



Applicazioni Bluetooth con Linux

Stefano Sanna

<http://www.gerdavax.it>



Agenda

- Bluetooth: come e perché!
- Il protocollo Bluetooth e la sua implementazione
- Giochiamo con il protocollo :-)
- Applicazioni
 - Connessione ad un cellulare
 - File transfer
- DEMO
- Bibliografia e Conclusioni





Bluetooth

- Bluetooth è una tecnologia wireless per l'interfacciamento di dispositivi su canale radio a 2.4GHz in modalità spread spectrum (FHSS)
- Semplificando, Bluetooth sostituisce la moltitudine di cavi e connettori con una interfaccia standard per il discovery e l'utilizzo di periferiche in ambito personale
- E' una specifica riconosciuta da tutti i sistemi operativi e produttori di hardware



Bluetooth in breve

- Caratteristiche
 - Modulazione FHSS, 79CH 1MHZ banda 2.45GHz
 - Identificazione device BD_ADDR a 6byte
 - Realizza reti di 8 dispositivi (1 master + 7 slave)
 - Trasmette dati a 723kpbs
 - Gestisce dati e voce
- Bluetooth fornisce:
 - Funzionalità di discovery per dispositivi e servizi
 - Un layer di comunicazione sufficientemente sicuro



A cosa serve?

- Accesso a periferiche personali:
 - *cable replacement* per cellulare, palmare, auricolare, access point PSTN, stampanti
 - scambio rapido di (piccoli) oggetti binari
- Accesso a periferiche d'ambiente:
 - impianto vivavoce in auto
 - controllo accessi e rilevatore di presenza
- Posizionamento:
 - connessione a GPS Bluetooth
 - localizzazione attraverso Bluetooth-beam



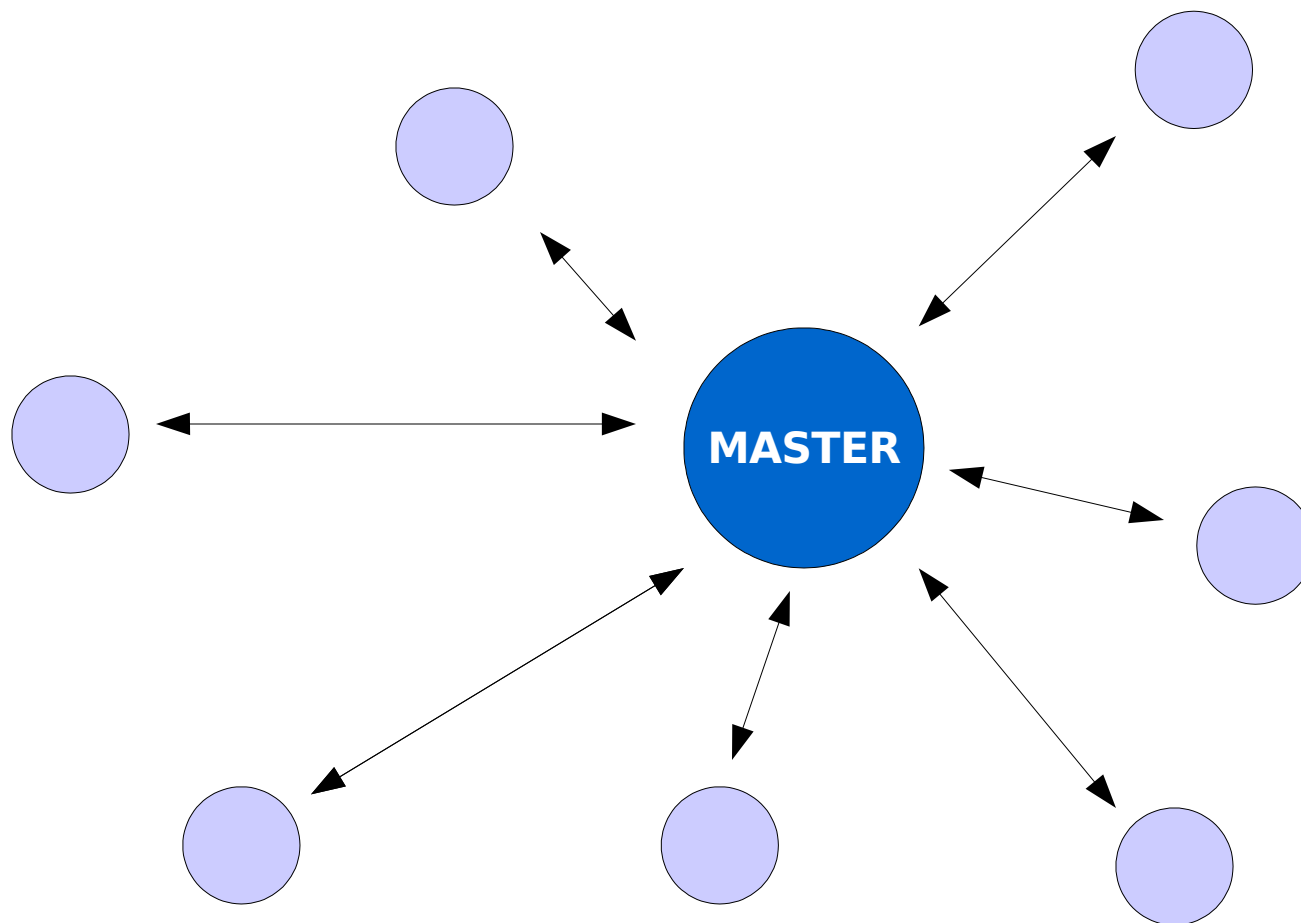
A cosa NON serve!

- Bluetooth è una tecnologia utilissima... se usata nel contesto giusto! Ad esempio, è sconsigliata per:
 - Collegamenti a lungo raggio
 - Reti veloci
 - Reti costituite da molti dispositivi
 - Reti esclusivamente dedicate al traffico IP

- Bluetooth da il meglio di sé su applicazioni personali, più legate all'interazione del singolo che alla realizzazione di vere e proprie infrastrutture di comunicazione

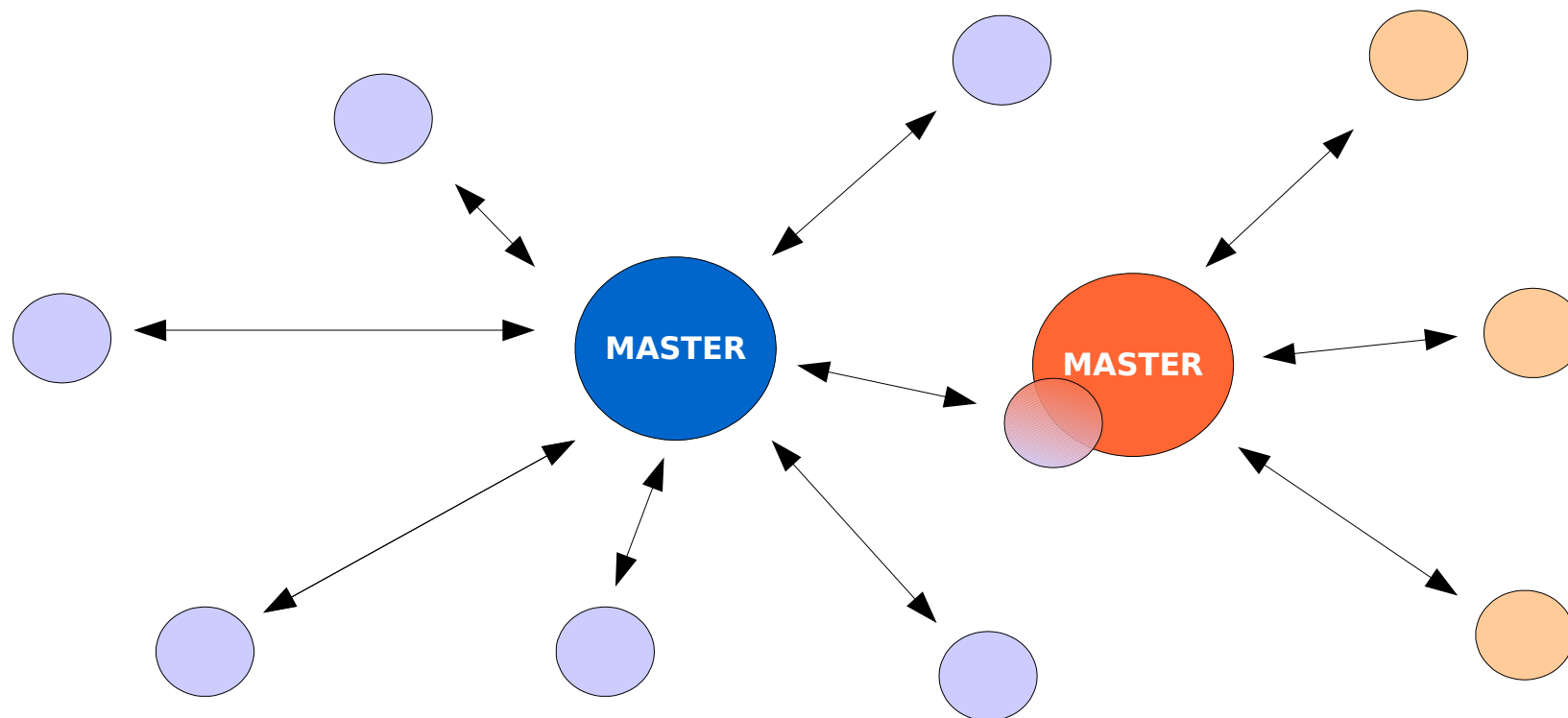


Piconet



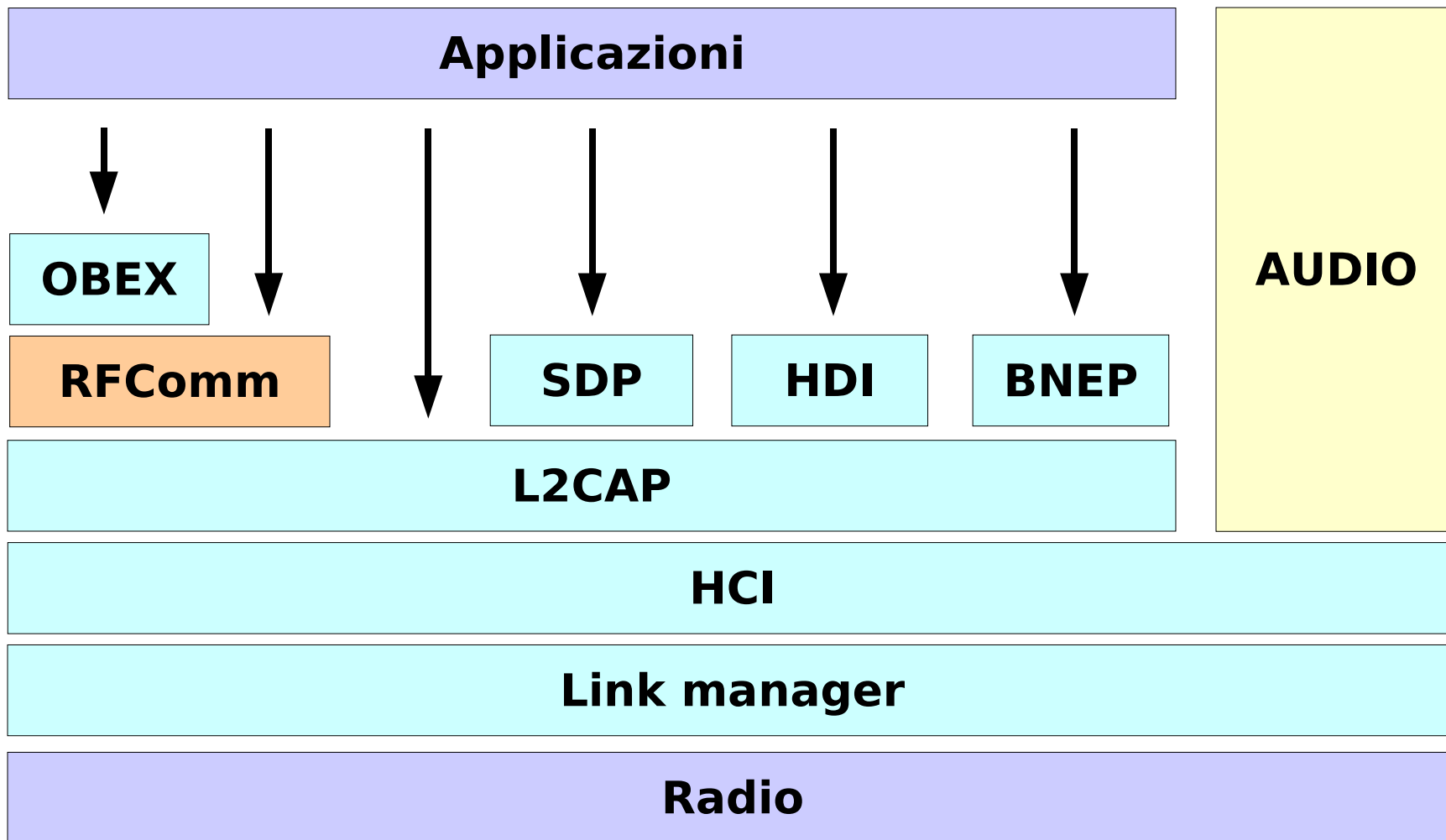


Scatternet





Lo stack Bluetooth





Protocolli e profili

- La specifica Bluetooth definisce questa tecnologia in termini di protocolli e profili (applicativi).
- Protocollo:
 - Descrive come sono codificati i dati scambiati tra due nodi della rete
- Profilo:
 - Descrive le funzionalità fornite dalle applicazioni
 - Profili diversi possono essere basati sugli stessi protocolli: essi differiscono nelle applicazioni alle quali offrono servizi (es: porta seriale)



Layer e Protocolli Bluetooth /1

- Link Manager
 - Gestisce la connessione a basso livello tra i device
- HCI (Host Controller Interface)
 - È l'interfaccia tra il modulo radio Bluetooth e il sistema operativo
- L2CAP (Logical Link Control & Adaptation Protocol)
 - Fornisce i canali logici di connessione (multiplex)
- SDP (Service Discovery Protocol)
 - Fornisce i meccanismi per il discovery di device e servizi



Layer e Protocolli Bluetooth /2

- RFCOMM
 - Fornisce una emulazione di porta seriale sul canale radio (simile a IrCOMM)
- BNEP (Bluetooth Network Encapsulation Protocol)
 - Permette di incapsulare pacchetti Ethernet su connessione Bluetooth
- TCS (Telephony Control Protocol Specification)
 - Fornisce i meccanismi di controllo delle funzioni telefoniche
- OBEX
 - Gestisce il trasferimento di oggetti tra device



Profili Bluetooth /1

- GAP (Generic Access Profile)
- SDAP (Service Discovery Access Profile)
- SPP (Serial Port Profile)
- HCRP (Hardcopy Cable Replacement)
- DUN (Dial-up Profile)
- LAP (LAN Access Profile)
- HS (Headset Profile)
- GOEP (Generic Object Exchange Profile)
- FTP (File Transfer Profile)



Profili Bluetooth /2

- SP (Synchronization Profile)
- HID (Human Device Profile)
- PAN (Personal Area Networking)
- BPP (Basic Printing Profile)
- Un device può supportare uno o più profili, corrispondenti alle funzionalità implementate da hardware e software.
- Cellulare: DUN, SP, SPP, GOEP...



Adattatori Bluetooth

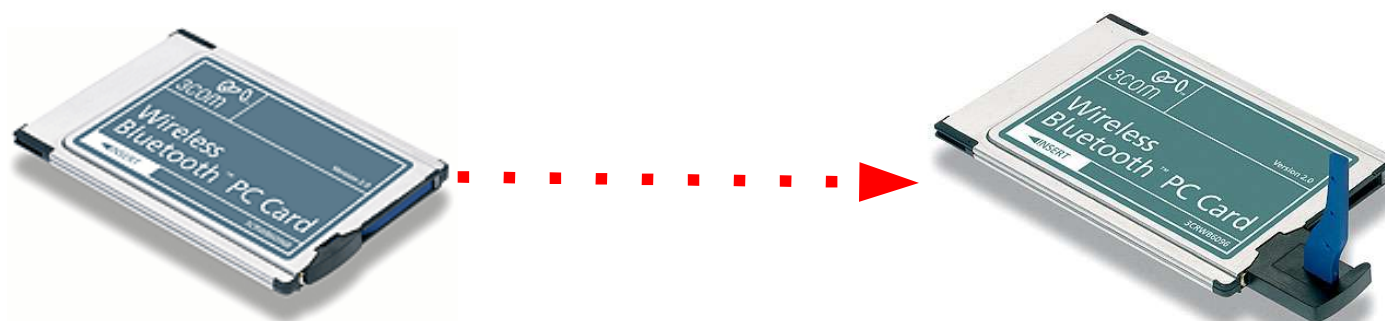
- Per utilizzare Bluetooth è necessario disporre di un opportuno modulo hardware che fornisce l'interfaccia radio e il firmware con gli elementi base dello stack
- Esistono diversi tipi di interfacce:
 - Seriale (poco diffusa)
 - USB
 - PCCard e CompactFlash
 - Integrate





Adattatori USB e PCMCIA

- Alcuni adattatori USB necessitano di un aggiornamento del firmware
- Sui portatili può essere comodo utilizzare un adattatore PCMCIA:
 - permette di avere una porta USB in più
 - offre una configurazione hardware “senza sporgenze”, che ne rende più sicuro il trasporto! :-)





Bluez

- Bluez è lo stack Bluetooth ufficiale per Linux
- Bluez fornisce driver, librerie e utility a linea di comando per l'accesso ai servizi base di Bluetooth
- I pacchetti:
 - **bluez-utils**: Bluetooth tools e daemons
 - **bluez-pin**: Bluetooth PIN helper con supporto D-BUS
 - **bluez-pcmcia-support**: supporto PCMCIA
 - **bluez-cups**: driver di stampa Bluetooth per CUPS
 - **bluez-hcidump**: per analizzare pacchetti Bluetooth
- Le distribuzioni recenti includono pacchetti e utility di Bluez



Driver e configurazioni

- Ci sono fastidiosi problemi di certificazione:

Latest information

On March, 9th 2005 the [Bluetooth Qualification Administrator \(BQA\)](#) informed me that a registered member of the Bluetooth SIG complained about the non-qualified use of Bluetooth products on this page and the [features document](#).

The current statement why I can't list the products that are known to be working with Linux goes like this:

Whether or not you're selling them makes no difference. The problem is due to the distribution of them from your Web site. Please note that the use and distribution of non-qualified products is a violation of the Bluetooth License Agreement. As neither of these products have been qualified using Linux it is illegal to make them available for public use.

It is quite hard for me to understand his point, because I am also not distributing any non-qualified Bluetooth products through these specific web pages. However they asked for the removal of these information and now all of them are gone

I am keeping the [features document](#), because it has nothing to do with Linux. These products are available on the market and thus all of them should be qualified. If the HCI Version field is filled in this table, then this device should also work perfect with Linux.

For further information contact [Member Relations](#) of the Bluetooth SIG.

Qualification

Since April, 11th 2005 the BlueZ protocol stack is [qualified as a Bluetooth subsystem](#). Companies can now use this listing to qualify their adapters with Linux support.

<http://www.holtmann.org/linux/bluetooth/devices.html>



Configurazione Bluez

- I file di configurazione risiedono su /etc/bluetooth:
 - **hcid.conf**
Contiene la configurazione principale dell'host controller interface, cioè dell'interfaccia Bluetooth della workstation
 - **pin**
Contiene il Personal Identification Number di default della workstation
 - **rfcomm.conf**
Contiene le impostazioni del mapping delle porte seriali



hcid.conf

[options/1]

```
options {
    # Automatically initialize new devices
    autoinit yes;

    # Security Manager mode
    # none - Security manager disabled
    # auto - Use local PIN for incoming connections
    # user - Always ask user for a PIN
    #
    security auto;

    . . . . .
}
```



hcid.conf

[options/2]

```

# Pairing mode
# none - Pairing disabled
# multi - Allow pairing with already paired devices
# once - Pair once and deny successive attempts
pairing multi;

# PIN helper
pin_helper /usr/bin/bluepin;

# D-Bus PIN helper
# dbus_pin_helper;
}
    
```



hcid.conf

[device/1]

```
device {
    # Local device name
    # %d - device id
    # %h - host name
    name "Kaa";

    # Local device class
    #class 0x100;

    class 0x120104

    # Default packet type
    #pkt_type DH1,DM1,HV1;
```



hcid.conf

[device/2]

...

```
# Inquiry and Page scan
iscan enable; pscan enable;
```

```
# Default link mode
```

```
# none - no specific policy
```

```
# accept - always accept incoming connections
```

```
# master - become master on incoming connections,
```

```
# deny role switch on outgoing connections
```

```
lm accept,master;
```

...



hcid.conf

[device/3]

```
# Default link policy
# none      - no specific policy
# rswitch  - allow role switch
# hold     - allow hold mode
# sniff    - allow sniff mode
# park     - allow park mode
```

```
lp rswitch,hold,sniff,park;
```

```
# Authentication and Encryption
```

```
#auth enable;
```

```
#encrypt enable;
```

```
}
```




Utility a linea di comando

- Bluez fornisce due comode utility per esplorare le caratteristiche di Bluetooth:
 - **hcitool**: permette di effettuare il discovery dei device Bluetooth
 - **sdptool**: mostra i servizi esportati da un device Bluetooth

- **hcid** è il demone responsabile della gestione dell'interfaccia Bluetooth. Avviandolo con l'opzione -n è possibile leggere in console gli eventi generati dal sistema



hcitool

dev	Display local devices
inq	Inquire remote devices
scan	Scan for remote devices
name	Get name from remote device
info	Get information from remote device
cmd	Submit arbitrary HCI commands
con	Display active connections
cc	Create connection to remote device
dc	Disconnect from remote device
sr	Switch master/slave role
cpt	Change connection packet type
rsi	Display connection RSSI
lq	Display link quality
tpl	Display transmit power level
afh	Display AFH channel map
lst	Set/display link supervision timeout
auth	Request authentication
enc	Set connection encryption



sdptool

Commands :

search	Search for a service
browse	Browse all available services
add	Add local service
del	Delete local service
get	Get local service
setattr	Set/Add attribute to a SDP record
setseq	Set/Add attribute sequence to a SDP record

Services:

SP DUN LAN FAX OPUSH FTRN HS HF NAP GN HID CIP CTP A2SRC A2SNK



OBEX

- **OB**ject **EX**change protocol per lo scambio di oggetti su una specie di http binario
- E' un protocollo nato dalla specifica IrDA per lo scambio di oggetti attraverso porta infrarossi
- Bluetooth lo ha adottato per lo scambio di vcard, calendar, messaggi, immagini, applicazioni e per il trasferimento in modalità ftp-like
- Il progetto **OpenObex** fornisce uno stack OBEX opensource per Linux! :-)

OpenOBEX
object exchange. open source.



OBEX Ftp e OBEX Push

- OBEX Ftp permette di trasferire file con modalità analoghe al File Transfer Protocol su TCP/IP:

obexftp -b BDADDR -l

- OBEX Push permette di trasferire oggetti binari (audio, immagini...) one-shot P2P:
 - OBEX Push Client: è il dispositivo che invia l'oggetto al peer remoto
 - OBEX Push Server: è il dispositivo che riceve l'oggetto



Utility Gnome

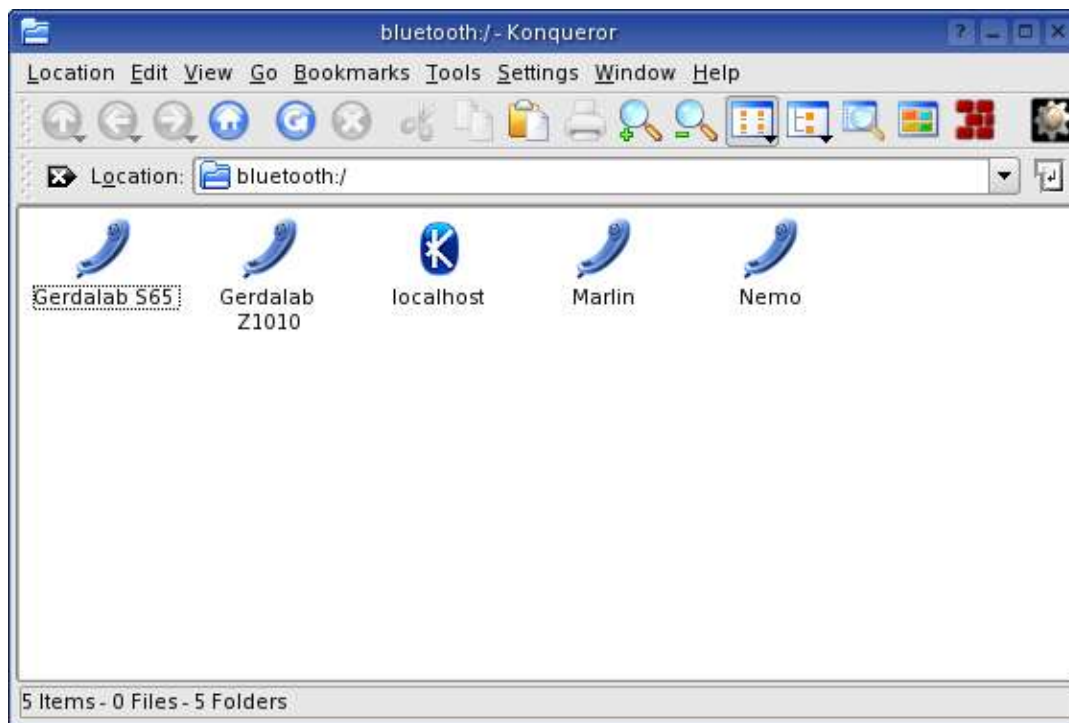
- Il pacchetto gnome-bluetooth fornisce le utility per il discovery dei dispositivi e la gestione del trasferimento file attraverso OBEX





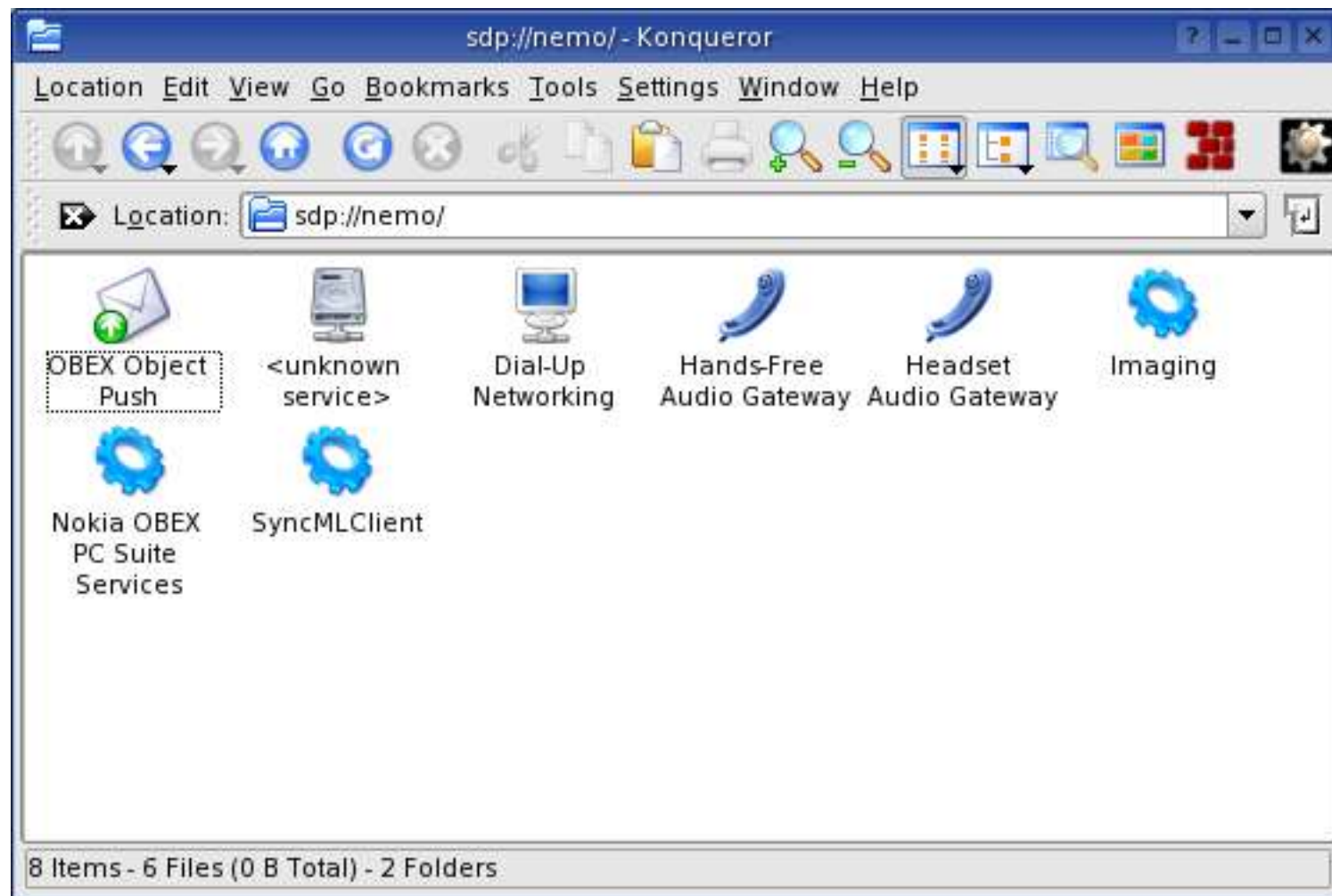
Utility KDE

- Il pacchetto kde-bluetooth offre il KDE Bluetooth Framework che fornisce utility per il discovery dei device e dei servizi, applet per la gestione dei servizi locali



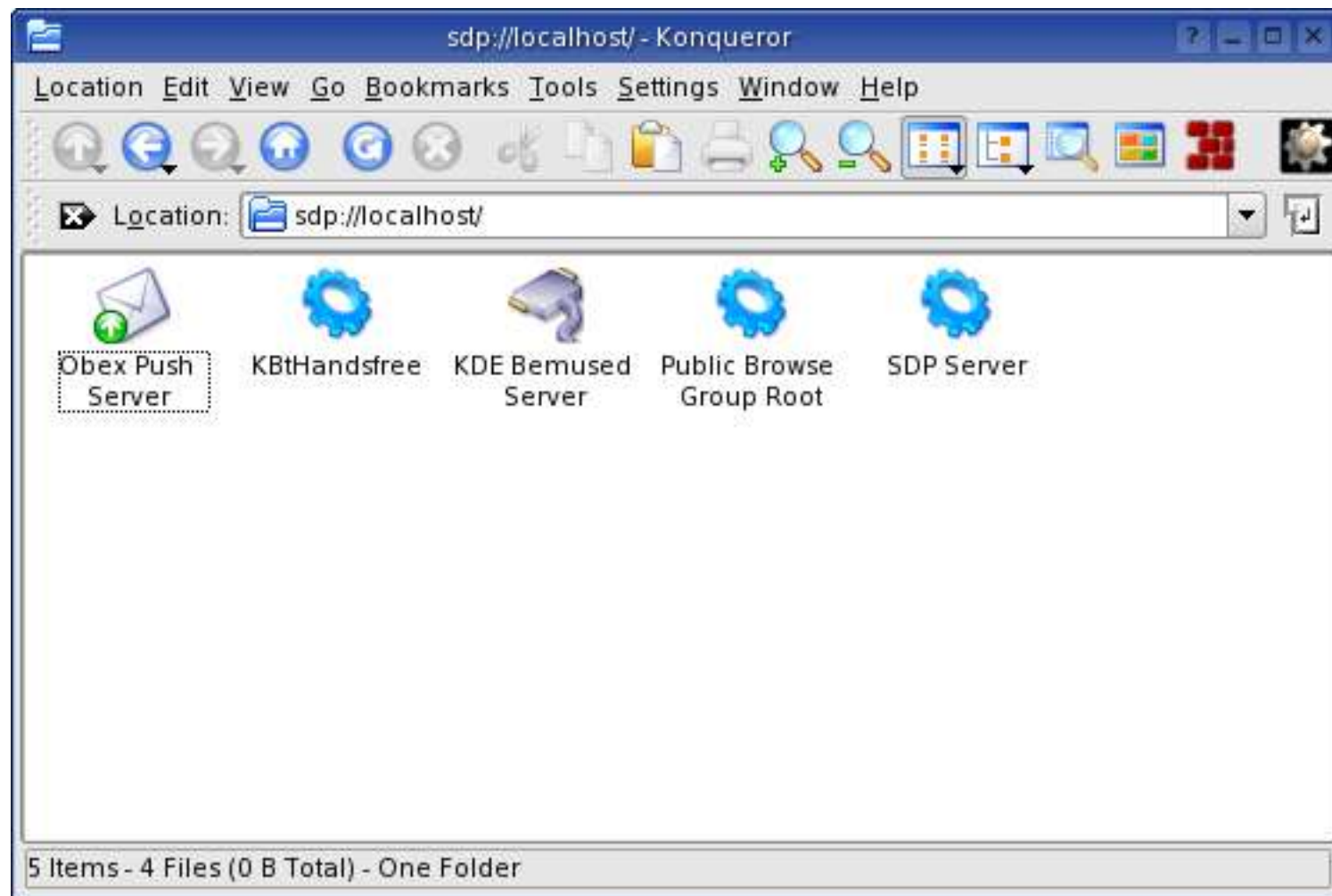


KDE Bluetooth: discovery /1



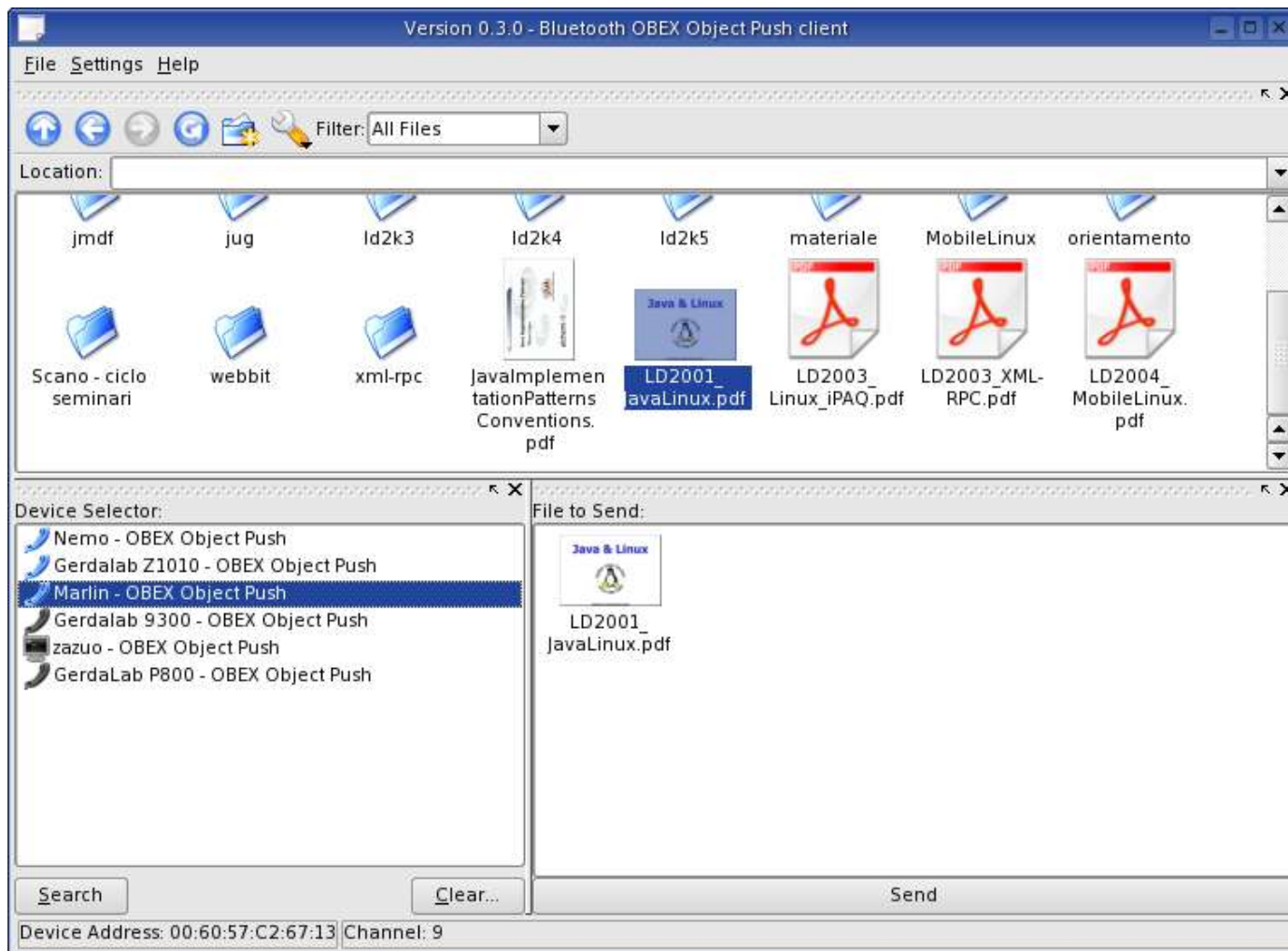


KDE Bluetooth: discovery /2





KDE Bluetooth: OBEX Push



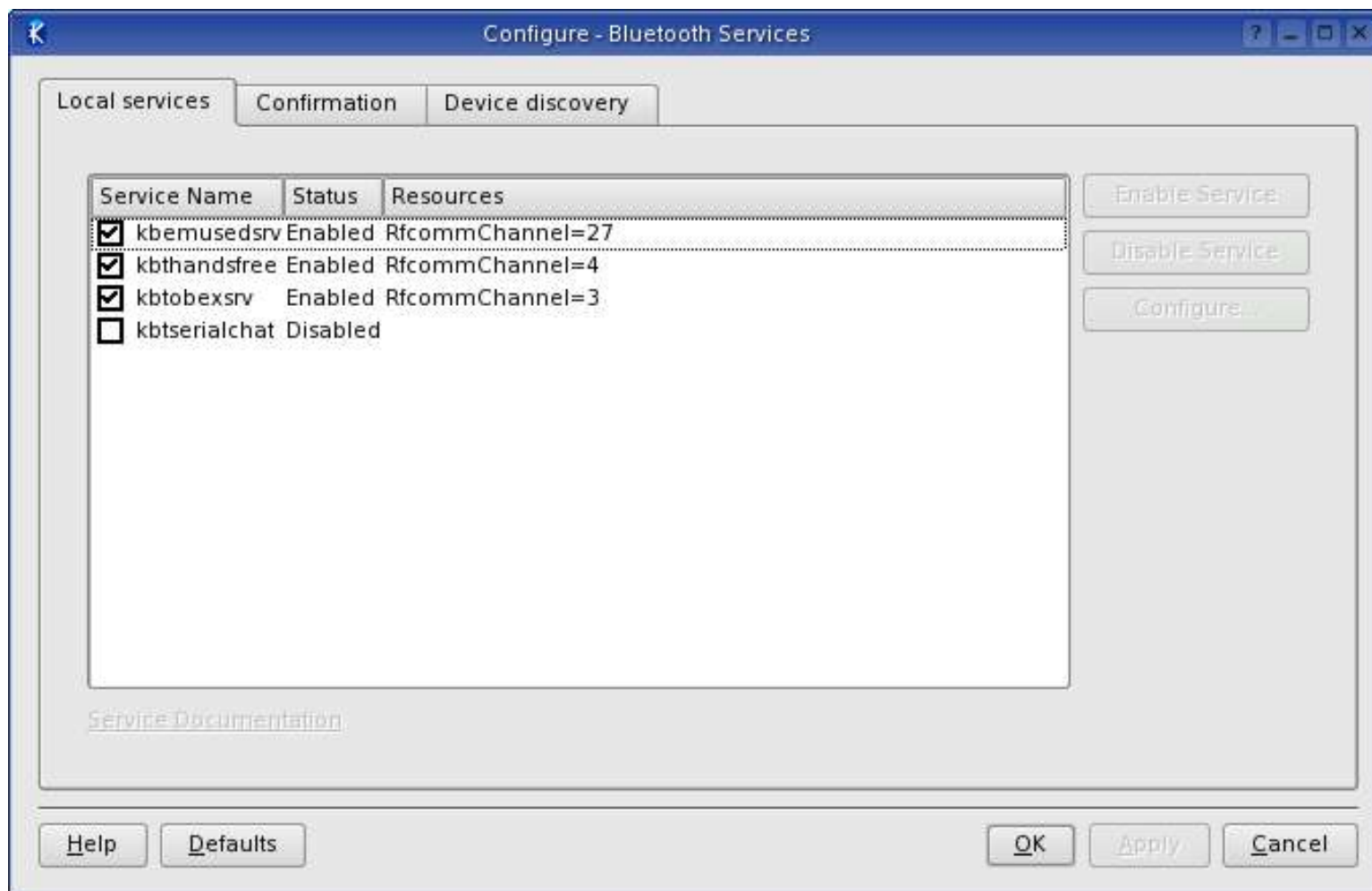


Un pendrive... wireless! :-)

- I pendrive USB hanno ridotte dimensioni, alte capacità, compatibilità con qualsiasi sistema operativo ma... anche uno svantaggio: per poter scambiare i dati tra due utenti è necessario disporre di un computer o di un datatransfer
- Disponendo di supporto Bluetooth sul PC e sul cellulare, possiamo usare il telefono come pendrive ridotto, ma con grandi vantaggi:
 - Trasferimento dati P2P senza adattatori
 - Invio e ricezione via MMS
 - ... evitiamo di portarci appresso telefono + pendrive!

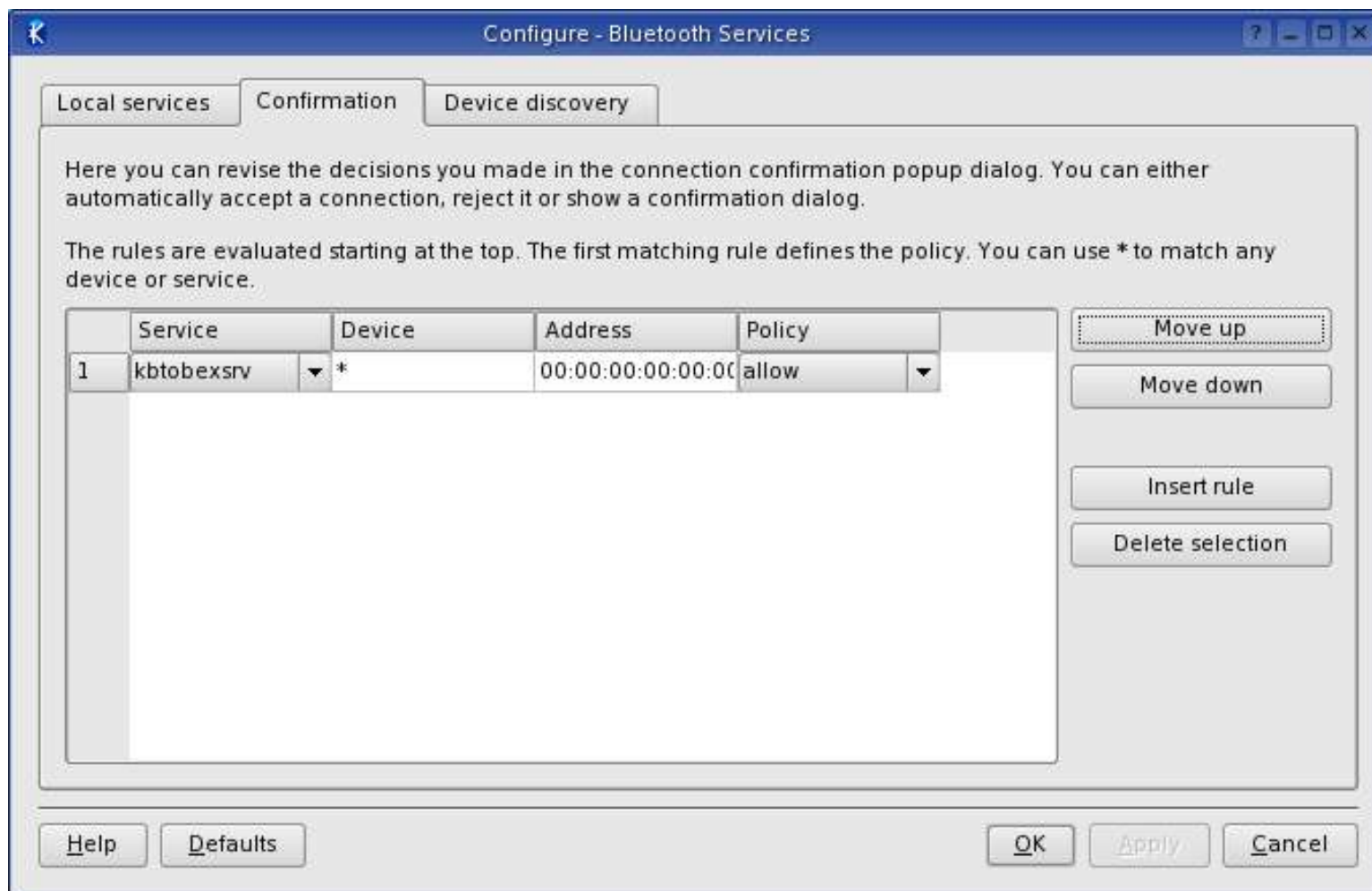


KDE Bluetooth: Servizi locali



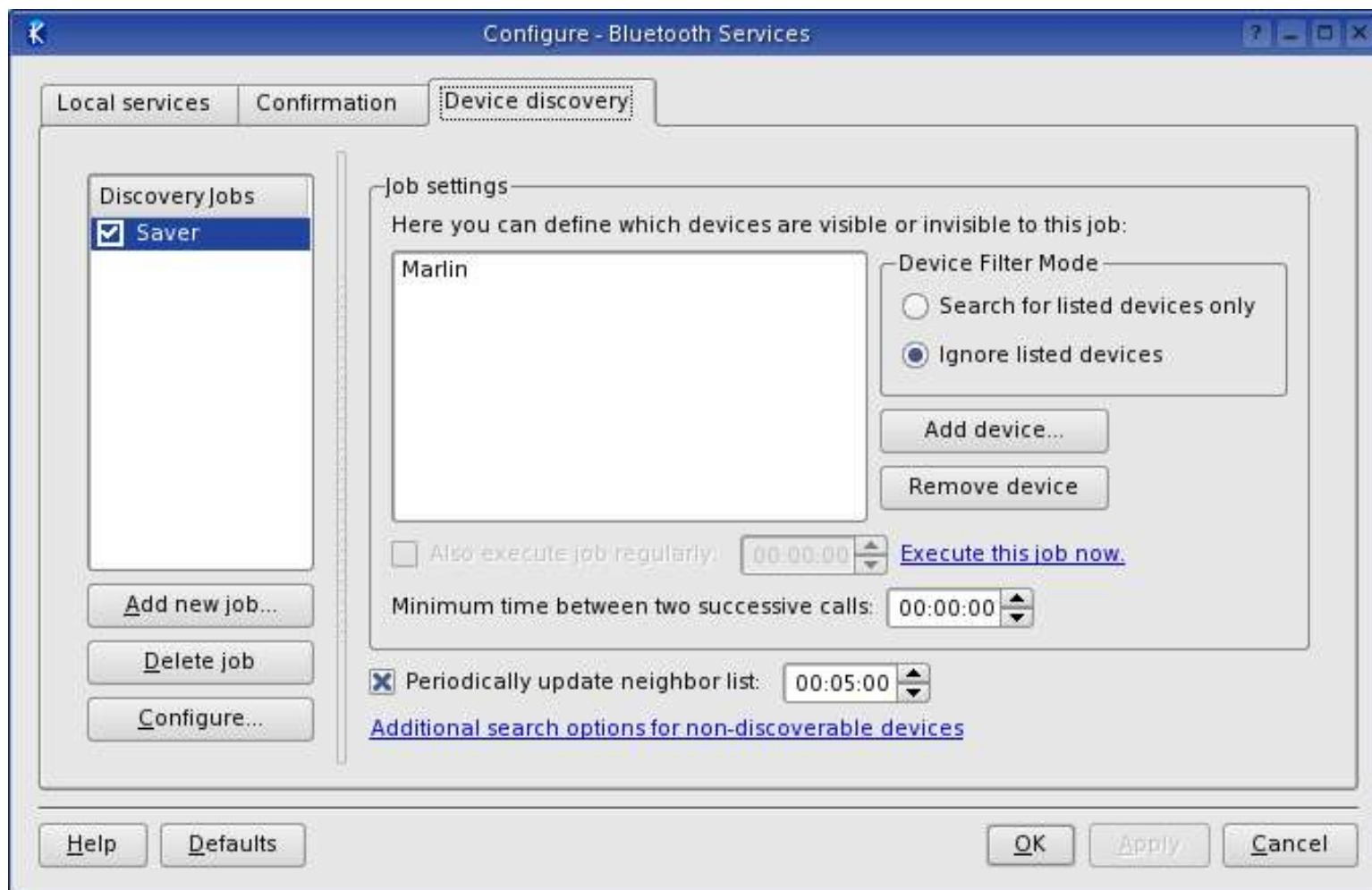


KDE Bluetooth: autorizzazioni





KDE Bluetooth: discovery





Connessione modem Bluetooth

Browsing **AA:BB:CC:DD:EE:FF** ...

Service ReHandle: 0x0

Service Class ID List:

"SDP Server" (0x1000)

Protocol Descriptor List:

"L2CAP" (0x0100)

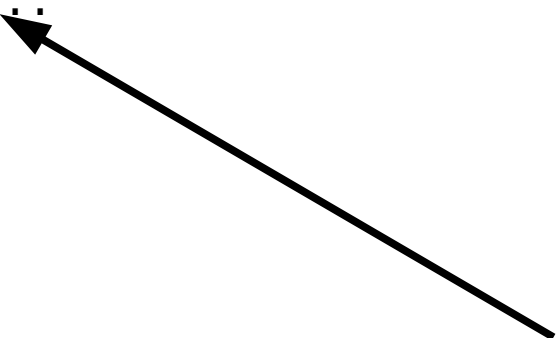
"SDP" (0x0001)

Profile Descriptor List:

"" (0x1000)

Version: 0x0100

BD_ADDR



... il dispositivo si presenta...



... e descrive i servizi offerti:

Nome del servizio

Service Name: **Dial-up networking Gateway**

Service Description: Dial-up networking Gateway

Service Provider: Motorola

Service RecHandle: 0x10001

Service Class ID List:

"**Dialup Networking**" (0x1103)

Protocol Descriptor List:

"L2CAP" (0x0100)

"**RFCOMM**" (0x0003)

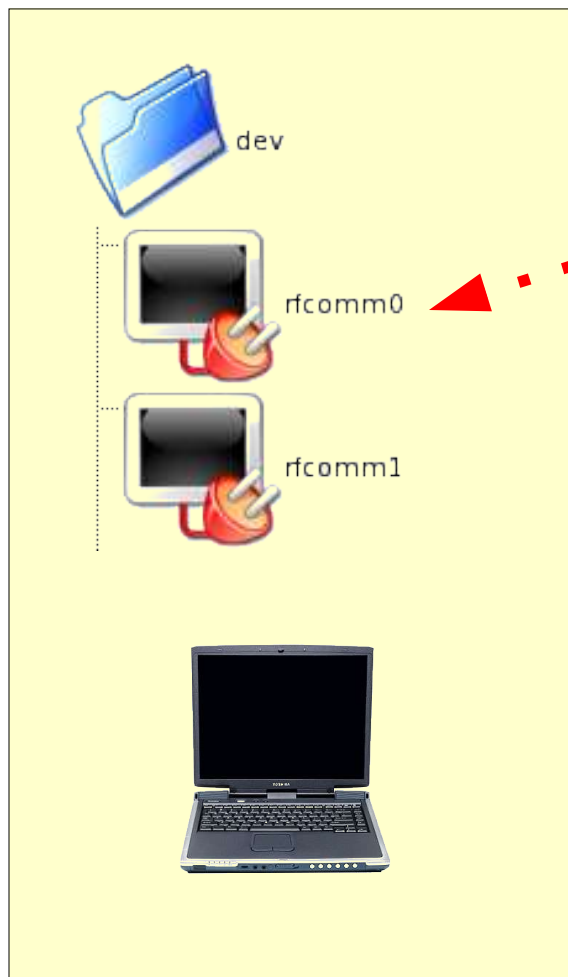
Channel: 1

Si tratta pur
sempre di una
connessione
seriale... :-)

Canale (porta) del
servizio Dialup



Assegnazione porta dial-up



Dialup	1
Hands-free	7
OBEX Push	8
OBEX FTP	9
Headset	3



00:11:22:AA:BB:CC



Assegnazione porta dial-up

- Una volta identificato il servizio di interesse, possiamo assegnare la porta locale `rfcomm0` al servizio 1 del dispositivo in uso:

```
rfcomm bind 0 00:11:22:AA:BB:CC 1
```

- `/dev/rfcomm0` diventa l'handle locale per la seriale esportata dal telefonino



rfcomm.conf

```
rfcomm0 {
    # Automatically bind the device at startup
    bind no;

    # Bluetooth address of the device
    device 11:22:33:44:55:66;

    # RFCOMM channel for the connection
    channel 1;

    # Description of the connection
    comment "Example Bluetooth device";
}
```



Configurazione pppd...

- ... la trovate sulle slide del Linux Day dello scorso anno o nel seminario extended “Mobile Linux” di aprile!



- Materiale:
 - Seminario introduttivo:
<http://linuxday.gulch.it/2004/>
 - Seminario esteso:
<http://seminari.gulch.crs4.it/slides/mobilelinux/>





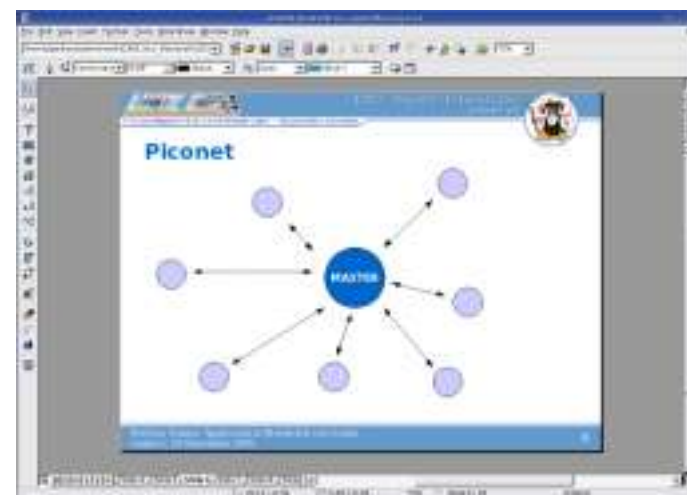
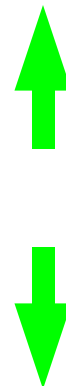
Alcune note...

- Non sempre un dispositivo pubblica un servizio con il medesimo ID (vedi iSync su 6630...)
- Talvolta lo stack Bluetooth di PC e dispositivi entra in uno stato inconsistente... e non funziona più niente: piuttosto che insistere un sano riavvio dello stack risolve i problemi e fa risparmiare tempo! Bluetooth 2.0 dovrebbe risolvere questi e altri problemi! :-)
- Ogni tanto... nella mia stanza Linux e Mac OS X litigano: il discovery fallisce se entrambi hanno il Bluetooth attivo! :-)



Blix

- Blix è un framework per il controllo remoto del desktop attraverso Bluetooth.
- Attualmente implementato con Java ME + JSR 82 su cellulare e Avetana Bluetooth API for Linux
- Sarà presentato a Java Polis...



BLIX

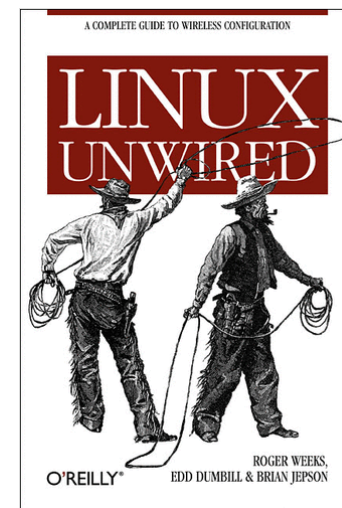


BLIX server



Bibliografia

- **Bluetooth and Linux**
 - <http://www.holtmann.org/linux/bluetooth/>
- Bluetooth:
 - <http://www.bluetooth.org>
- **Linux Unwired**
 - O'Reilly: <http://www.oreilly.com/catalog/lnxunwired/>
- Linux & Bluetooth
 - DEV 128-130-131, Gruppo Editoriale Infomedia
- Motorola A835 e Linux Mini-HOWTO
 - <http://www.gerdavax.it/linux/umts>





Software

- Bluez:
 - <http://www.bluez.org>
- OpenObex
 - <http://openobex.sourceforge.net>
- KDE Bluetooth Framework
 - <http://kde-bluetooth.sourceforge.net/>



Grazie per l'attenzione.



Licenza

Applicazioni Bluetooth con Linux

(Versione 1.0)

(C) 2005 Stefano Sanna (gerdavax@tiscali.it)

è garantito il permesso di copiare, distribuire e/o modificare questo documento seguendo i termini della Licenza per Documentazione Libera GNU, Versione 1.1 o ogni versione successiva pubblicata dalla Free Software Foundation. Una copia della licenza in lingua italiana è disponibile presso: <http://www.softwarelibero.it/gnudoc/fdl.it.html>

Realizzato in ambiente Linux con OpenOffice 1.1.2

Tutti i marchi commerciali sono di proprietà dei rispettivi titolari e sono stati citati in questa presentazione a solo scopo illustrativo.