



Bluetooth™ Profiles Overview

- Bluetooth Wireless Technology made simple, efficient and reliable -



Document history

Revision	Date	Description
1.00	May 6 th 2001	Creation
1.01	June 13 th 2001	More profiles & new document template
1.02	June 18 th 2001	Corrections
1.03	Dec. 18 th 2001	Minor revisions for public release
1.04	Jan. 4 th 2001	Profile list update

Legal notices

The ST logo is a registered trademark of STMicroelectronics. *BLUETOOTH* is a trademark owned by Telefonaktiebolaget L M Ericsson, Sweden and licensed to STMicroelectronics. All other brands and names are property of their respective owners.

Information furnished is believed to be accurate and reliable. However, STMicroelectronics assumes no responsibility for the consequences of the use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of STMicroelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. STMicroelectronics products are not authorized for use as critical components in life support devices or systems without express written approval of STMicroelectronics.

© Copyright 2002 STMicroelectronics.

For selected STMicroelectronics sales offices fax: France +33 1 47407910; Germany +49 89 4605454; Italy +39 02 8250449; Japan +81 3 57838216; Singapore +65 4820240; Sweden +46 8 7504950; Switzerland +41 22 9292900; United Kingdom and Eire +44 1628 890391; USA +1 781 861 2678

Index

1 - Introduction	3
2 - Services	5
3 - Generic Access profile.....	6
4 - Service Discovery profile.....	7
5 - Serial Port profile.....	8
6 - Headset profile	8
7 - Generic Object Exchange profile.....	9
8 - Basic Printing profile	11
9 - Human Interface Device	13
10 - Synchronization.....	15
11 - Other profiles	16
12 - Profile implementation steps.....	17
13 - Profiles List	18
14 - Reference documents.....	19

1 - Introduction

1.1 About this document

This document explains the concept of profiles and its role in a Bluetooth application.

Understanding profiles may help designers and programmers to adjust and customize their application to the interoperability requirements. This document is dedicated to technical people or marketing people who do not need a high level of details.

The reader should be familiar with Bluetooth concepts, described in BluePass Bluetooth Tutorial technical note for instance.

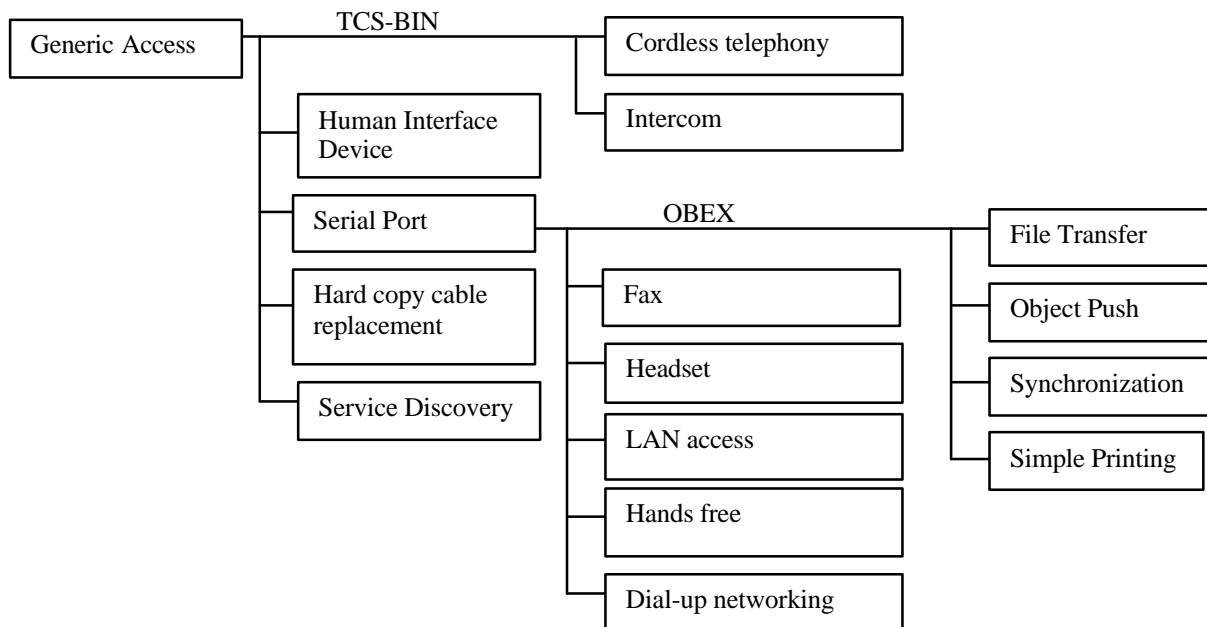
1.2 About profiles

The goal of defining profiles is to allow two devices from different manufacturers to work together, such as a PDA and a printer. A profile specifies a list of mandatory and optional features that a device should implement. It also specifies the communication procedures to guarantee interoperability between different kinds of products.

In our case, we will consider profiles as a set of high level procedures using lower layers of Bluetooth. At the user level, a profile is device family, regarding its functions. For a developer or a designer, profiles organize basic components into interoperable groups. Regarding Bluetooth architecture, a profile is a device description from the application point of view.

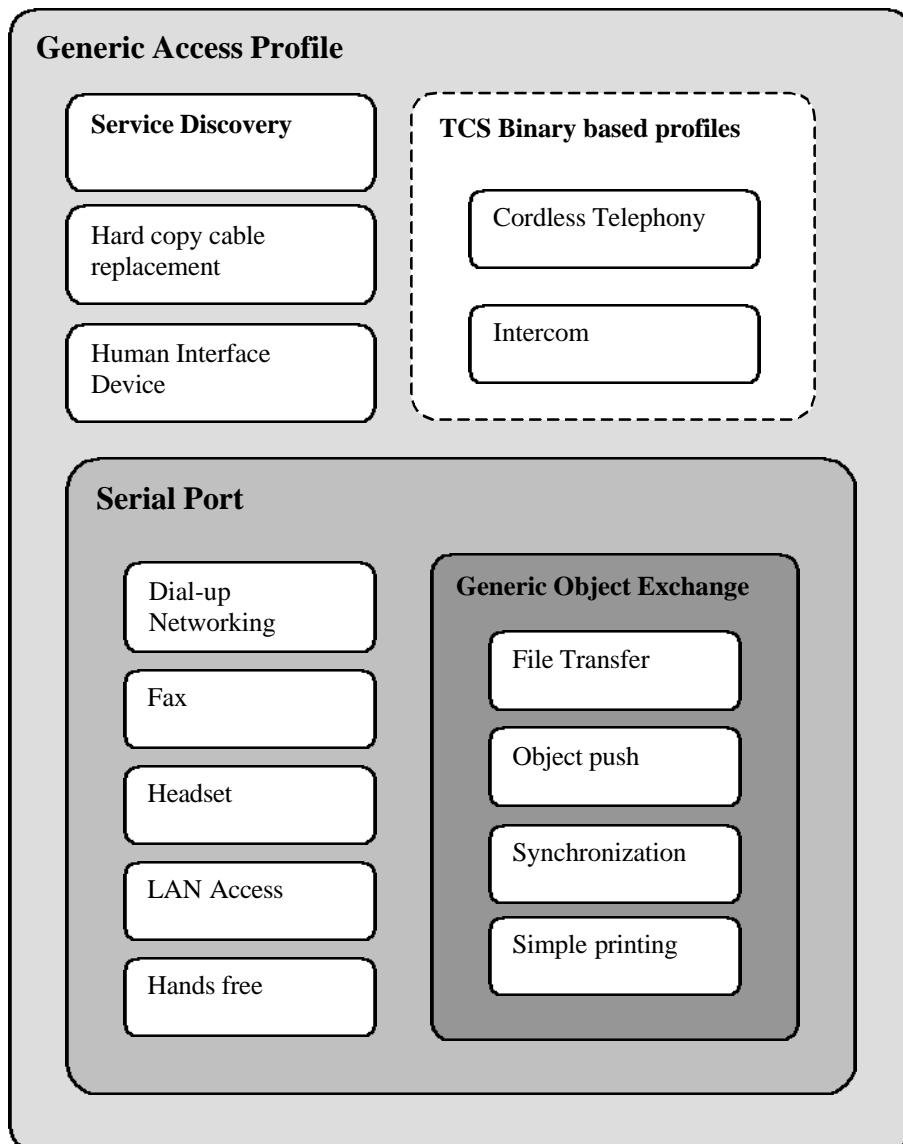
1.3 Hierarchy

Because some features are common to many different applications, simple profiles have been defined so that they can be reused, or inherited, by more complex profiles. These dependencies give us a tree organization as follows:



This diagram shows for example that a device implementing file transfer profile requires OBEX services, serial port profile and GAP.

The SIG documentation contains a slightly different presentation of profiles:



2 - Services

2.1 Definition

A service is a function that a device can execute. It can be a simple exchange or a set of low level exchanges. “Send Data” is a service, “Set Voice Connection” is also a service. Some services are specific to a device or a profile, such as “Get Margins” for a printer device, some services are common to most devices and profiles, such as “Get Device Name”.

A service is identified by an UUID (Universal Unique Identifier), a 128 bit number created by the SIG or by manufacturers, according to IEC and RPC (remote procedure calls) standards.

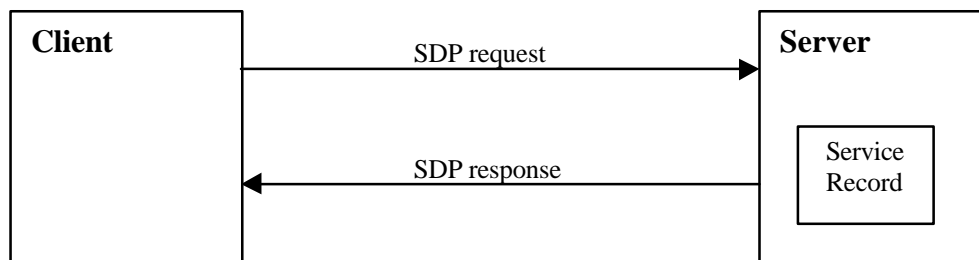
2.2 Service discovery

Service Discovery is a plug and play protocol over the air: it allows discovery of all Bluetooth devices in range and lists their available service’s characteristics. The set of available services is usually dynamically updated regarding to RF proximity of the devices.

2.3 Principle

SDP protocol involves communication between an SDP server and an SDP client. A SDP server maintain a database of service records. Each service record provides information that a client needs to access the service. This information may include URLs for executables, documentation and icons associated with the service.

Services attributes are exchanged through L2CAP exchanges. The format of the attribute list is specific to each service and can be very complex.



3 - Generic Access profile

Generic access profile (GAP) is inherited by all other profiles, as it is the minimum set of services needed to be Bluetooth enabled.

3.1 Modes

A Bluetooth device can be in different states, GAP makes some of them mandatory

Service name	Required	BluePass	Description
Non-discoverable mode	O	Y	Does not respond to inquiries
Limited discoverable mode	O	Y	Periodically responds to inquiries, not permanently
General discoverable mode	O	Y	Permanently discoverable
Non connectable mode	O	Y	Refuses connection
Connectable mode	M	Y	Can be connected
Non pairable mode	O	Y	Cannot be paired for authenticated connection
Pairable mode	O	Y	Can be paired for authenticated connection

M means mandatory, O means optional, Y means that the feature is supported in BluePass.
To keep things simple, some services are marked optional although SIG document gives more conditions.

3.2 Security aspects

Security mode 1 means 'no security' : a third Bluetooth device can spy and understand all exchanges.
Security mode 2 means that only L2CAP exchanges requires authentication. (no security at LMP level)
Security mode 3 means that a device needs to be paired before connection, authentication is required..

Service name	Required	BluePass	Description
Authentication	O	Y	Bonding is possible to get a secret key
Security mode 1	O	Y	Connection without key is possible
Security mode 2	O	Y	Some services requires a key
Security mode 3	O	Y	Connection requires secret key

Of course, when Security mode 2 or mode 3 are implemented, authentication is mandatory.

3.3 Idle modes

Idle mode procedures are discovery services.

Service name	Required	BluePass	Description
General inquiry	O	Y	Able to discover any kind of device
Limited inquiry	O	Y	Able to discover limited discoverable devices
Name discovery	O	Y	Get name request
Device discovery	O	Y	Able to discover and get name of devices
Bonding	O	Y	Enable link between two paired devices

3.4 Establishment procedures

Service name	Server	Client	BluePass	Description
Link establishment	M	M	Y	Able to establish a LMP link
Channel establishment	M	O	Y	Able to establish a L2CAP link
Connection establishment	O	O	Y	Able to establish a RFCOMM, TCS,... link

4 - Service Discovery profile

4.1 Definition

Service discovery differs from other service as it does not execute physical actions: it is a mean of informing devices before accessing other services.

Most of devices will implement service discovery, as it is a one of the fundamental concept of Bluetooth: being able to make a dynamic connection with anyone.

There are of 3 types of requests:

- what services are available (browsing)
- what services are available in this class of service ?
- what are the attributes of this service ?

Service name	Server	Client	Description
SDP Client	M	O	Able to discover devices
SDP Server	O	M	Able to be discovered

4.2 SDP requests

Service discovery is a set of requests/answers initiate by a client to a server.

Some devices will always be servers (a printer is usually a server), some will always be client, some devices may be both.

Service name	Send from client Receive on server	Receive on client Send from server	BluePass
SDP_ErrorResponse	O	M	Y
SDP_ServiceSearchRequest	M	O	Y
SDP_ServiceSearchResponse	O	M	Y
SDP_ServiceAttributeRequest	M	O	Y
SDP_ServiceAttributeResponse	O	M	Y
SDP_ServiceSearchAttributeRequest	M	O	Y
SDP_ServiceSerarchAttributeResponse	O	M	Y

5 - Serial Port profile

5.1 Definition

The serial port profile provides a way of using legacy (existing) applications designed for RS232 serial connections with Bluetooth devices.

5.2 Link establishment

Here are the services available in this profile:

(The connection initiator is called client, the other device is called server)

Service name	Client	Server	BluePass
Establish link and set up RFCOMM connection	M	O	Y
Accept link and establish RFCOMM connection	O	M	Y
Disconnect RFCOMM	M	M	Y
Register service record in local data base	O	M	Y
RS232 control signal management	M	M	Y
Transfer information	M	O	Y
Send remote line status indication	O	O	Y
Receive remote line status indication	O	M	Y
Send remote port negotiation	O	O	Y
Receive remote port negotiation	O	M	Y

Remote line status indication is a way of reading the serial line error status.

Port negotiation consists in accepting or refusing port settings.

6 - Headset profile

6.1 Definition

The headset profile may be used for headsets, personal computers and cellular phones, every time simple audio has to be transmitted.

We call Headset (HS) the device on the user side (usually integrating a microphone and a loudspeaker).

We call Audio Gateway (AG) the other device.

A common usage is to have a headset which can receive incoming phones calls, and possibly adjust headphones volume.

6.2 Principle

The audio transmission is monophonic with non perceivable audio degradation, but it is not a CD quality.
The AG controls the SCO establishment and release.
The HS initiates the connection.
The connection may be encrypted, using Gap services or may not.
An RFCOMM connection may be used to exchange modem AT commands.

6.3 Application layer

Here is the list of the headset features

Service name	HS	AG	BluePass
Incoming audio connection	M	M	Y
Outgoing audio connection	M	O	Y
Audio connection transfer	M	M	Y
Remote audio volume control	O	O	Y

In a headset device, the ACL link can be used to exchange AT commands in the following standards format:

Question: AT <cmd>=<value><cr>
Answer: <cr><lf>OK<cr><lf> or <cr><lf>ERROR<cr><lf>
Event: <cr><lf><result code><cr><lf>

The SIG defines a list of optional AT commands which can be used in this profile.

7 - Generic Object Exchange profile

7.1 Definition

OBEX (OBject EXchange) is a client server protocol, part of the IrDA (Infrared Data Association) standard. The SIG has adopted an existing standard for the high level object exchange layers, to ease interoperability and reduce technical risks.

OBEX can be used over a RFCOMM link: each server has it's own channel and has to be registered in the service data base. OBEX can also be used over TCP/IP.

7.2 Features

Service	Description
Establishing a session	Uses the OBEX connection protocol to initiate exchange
Pushing data object	Send data from client to server
Pulling data object	Send data from server to client

Connection can be performed with or without authentication. In case of authentication, the pin code is used as OBEX password.

7.3 OBEX headers

All OBEX packets contains a Header where the first byte specified the type of packet. It is called Header ID. Header ID low bits: Packet size code

Bit 7	Bit 8	Definition
0	0	String packet, 2 next bytes specifying length
0	1	Byte packet, 2 next bytes specifying length
1	0	Single byte packet
1	1	4 bytes packet

Here is the list of the 17 standard Header ID codes :

HID Code	Data size	Definition
0x01	2+N	object name as a text string
0x05	2+N	object description as a text string
0x42	1	type of object
0x44	2+N	Type of object, as a null terminated string
0x46	1	Target service name
0x47	4	HTTP version 1.x header
0x48	2+N	Body
0x49	2+N	end of body
0x4A	1	who: 128 bits UUID of the OBEX service destination
0x4C	2+N	application parameters
0x4D	2+N	Authentication challenge
0x4E	1	Authentication response
0x4F	4	OBEX object class
0xC0	2+N	number of objects
0xC3	4	object length
0xC4	2+N	4 bytes date and time stamp
0xCB	4	connection ID in case of multiplexed OBEX connections

Values are always sent high byte first. N is the variable length of string or the number of objects. The use of a header is not mandatory, when the 2 devices exactly know what they expect to receive

7.4 OBEX operations

There are only 6 types of operations in the OBEX session protocol:

OBEX command	Definition
0x80	Connect
0x81	Disconnect
0x02	Put object: send name header, length header, body and end of body packets
0x03	Get object: send name header, send get until 'continue' answer is replaced by 'success'
0x85	Set path of the receiving side
0xFF	Terminate operation before normal end

8 - Basic Printing profile

8.1 Definition

The printer is the GOEP (Generic Object Exchange Profile) server receiving an object to print.
The sender is the COEP client pushing an object to print.

This profiles defines the requirements for interoperability between phones, pagers , PDA... and printers for text document such as Emails, short documents...
Optional support for printing objects such as vCars and vCalendar, and other formats negotiation is also defined.

8.2 Requirements

Link level authentication and encryption are mandatory to support and optional to use: the printer must support authentication if the sender wants security.
OBEX authentication is mandatory for senders, optional for printers.

8.3 Printer modes

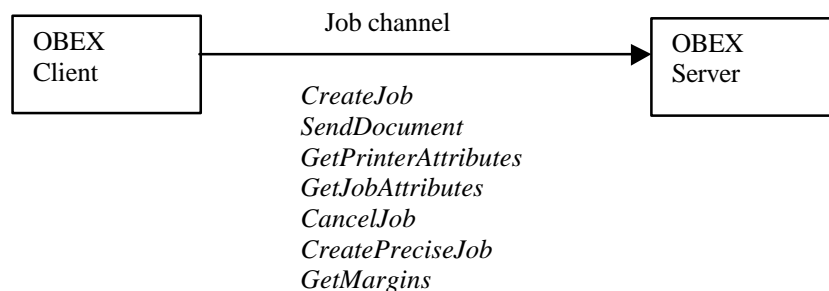
Printer mode	Required	Description
Off line mode	O	Printer cannot receive job, cannot be discovered
Bonding mode	M	Printer is ready to be bonded
Private on line mode	O	Only clients knowing printer address can send job Printer cannot be discovered
Public on line mode	M	Anyone can discover the printer and send job

8.4 Application layer

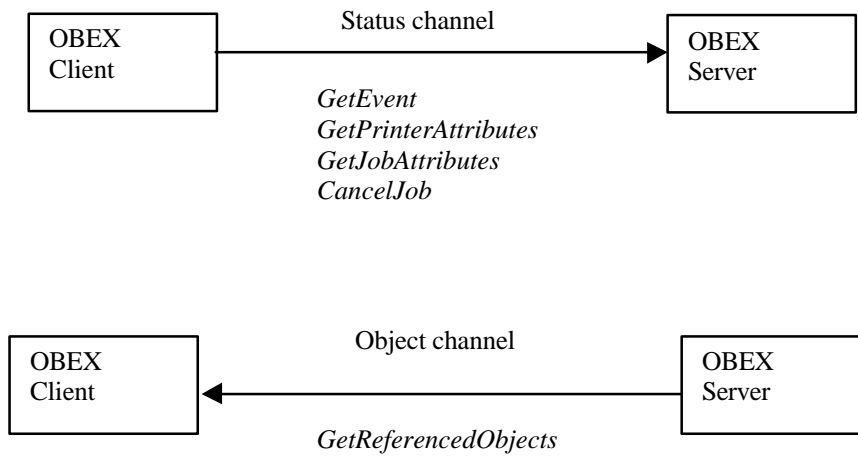
There are basically two type of application model:

- The simple file push model: the file is sent to the printer and no control is possible.
This implementation uses only the object push model
- The Internet Print Protocol 1.1 using the concept of print job described here after:

The client creates a job in the printer and get a job number in return.



Optional OBEX servers can be implemented to offer more features



The SIG profile document gives 20 pages of protocol descriptions to set margins, fonts, .. and many printer settings. These specification are described in XML format.

8.5 Objects format

The basic object formats supported by the profiles are the following:

Format	Sender	Printer	Description
XHTML-print	M	M	Compliant to XHTML standards
Basic text	O	O	Unicode character stream, default format is assumed to be A4
Vcard	O	O	Vcard is a MIME standard for mail servers
Vcalendar	O	O	Idem
Vmessage	O	O	Idem

Other optional object formats to be supported

Format	Description
Postscript 2	Adobe protocol for text & graphics
Postscript 3	Adobe protocol for text & graphics
PCL5E	Hewlett Packard for text & Graphics
PCL3E	Hewlett Packard for text & Graphics
PDF	Adobe Acrobat format for on line documentation
JPEG	29 compression formats for pictures
GIF89A	Unisys compression format for pictures

8.6 Media formats

Here is a list of the media types that may be supported

Value	Description
Plainpaper	standard copy, bond or recycled
Letterhead	Pre-printed letterhead stock
Transparency	Coated or uncoated transparent media
PhotoPaper	Glossy quality printing paper
InkJetPaper	
LabelStock	Label paper
CardStock	Thick, stiff media
Textured paper	
Envelope	Multi layer envelope
EnvelopeWin	Envelope with windows
CoatedPaper	
Other	
Unspecified	Unknown, default...

As the protocol uses XML language, all values here defined as text string in long lists not mentioned here. Please refer to SIG document for more details.

9 - Human Interface Device

9.1 Definition

Typical HID devices are keyboards, mice, game controllers, sensors, simple alphanumerical displays, etc... usually connected to a computer using a cable or an infrared link. The computer is called host. The Bluetooth HID profile uses the USB HID specifications, and targets a wide range of devices and features.

9.2 Requirements

9.2.1 First use configuration

Most of HID devices will be used by non professional users, such as mice and keyboards. Thus the installation and the setting must be simple: the host computer must be able to detect the device, and to show to user the PIN code to enter in the device, to get them paired.

9.2.2 Power management

Most of HID devices will be powered by batteries. The expected life duration is 3 months. Thus the device has to deal with HOLD, PARK and SNIFF mode.

Active mode	Data is polled at about 100 times per second
Standby mode	Sniff mode is entered after a configurable duration of inactivity. It takes 50ms after wake up before being able to transmit.
Sleep mode	Sleep mode is entered after a configurable duration of standby mode. The device just keep synchronized with host radio
Suspend state	Suspend state is entered when the connection is lost in sleep mode. The device periodically listen for a host in page scan mode. It takes 2.5 seconds to reconnect

9.3 Protocol

The profile defines 3 types of exchanges:

Type	Description
Feature reports	Standard speed exchanges, from device to host, or from host to device
Input reports	Low latency exchanges (interrupts) from device to host
Output reports	Low latency exchanges (interrupts) from device to host

All exchange data are preceded by a header from the following list

Hex code	Transaction type	Payload length
0	Handshake	1
1	Hid control	1
2-3	Reserved	
4	Get report	1 to 4
5	Set report	1 + report data
6	Get protocol	1
7	Set protocol	1
8	Get idle	1
9	Set idle	2
A	Data	1 + report data
B	Datc	1 + continuation data
C-F	Reserved	

10 - Synchronization

10.1 Definition

Synchronizing devices consist in comparing objects presents on two devices, and to make objects transfers in order to have the same image on both devices. Update rules of course are configurable and the user can control each operation if necessary.

Typical example of devices to be synchronized are computers, PDA and mobile phones.

IrMC Server: device providing an object exchange server

IrMC Client: device providing the synchronization engines, which pushes and pulls the objects to the server. Usually the client is a PC.

Compared to most other profiles, synchronization usually does not require any automatic discovery and connection: the user has to set up the connection manually on both sides.

10.2 Features

Features	IrMC Client	IrMC Server
Synchronization of phonebooks	O	O
Synchronization of calendars	O	O
Synchronization of emails	O	O
Synchronization of notes	O	O
Synchronization command	M	O
Automatic synchronization	O	M

Synchronization command is a temporary exchange of client/server roles.

Automatic synchronization is the mode where the client automatically connects the server coming in range.

11 - Other profiles

11.1 Cordless Telephony and Intercom profiles

These two profiles are dedicated to mobile phone application. They require the TCS-BIN module (telephony control protocol specifications).

Cordless telephony enable a base connected to the telephone network to manage and respond to wireless devices for making outgoing call, receiving incoming calls, etc...

Intercom is a point to point connection using TCS-BIN, for instance 2 cellular phones.

11.2 Dial up Networking and Fax profiles

Dial up networking enable a wireless device to access a public telephone network, using AT commands.

Fax profile does something equivalent...

11.3 LAN access profile

The **LAN** profiles describes a PPP (point to point internet protocol) implemented over RFCOMM. It provides a wireless access to a LAN , as a normal LAN access point.

11.4 Hands Free profile

Hands free enables a car driver to control his mobile phone from a purely audio interface, except as optional features, some simple buttons.

A hand free device has to manage an ACL channel for AT commands and a voice channel for voice transport. The hand free device also has to connect automatically the mobile when it enters the car...

11.5 File Transfer

This profile is defined to access the directory and file structure of a remote device, for read and write access. A typical application of the File transfer is a wireless browser.

11.6 Object push

Object push profile is a simple application of OBEX which enable a **push client** to send data to a **push server** 'inbox'. There are 3 functions: object push, business card pull, business card push.

A business card object is commonly used in mail systems such as Microsoft Outlook™.

12 - Profile implementation steps

12.1 BluePass™ delivery

The BluePass package contains sample applications for Generic Access profile, Service Discovery profile, Serial Port profile and Headset profile. Please refer to “BluePass PC software documentation” for detailed information.

Programmers should refer to existing application to see how a prototype application can be structured.

12.2 Bluetooth specifications

Implementing a profiles basically means creating an application which complies to the profile definition in the document “ Specifications of Bluetooth system Volume 2: Profiles “

Thus, it is important to understand perfectly what mandatory services are required in the profile. It is also necessary to understand other profiles, to make sure that the profile really fits the targeted application.

12.3 Profile architecture

Most of profiles are built upon GAP, SDP and serial port profiles, but the implementation of these lower profile may change regarding to the top profile using them. Features such as encryption or ability to be discovered which are optional in the generic profile can become mandatory when the top profile requires it. This means that the minimum implementation of mandatory services is not sufficient for most of profile reuse, and the SIG documents should always be checked.

Building an application that uses existing profiles means that an existing application will be completed by new services and state machines. In simples cases, the new services will simply be added in the service data base, and the event management will provide handling for the new exchanges. In some cases, the Generic Access profile, and especially the security management will have to be modified to comply to the new requirements. The GAP implementation used for a Headset may not be the same than a GAP for a Hard copy cable replacement...

12.4 Functional tests

Before doing any compatibility test, programmers should verify that the functions that have been implemented simply works. Although the stack provide all tools and components, the combination of services and attributes may cause unexpected effects. It is important to verify every test case in a familiar and proprietary environment before trying a connection to a third party device.

12.5 Compatibility tests

The easiest way to improve the implementation of a profile would be to operate the device with third party reference devices such as mobile phones, headsets, etc.. This certainly gives a good understanding on how to implement the standard, and what features are really mandatory in terms of products interoperability.

12.6 Certification

The reader should refer to BluePass document “Bluetooth qualification process” for product qualification.

13 - Profiles List

13.1 Specifications Version 1.1

The Bluetooth Profile Specifications Version 1.1 is available online at:

http://www.bluetooth.com/pdf/Bluetooth_11_Profiles_Book.pdf

Component name	Owner	Index
Generic Access Profile	Patrik Lind	Part K:1
Service Delivery Profile - Application Profile	Chatschik Bisdikian	Part K:2
Cordless Telephone Profile	Erik Slotboom	Part K:3
Intercom Profile	Erik Slotboom	Part K:4
Serial Port Profile	Johan Sörensson	Part K:5
Headset Profile	Erik Slotboom	Part K:6
Dial-up Networking Profile	Erik Slotboom	Part K:7
Fax Profile	Erik Slotboom	Part K:8
LAN Access Profile	Dean Gratton	Part K:9
Generic Object Exchange Profile	Patrik Olsson	Part K:10
Object Push Profile	Patrik Olsson	Part K:11
File Transfer Profile	Shaun Astarabadi	Part K:12
Synchronization Profile	Stephane Bouet	Part K:13

13.2 Draft Specifications

The draft specifications are only available to registered members.

Profile name	Owner	Date	Version
Audio / Video	-		V0.95 Draft
Car - Hands Free	Jesus A.G. Pulido	22 nd Oct. 2001	V0.96
Human Interface Device	Craig Ranta/Steve McGowan	8 th Dec. 2001	V0.95 Draft
Common ISDN Access	Steve Trevisany	18 th Dec. 2001	V0.90b3
Basic Imaging	Imaging Working Group	23 rd Nov. 2001	V0.95c
Personal Area Network	Kris Fleming	26 th Jun. 2001	V0.95a
Network Encapsulation Protocol	Kris Fleming	12 th Jun. 2001	V0.95a
Basic Printing	-	5 th Oct. 2001	V0.95a
Hardcopy Cable Replacement	-	11 th Oct. 2001	V0.95a

The Audio/Video profile contains many sub sections that are not shown in this table.

14 - Reference documents

Name	Publisher, Author	Version
Bluetooth tutorial	STMicroelectronics	Feb 2001
Connection without cables	Prentice Hall, J.Bray	2001
Specifications of Bluetooth system Volume 2: Profiles	SIG	V1.1 (Feb 22 nd 2001)
Bluetooth demystified	Mc Graw Hill, J. Muller	2001
Basic printing profile draft	SIG	0.70 (Feb 25 th 2001)
Bluetooth Qualification Process	STMicroelectronics	V1.3 (Jan 4 th 2001)