caps, Tap-plugins, Swh-plugins, Dssi-example-plugins sono collezioni di effetti compatibili con jack-rack e che possiamo usare per modificare i nostri suoni come più preferiamo

■ **Jamin**

Jamin è un ottimo programma per processare suoni in tempo reale. Questo programma analizza graficamente lo spettro delle frequenze presenti nella sorgente in ingresso e le mostra su schermo. Permette anche di equalizzare sempre graficamente il suono, di amplificare il suono per portarlo al giusto livello. Integra anche un compressore sonoro con varie curve di compressione

■ **Timemachine**

Timemachine è un programma per registrare semplicissimo da usare: ha un solo grande pulsante verde. Cliccando sopra a questo pulsante comincia la registrazione. Timemachine integra anche un indicatore di livello con il quale possiamo capire se sta ricevendo o meno segnali in ingresso

■ **Gtick**

Gtick è un metronomo dotato di tutte le funzioni che si possono chiedere ad un metrono. Gtick non dipende dal server jack, questo vuol dire che non può essere controllato tramite jack-control e che suonerà sempre sull'uscita principale indipendentemente dai collegamenti che abbiamo effettuato

■ **Lingot**

Lingot è un accordatore molto preciso e molto comodo da usare. Anche Lingo non dipende dal server jack, questo significa che non può essere controllato tramite jack-control e che intonerà sempre il suono proveniente dall'ingresso principale della scheda sonora, indipendentemente dai collegamenti che abbiamo effettuato

■ **Audacity**

Audacity è un ottimo programma per modificare audio multitraccia. Lo possiamo usare per ritoccare le nostre registrazioni oppure per mescolare insieme le registrazioni di un intero gruppo e realizzare un demo completo

■ **K3b**

Con K3b possiamo masterizzare i nostri lavori su cd. Molto potente e flessibile, ha un'interfaccia pulita e ricca di opzioni

in Pratica

Configurare Ubuntu Studio

Configurare Ubuntu Studio dopo aver installato seguendo le istruzioni contenute nel dvd o dopo aver aggiornato Ubuntu a Ubuntu Studio

- Non è necessario aggiustare alcun driver aggiuntivo: se avete hardware compatibile tutto viene riconosciuto automaticamente da ubuntu
- Tutto quello che bisogna fare è installare i programmi che ci servono
- Tutti i programmi sono già presenti nei repository di Ubuntu Studio. Possiamo quindi installare tutto quello che ci manca con un semplicissimo comando da terminale. Avviate un terminale (Applicazioni>Accessori>Terminale) e incollate le seguenti righe. Dopo aver incollato e avviato il comando con il tasto invio, vi verrà chiesta la vostra password di sistema, perché stiamo modificando la nostra installazione di Ubuntu Studio

```
sudo apt-get install linux-rt jack qjackctl ubuntustudio-audio-  
plugins jack-rack jamin timemachine gtick lingot audacity k3b
```

Lingot

Cominciamo! Avviamo Lingot (si trova in Applicazioni>Audio e Video>Lingot)

- Se Lingot non è presente nell'elenco delle applicazioni Audio e Video, lo possiamo lanciare da terminale (Applicazioni>Accessori>Terminale) con il comando

```
lingot &
```

- Teniamo aperto lingot, ci sarà comodo per intonare lo strumento che stiamo suonando e per verificare che arrivi effettivamente segnale audio al programma dalla nostra scheda sonora

Configurare la scheda sonora

Dopo aver collegato il nostro strumento musicale all'ingresso di linea della scheda sonora, modifichiamo le impostazioni della scheda sonora fino a che Lingot non ci segnala la presenza di suono in ingresso

- Per accedere alle opzioni della scheda sonora, cliccare due volte sulla piccola icona dell'altoparlante posta in alto a destra nel desktop di ubuntu

- Dal menù file scegliere il Device che vogliamo configurare. A volte ad una singola scheda sonora corrispondono più device, in questo caso scegliere il device che presenti tutte le opzioni che ci servono. Sbloccare poi dal menù (Modifica>Preferenze) tutte le opzioni di registrazione presenti.
- Ora nel programma di configurazione della scheda sonora, spostarsi alla scheda registrazione e portare all'80% tutti gli indicatori presenti. Infine commutare la registrazione da un canale all'altro, cliccando sulla piccola icona in basso del microfono. Per la scheda sonora creative soundblaster live! bisogna togliere il muto ad AC97 Cattura
- Se ancora la lancetta di lingot non dà segni di vita, spostarsi nella scheda interruttori e commutare Aux Cattura su Microfono Cattura o viceversa oppure sulle altre opzioni di Cattura disponibili. Se ancora la lancetta non si muove, provare a collegare il jack dello strumento nelle altre prese libere della scheda sonora.
- Ora possiamo intonare il nostro strumento! Eheh

Configurare qjackctl e lanciare jack

Proseguiamo lanciando il server jack con il programma qjackctl (si trova in Applicazioni>Audio e Video>Jack-Control)

- Viene mostrata una piccola finestra dal titolo jack audio connection kit. Clicchiamo sul pulsante Setup e configuriamo il server jack. Le opzioni più importanti sono evidenziate in rosso.

Linux e Audio Real time

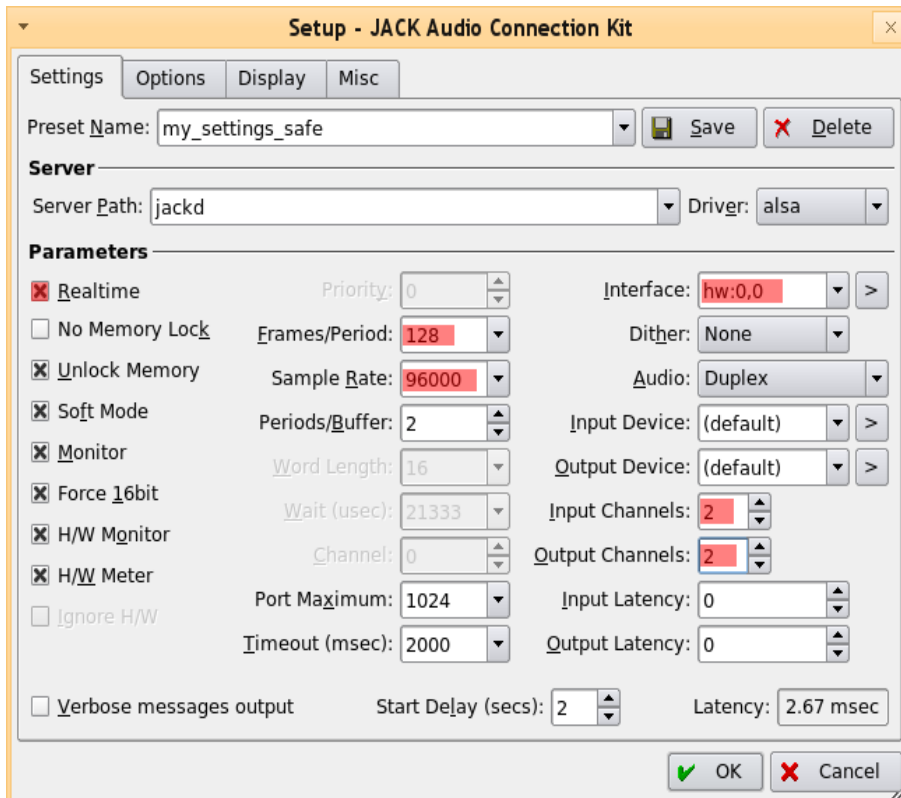


Illustrazione 1: le opzioni di avvio del server Jack in Jack-Control

- **Realtime**

Spuntando questa opzione le prestazioni di jack migliorano sensibilmente

- **Frames/Period**

Questo valore controlla la lunghezza del buffer. Inserendo qui un valore piccolo abbiamo una latenza bassa. Tuttavia valori troppo bassi di questa opzione rendono jack instabile. L'instabilità di jack causa poi xrun, cioè interruzioni temporanee del flusso sonoro mentre il computer si ferma a caricare. Dobbiamo quindi cercare di abbassare al minimo questo valore, senza che si verifichino mai xrun. In questo caso si è scelto il valore 128.

- **Sample Rate**

Il Sample Rate controlla la quantità di informazione che viene prelevata per ogni secondo di musica che suoniamo o ascoltiamo. Valori più alti danno una maggiore qualità audio, per cui è raccomandabile scegliere il massimo valore disponibile. In questo caso si è scelto il valore 96000.

- **Interface**

Qui si sceglie quale scheda sonora si preferisce usare. Cliccando sulla piccola freccia sulla destra vengono mostrate tutte le schede disponibili. A volte, per una stessa scheda sono disponibili più opzioni. Spesso si tratta di opzioni che sbloccano certe funzionalità a scapito di altre. In questo caso ad esempio si è scelto il valore hw:0:0 corrispondente all'opzione ADC Capture/Standard PCM Playback della scheda sonora SBLive 5.1, che consente di ottenere un Sample Rate elevato, ma con solo due canali in ingresso e in uscita.

- **Input Channels**

Il numero di canali che vogliamo registrare contemporaneamente. Ha senso scegliere più di due canali solo se la scheda sonora è in grado di supportarli. In questo caso, è stato scelto il valore 2.

- **Output Channels**

Il numero di canali che vogliamo ascoltare contemporaneamente. In questo caso è stato scelto il valore 2, tuttavia a volte potrebbero essere utili valori superiori al due, per esempio per controllare contemporaneamente cuffie e casse. Nel caso della soundblaster live! i valori consentiti sono 2 e 16, mentre non sono consentiti valori come 4 o 6. In caso di dubbio impostate pure il valore più alto consentito dalla vostra scheda.

- **Latency**

In fondo a destra nella finestra di opzioni c'è un piccolo numero, etichettato Latency, che possiamo consultare per verificare quanto le nostre impostazioni influiscano sulla latenza complessiva.

Linux e Audio Real time

- Dopo aver configurato le impostazioni del server jack possiamo cliccare su OK. La finestra si chiude e, nella piccola finestra di controllo, possiamo ora cliccare sul grande pulsante verde start ed avviare il server jack. Se dovessero comparire messaggi di errore o messaggi di xrun, fermate il server jack con il pulsante stop, modificate le opzioni di avvio del server jack come visto sopra e riavviate. Spesso è necessario effettuare più di una prova prima di ottenere la configurazione ideale.
- Dopo aver avviato jack, riduciamo ad icona qjackctl e lasciamolo in esecuzione, ci servirà più avanti per configurare le connessioni

Jamin e le prime connessioni con Jack-Control

Ora possiamo suonare: avviamo Jamin (Applicazioni>Audio e Video>JAMin)

- Ci si presenta davanti una grande finestra dominata da un grande equalizzatore grafico a sfondo nero. Tuttavia, per quanto suoniamo, non si vede alcun indicatore alzarsi, neanche aumentando il guadagno con l'indicatore evidenziato in rosso.

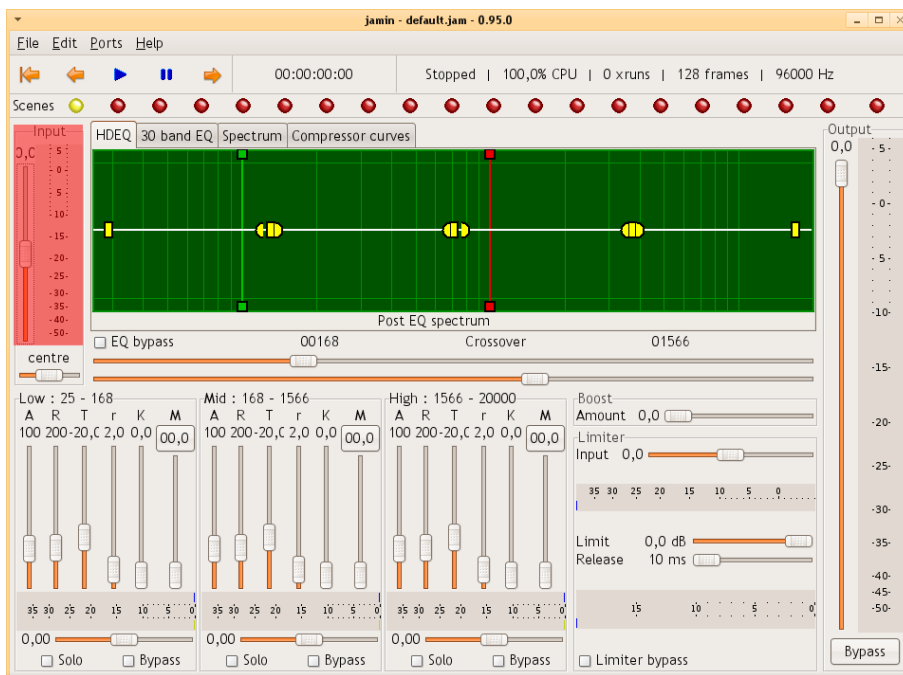


Illustrazione 2: Jamin

- Questo perché non sono ancora state realizzate le dovute connessioni all'interno di jack. Ritorniamo a considerare il programma jack-control che avevamo ridotto ad icona. Dopo averlo ripristinato clicchiamo sul grande pulsante con icona rossa, Connect. Si apre una finestra che mostra le connessioni attuali. Facciamo doppio click su `alsa_pcm` per espandere. Poi aggiungiamo una nuova connessione trascinando `capture_1` di `alsa_pcm` (a sinistra) su `in_L` di `jamin` (a destra). Eseguiamo la stessa operazione tra `capture_1` di `alsa_pcm` (a sinistra) e `in_R` di `jamin` (a destra).

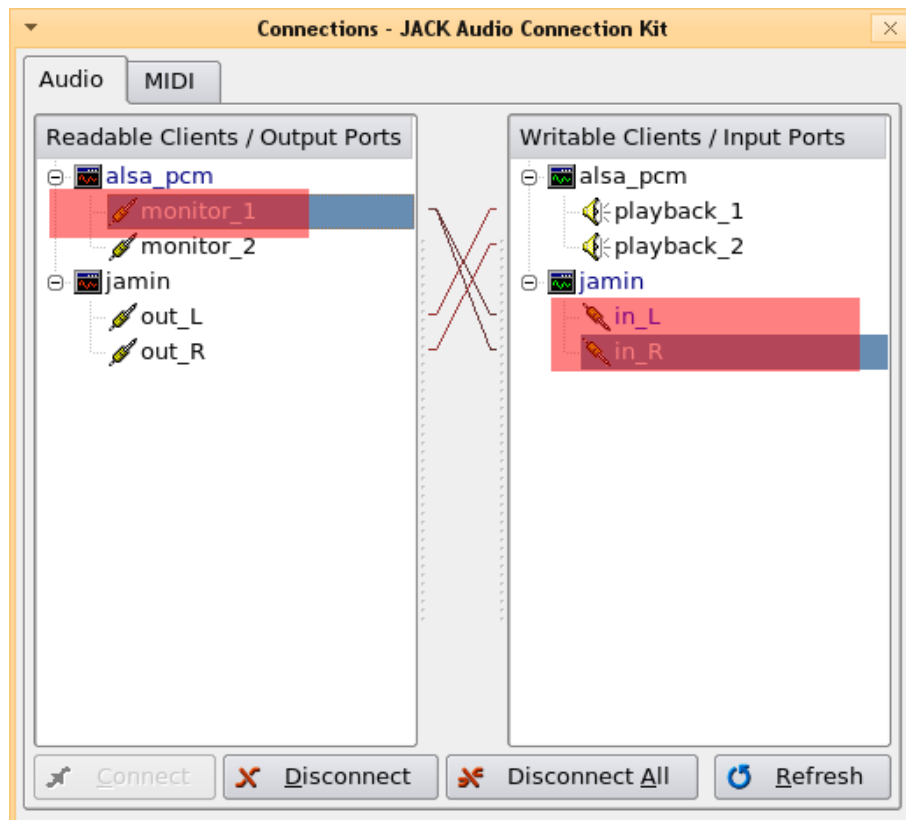


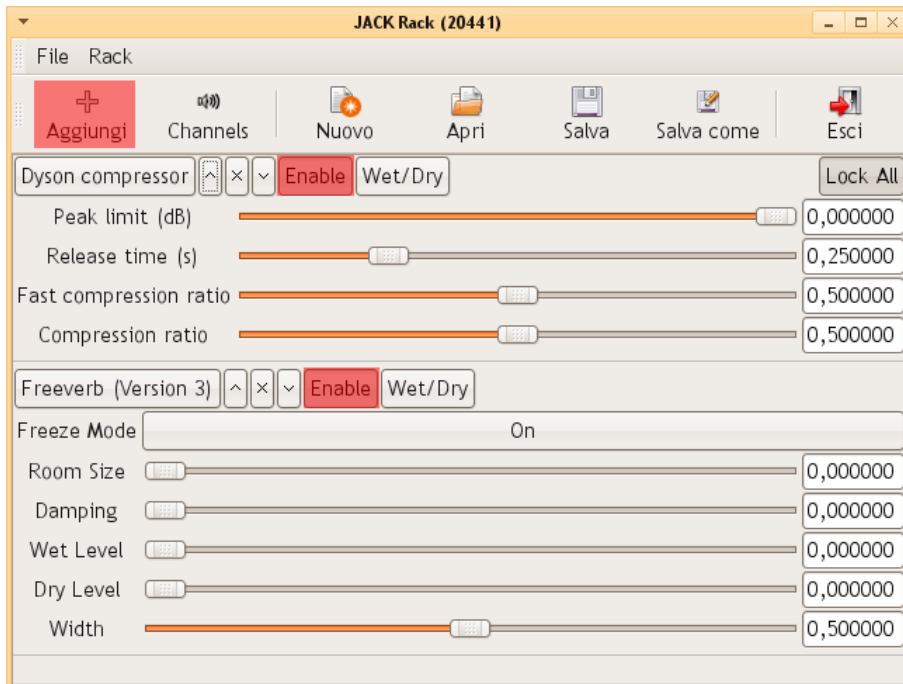
Illustrazione 3: Controllare le connessioni con Jack-Control

- Infine colleghiamo, se non sono già collegate, out_L di jamin (a sinistra) a playback_1 di alsa_pcm (a destra) e out_R di jamin (a sinistra) a playback_2 di alsa_pcm (a destra)
- Ora abbiamo a disposizione un monitor in cuffia per sentire cosa stiamo suonando. Aumentiamo il volume di ingresso di jamin con il controllo Input, evidenziato in rosso nella seconda Illustrazione, fino a che il livello sonoro risulta soddisfacente
- Mentre suoniamo nel riquadro scuro viene continuamente ridisegnata una rappresentazione grafica della forma d'onda in ingresso. Si tratta di un monitor visivo molto utile. Disegnando infatti sulla curva bianca al centro dell'illustrazione possiamo enfatizzare o cancellare le frequenze che ci interessano per ottenere un suono più adatto alle nostre esigenze

Jack-rack e gli effetti

Con Jack-rack (Applicazioni>Audio e Video>JACK Rack) possiamo aggiungere effetti di grande impatto sonoro

- Carichiamo gli effetti che ci interessano cliccando sul grande pulsante con la croce Aggiungi. Ce ne sono veramente tanti. Per cominciare, carichiamo Dyson Compressor (Aggiungi>Amplitude>Dynamics>Compressor>Dyson Compressor) e Freeverb (Aggiungi>Uncategorized>F>Freeverb)



- Ora dobbiamo abilitare i singoli effetti, cliccando sul pulsante Enable a destra del nome di ogni effetto
- Infine, dopo aver configurato Jack-Rack, passiamo alla gestione delle connessioni. In questo caso vogliamo collegare Jack-Rack in serie a Jamin, cioè facciamo agire gli effetti dopo che il segnale è stato equalizzato e portato a livello da Jamin
- Apriamo quindi nuovamente la finestra Connections di Jack Control (da Jack Control, cliccare su Connect). Rompiamo le connessioni attuali con il pulsante Disconnect All e realizziamo le seguenti connessioni. Colleghiamo il monitor_1 di alsa_pcm (a sinistra) con jamin (a destra). Poi colleghiamo jamin (a sinistra) con jack Rack (a destra, jack Rack è solitamente seguito da un numero, come ad esempio jack_Rack_20441). Infine colleghiamo jack-Rack (a sinistra) con alsa_pcm (a destra).
In alternativa possiamo rompere la connessione esistente tra jamin (a sinistra) e alsa_pcm (a destra) selezionandoli entrambi e cliccando su Disconnect, e proseguire poi con le connessioni mancanti

Gtick

Per registrare bene può essere necessario un metronomo. Avviamo gtick (Applicazioni>Audio e Video>Gtick)

- Gtick è veramente intuitivo: dalla finestra principale scegliamo il tempo che preferiamo e poi clicchiamo su Avvia. Possiamo interrompere il metronomo cliccando su Arresta
- Gtick non fa uso di Jack, per cui non è necessario ridefinire le connessioni in Jack-Control

Timemachine per registrare

Passiamo alla registrazione. Lanciamo Timemachine (Applicazioni>Audio e Video>Timemachine)

- Timemachine si presenta con un grande pulsante verde e un piccolo indicatore di livello sotto.
- Possiamo far partire la registrazione in ogni momento cliccando sul grande pulsante verde e fermarla cliccando sul pulsante che durante la registrazione è diventato rosso. Tuttavia prima di registrare dobbiamo stabilire le giuste connessioni
- Torniamo al programma Jack-Control e alla schermata delle connessioni (accessibile cliccando sul pulsante connect). Stabiliamo una nuova connessione tra jack_rack (a sinistra) e timemachine (a destra). Ora l'indicatore di volume di Timemachine dovrebbe segnalare la presenza di suono
- Timemachine registra in un formato particolare, chiamato w64, nella nostra cartella home (cioè /home/il_vostro_nome_utente). Possiamo personalizzare Timemachine e costringerlo a registrare in formato wav avviandolo da un terminale (Applicazioni>Accessori>Terminale) con questo comando. Con altre opzioni da riga di comando si può impostare anche la cartella dove timemachine registra i clip audio

```
timemachine -f wav &
```

Audacity

Infine possiamo modificare i nostri clip con il programma Audacity (Applicazioni>Audio e Video>Audacity Sound Editor)

- Prima di tutto configuriamo Audacity per funzionare con Jack. Dal menù Preferenze (Modifica>Preferenze) scegliamo come device di riproduzione Jack Audio Connection Kit: alsa_pcm e come device di registrazione Jack_Audio_Connection_Kit: jack_rack
- Apriamo il nostro campione registrato precedentemente da Timemachine cliccando su File>Apri e cercando un file il cui nome inizia per tm, seguito dalla data di registrazione (per esempio tm-2007-10-07T14:40:51.wav) nella cartella home (cioè /home/il_vostro_nome_utente)
- Ora possiamo riascoltare il nostro file salvato, e se vogliamo possiamo suonarci sopra. Possiamo pure registrare sopra al file salvato all'interno da Audacity, cliccando sul grande pulsante rosso Record
- Se capita che la configurazione con jack di Audacity dia dei problemi, perché è troppo pesante, possiamo chiedere ad Audacity di usare direttamente alsa, dal menù device accessibile in (Modifica>Preferenze). Tuttavia così avremo una latenza maggiore e non potremmo sincronizzare riproduzione e registrazione

K3b

Quando abbiamo bisogno di masterizzare i nostri lavori, avviamo K3b (Applicazioni>Audio e Video>K3b)

- Dalla finestra di K3b, clicchiamo su Nuovo Progetto Cd Audio e trasciniamo i file audio che vogliamo masterizzare su cd nella finestra CDAudio0.
- Controlliamo che i file si adattino alla dimensione del cd
- Infine cliccando sul pulsante Scrivi masterizziamo il cd. Normalmente non è necessario configurare alcuna opzione aggiuntiva e si possono accettare tranquillamente le configurazioni predefinite.

Problemi comuni

- **Ho installato ubuntu con il kernel realtime ma jack si rifiuta di funzionare in modalità realtime. L'avvio fallisce con questo messaggio di errore**

```
cannot use real-time scheduling (FIFO at priority 10) [for thread
-1210194240, from thread -1210194240] (1: Operation not permitted)
```

La soluzione gruppo audio

Il problema si presenta perché non abbiamo creato i permessi speciali che permettono a jack di funzionare in modalità real time. Lanciamo questi comandi da terminale

```
sudo su -c 'echo @audio - rtprio 99 >> /etc/security/limits.conf'

sudo su -c 'echo @audio - memlock 250000 >>
/etc/security/limits.conf'

sudo su -c 'echo @audio - nice -10 >> /etc/security/limits.conf'
```

Dopodiché usciamo dalla sessione corrente (un click sul pulsante rosso in alto a destra e un click sull'opzione Termina Sessione). Infine rientriamo nel desktop di ubuntu. Ora siamo in grado di avviare jack in modalità real time

- **Il sistema funziona, ma è scomodo dover avviare uno per uno tutti questi programmi ogni volta. Si può semplificare la procedura?**

Sì, è possibile automatizzare e quindi rendere più veloce e più semplice il caricamento delle applicazioni audio che utilizziamo con frequenza. Per prima cosa, facciamo una piccola modifica alle opzioni di jack-control.

Jack-Control

All'interno di Jack-Control, clicchiamo sul pulsante Setup. Si apre il menù con le opzioni di avvio di Jack. Spostiamoci alla quarta delle schede di opzioni disponibili, che si chiama Misc. Spuntiamo l'opzione "Avvia il server jack all'avvio", dopodiché salviamo le impostazioni personalizzate e chiudiamo Jack-Control. Ora ogni qual volta avviamo jack-control, viene automaticamente avviato anche il server jack.

Jack-All

Dopo le modifiche fatte alla configurazione di Jack-Control, si può creare un piccolo script che si occupi di avviare per noi tutte le applicazioni di cui abbiamo bisogno. Apriamo gedit (Applicazioni>Accessori>Editor di testi) e incolliamo in gedit queste righe di codice

Linux e Audio Real time

```
#!/bin/sh
qjackctl &
sleep 3
jack-rack &
jamine &
timemachine -f wav &
```

Salviamo il documento sul Desktop (File>Salva) con il nome jack-all e chiudiamo. Ora dobbiamo cambiare le proprietà di questo documento jack-all in modo da renderlo avviabile. Per fare ciò, sul Desktop, accediamo al menù Proprietà di jack-all con un click con il tasto destro sul file jack-all. Spostiamoci nella scheda Protezioni e da lì accendiamo l'opzione Esecuzione.

Fatto: ora possiamo comodamente avviare tutti i programmi che ci servono con un semplice doppio click sull'icona del documento jack-all che abbiamo creato sul Desktop

Commenti e conclusioni

Con le ultime versioni di ubuntu e del kernel di linux, nonostante qualche difficoltà iniziale di configurazione, i programmi per registrare in ambiente libero stanno diventando una vera e propria alternativa ai costosi rivali a pagamento.

■ **Vantaggi**

Tra i principali vantaggi delle soluzioni per la registrazione in ambiente libero c'è il server jack. Non esiste nulla di equivalente in ambiente windows. Jack infatti permette di collegare tra di loro programmi diversi in cascata, di estenderne così le funzionalità e le possibilità e di integrare le funzioni che mancano in un programma con le funzioni di un altro programma. Possiamo separare i compiti tra vari programmi dedicati ad una specifica funzione.

Inoltre in ambiente libero si può scegliere tra numerose applicazioni per la produzione di musica. Molte delle quali non sono state citate qui. Per esempio, ardour, che permette di registrare e di ascoltare contemporaneamente più di una traccia alla volta, sovrapponendo strumenti sintetizzati a strumenti veri. Oppure rosegarden, per scrivere e stampare partiture in formato midi. Tuxguitar, per leggere i file tablatura nel popolare formato .gp3. E Mixxx che sostituisce piatti, mixer e dischi, una vera e propria stazione di lavoro virtuale per dj. Infine, l'altro grande vantaggio dell'ambiente libero linux è la possibilità di utilizzare hardware magari vecchio e dal costo contenuto, come per esempio la soundblaster live! più volte citata, al massimo delle sue potenzialità. E di ottenere così risultati buoni anche là dove per far funzionare le cose in ambiente windows o mac bisogna investire molto denaro in costose apparecchiature professionali

■ **Svantaggi**

Il grande svantaggio di questa serie di applicazioni libere utilizzate per registrare è la loro complessità. Spesso per capire l'utilità di un'opzione rispetto ad un'altra è necessario spendere molto tempo. E alle volte si incontrano errori o malfunzionamenti nelle applicazioni che devono ancora essere corretti, per aggirare i quali è necessario configurare manualmente numerosi parametri

■ **Conclusioni: è possibile registrare un demo con linux?**

Sì. La risposta forse è azzardata, perché ci sono ancora delle problematiche che rendono difficile la registrazione in ambiente libero. Tuttavia, ne sono convinto, si tratta di problemi marginali che si possono risolvere leggendo qualche guida e capendo il funzionamento interno delle applicazioni. E, ne sono altrettanto convinto, sono problemi di oggi che spariranno in futuro. Non solo: sono convinto che molto probabilmente le applicazioni libere di domani saranno addirittura più semplici delle rivali applicazioni a pagamento.

Il motivo di questa ostentata sicurezza è da ricercarsi nella passione e nell'impegno che la comunità libera riserva quotidianamente al miglioramento dei propri programmi. Passione e impegno che hanno portato già oggi ubuntu, il sistema operativo libero, ad essere un valido rivale per le pesanti e costose alternative a pagamento. E che porteranno domani le applicazioni audio libere a essere degne rivali delle agguerrite alternative a pagamento