Transmission of IP Packets over Bluetooth Networks

STATUS OF THIS MEMO

This document is an Internet-Draft and is in full conformance with all
provisions of Section 10 of RFC2026.

Internet-Drafts are working documents of the Internet Engineering Task
Force (IETF), its areas, and its working groups. Note that other
groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months
and may be updated, replaced, or obsoleted by other documents at any
time. It is inappropriate to use Internet-Drafts as reference material
or to cite them other than as "work in progress."

The list of current Internet-Drafts can be accessed at
http://www.ietf.org/ietf/1id-abstracts.txt

The list of Internet-Draft Shadow Directories can be accessed at
http://www.ietf.org/shadow.html.

Abstract

Bluetooth is a wireless standard for low cost, low power, local radio
communications. This document describes how to use the Specification of
the Bluetooth System Core Version 1.1 standard for the transport of
Internet Protocol Version 4 and Internet Protocol Version 6 datagrams.
This document describes link MTU, path MTU, Address Resolution, IPv6
Interface ID, encapsulation, and how to route unicast, multicast,
anycast, and broadcast IP datagrams over the Bluetooth link-layer.

1. INTRODUCTION

Bluetooth is a wireless standard for low cost, low power, local radio
communications. The Specification of the Bluetooth System Core Version
1.1 [INTRO:1] is standardized in the Bluetooth Special Interest Group,
Inc. This document describes an IP over Bluetooth solution, and
discusses how it differs from current approaches.

                            TABLE OF CONTENTS
                            ----

2. **Motivation**

The current solution places IP over the Bluetooth Network
Encapsulation Protocol (BNEP). BNEP runs over the Bluetooth Logical
Link Control and Adaptation Protocol (L2CAP). While BNEP is a method
for IP Encapsulation, it has drawbacks, limitations, and is a
duplication of the functionality provided directly by IP.

BNEP Drawbacks:

o Adds an additional encapsulation layer between the L2CAP
   encapsulation layer, and the encapsulation provided by IP, such
   that there are three encapsulation layers for an IP Bluetooth
   solution: L2CAP, BNEP, and IP.

o BNEP/L2CAP is not optimized for the transmission of IP, the most
   widely used network protocol.

o BNEP is implicitly based on IETF "IP over IEEE 1394", which unlike
   Bluetooth has no encapsulation layer.

o Treats IP as less important than other protocols.

Limitations:

o Not interoperable across the Internet, or any other type of
   network.

o Will require BNEP headers to be removed and added for other types
   of link-layers like USB, I2C, CANBus, IEEE 1394, and 3G Cellular.

Duplication of Effort:

o IP already supports encapsulation of the protocols currently
   specified by BNEP, or envisioned such as IPX, AppleTalk, and
   Ethernet.

o While BNEP is easy to invent, it will require vendors to support
   and maintain two protocols that provide the same functionality:
   BNEP and IP.

o The support and maintenance costs for vendors will increase over
   time as more functionality is added to BNEP, while the needed
   functionality is already available in IP. At some point, vendors
   will have to weigh the costs of continuing to support two
   protocols with the same functionality.

The approach taken in this draft is not to support the creation of a
"Bluetooth Only" networking solution, that considers Bluetooth devices
as only forming a Bluetooth network in isolation of other types of
networks. But, to consider a Bluetooth device as a part of a larger
heterogeneous IP network, where each link-layer is responsible for
interacting with IP according to its own unique characteristics
[HOST:1]. The larger heterogeneous IP network may be composed of any
combination of link layers, such as, but not restricted to: USB,
IEEE 1394, Ethernet, I2C, CANBus, DeviceNet, Bluetooth, and
3G Cellular.

In this document Bluetooth will be treated as another link layer over
which all IP compatible protocols will be carried in an optimal
manner [HOST:2]. This approach has the advantage of allowing a
collection of Bluetooth devices the ability to communicate and
discover each other over the Internet, or any wired/wireless IP
capable network. Also, this approach allows an ad hoc network to be
treated as an isolated IP capable network, rather than as a special
case requiring special protocols.

This approach shows there to be two areas of interest in an IP
capable heterogeneous network:

1. IP: Network-layer specific
2. Bluetooth: Link-layer specific.

These two areas suggest the use of L2CAP for Bluetooth link-layer
specific encapsulation, and the use of IP for any network-layer
specific encapsulation. The limited functionality of BNEP is replaced
by the much greater functionality already provided by Protocol
Numbers [IANA:1], and UDP/TCP Port Numbers [IANA:2] in the Internet
Protocol.

2.1 Proposed Solution.

The proposed solution in this draft will have the following stack on
a Mobile Node attached to an IP capable network that may be an
isolated IP capable network, or a larger IP capable network:

```
+----------------------+
|   Internet Protocol  |
+----------------------+
|        L2CAP         |
+----------------------+
|         LMP          |
+----------------------+
```

The proposed solution will have the following stack on an IP Router
[ROUTER:1] attached to an IP capable wired-line network:

```
+----------------------------------+
|    Internet Protocol             |
+------+---+-----------------------+
|L2CAP |   |     Encapsulation     |
+------+   +-----------------------+ (I.E. Ethernet, CANBus, USB,..)
| LMP  |   | Wired-Line Link-Layer |---------------------------->
+------+   +-----------------------+   [To Wired-Line Network]
```

The proposed solution will have the following stack on an IP Router
attached to an IP capable wireless network:

```
+----------------------------------+
|    Internet Protocol             |
+------+---+-----------------------+
|L2CAP |   |     Encapsulation     |
+------+   +-----------------------+
| LMP  |   |   Wireless Link-Layer | (I.E. 802.11b, 2.5G, 3G,..)
+------+   +-----------------------+
```

3. L2CAP IP Encapsulation using a connectionless data channel

Unlike some other link-layers, such as IEEE 1394, Bluetooth has an
encapsulation protocol called L2CAP. L2CAP provides a Channel
Identifier (CID) field for the dynamic encapsulation of other protocols
after a Service Discovery procedure has been completed by SDP. Using
the CID field for the encapsulation for the Internet Protocol:

```
|                                  |
|  Length  | Channel ID |  Payload |
+----------+------------+----------+
|  16 bits | 16 bits    |          |
+----------+------------+----------+
```

3.1 IP Encapsulation:

```
|                                  |
|  Length  | Channel ID |   Payload |
+----------+------------+-----------+
|  16 bits |   CID_IP   | IP Packet |
+----------+------------+-----------+
```

The Service Discovery procedure can be further extended over an IP
capable network by merging SDP with the IETF Service Locator Protocol.
The SDP Record and SLP Service Attributes are to be defined.

[4](). Address Resolution

[4.1]() Link-Local Address Resolution (within a Piconet)

The Bluetooth link-layer is connection-oriented. Any two devices that
wish to communicate must setup a point-to-point connection. The
device initiating the connection assumes the role of "Master Device",
and the device being connected to assumes the role of "Slave Device".
The roles of "Master" and "Slave" may be reversed at any time. While
a single connection is point-to-point, like any other type of
Master-Slave network, the Master may have more than one point-to-point
connection active at any time. A device assuming the role of "Slave"
device may also be an active "Master" device on its own subnet
(A Piconet Bluetooth), or visa versa.

A connection must be setup before any IP packets can flow. There are
two ways that a connection can be initiated:

o   Inquiry followed by Page
o   Page

If the Bluetooth link-layer address of the other device is not known
previously, the Inquiry procedure must be preformed by the device
requesting a connection. If the link-layer address is known then the
requesting device can just perform the Page procedure. In any case,
after the connection setup process is complete both devices will know
the link-layer address of the other.

Therefore, a link-layer address resolution protocol [ARP:1] is not
necessary within a Piconet. The Master holds all link-layer addresses
within the Piconet as a result of connection setup.

[4.2]() Global Address Resolution (across an IP subnet)

An End-User wishing to discover the link-layer identity of device
attached at some distance away on the IP network will need an Address
Resolution Protocol [ARP:1].

Since the Bluetooth setup procedure is link-local, a method for
discovering the link-layer identity of a device is necessary across
an IP network. The End-user will need access to the same
link-layer information that is available within a single Piconet,
such as:

o   (CoD) Class of Device (24 bits)
o   (UAP) Upper Address Part (8 bits)
o   (NAP) Non-significant Address Part (16 bits)
o   (LAP) Lower Address Part (24 bits).

The ARP format [ARP:1],[IANA:3] can be used to convey link-layer
identity across an IP network:

```
    ar$hrd    16 bits          Hardware type (Bluetooth assigned by IANA)
    ar$pro    16 bits          Protocol type
    ar$hln     8 bits          Byte length of each hardware address (9)
    ar$pln     8 bits          Byte length of each protocol address (m)
    ar$op     16 bits          Operation code
    ar$sha     9 bytes         source hardware address (CoD,UAP,NAP,LAP)
    ar$spa     mbytes          source protocol address
    ar$tha     9 bytes         target hardware address
    ar$tpa     mbytes          target protocol address
```

Where the hardware address is nine bytes composed of:

o (CoD) Class of Device (24 bits)
o (UAP) Upper Address Part (8 bits)
o (NAP) Non-significant Address Part (16 bits)
o (LAP) Lower Address Part (24 bits).

The hardware address field packet format is:

```
|                                |
|    CoD   |  UAP   |   NAP   |   LAP   |
+----------+--------+---------+---------+
|  24 bits | 8 bits | 16 bits | 24 bits |
+----------+--------+---------+---------+
```

4.3 L2CAP ARP Encapsulation using a connectionless data channel

L2CAP provides a Protocol Service/ Multiplexer (PSM) field for
encapsulation of the Address Resolution Protocol. The PSM field can
can be used for the encapsulation of the Address Resolution
Protocol:

```
|                                                              |
|  Length  | Channel ID | Protocol Service/Multiplexer |  Payload  |
+----------+------------+------------------------------+-----------+
|  16 bits | 16 bits    |        16 bits               |           |
+----------+------------+------------------------------+-----------+
```

3.1 IP Encapsulation:

```
|                                                              |
|  Length  | Channel ID | Protocol Service/Multiplexer |   Payload  |
+----------+------------+------------------------------+-----------+
|  16 bits |   0x0002   |        PSM_ARP               | ARP Packet |
+----------+------------+------------------------------+-----------+
```
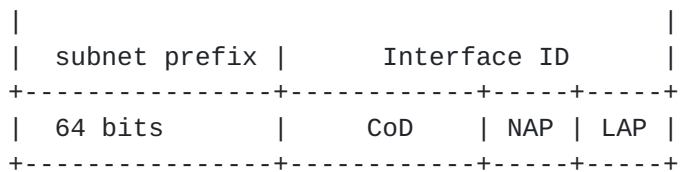
Also, a merged Service Discovery/Locator Protocol may convey link-layer indentity.

5. IPv6 Interface ID Mapping

In an IPv6 network, link-layer information can be provided through the
IPv6 Address [IPV6:1]. During connection setup the Bluetooth devices
will exchange their link-layer identity:

o (CoD) Class of Device (24 bits)
o (UAP) Upper Address Part (8 bits)
o (NAP) Non-significant Address Part (16 bits)
o (LAP) Lower Address Part (24 bits)

The 64-bit Interface ID of an IPv6 address will be formed by the CoD,
NAP, and LAP:

```
|                                     |
|   subnet prefix |      Interface ID      |
+----------------+------------+-----+-----+
|   64 bits       |     CoD    | NAP | LAP |
+----------------+------------+-----+-----+
```

This combination is included for three reasons. It allows:

o   IP devices to be addressed based on their function.
      (I.E.: All printers, or all fax machines)
o   there to be no unused bits in the Interface ID.
o   the Interface ID to be site unique.

Accessing devices based on their function (CoD) allows network
operators to account, monitor and control IP devices based on their
function. A network operator may send a request to all IP devices of
a particular type by setting bits in the interface ID, such as:
"Download maintenance records from all printers attached to the IP
network".

6. IP Routing

6.1 IP Routing Table:

Routing in a Bluetooth Network, like any other type of Master-Slave
network, is accomplished by a single device acting as the Master. The
Master must maintain a table of IP addresses to link-layer addresses
for every attached IP capable device [ROUTER:1]. The mapping may be
an IP address to a logical link-layer address (link local), or a
physical link-layer address. An implementation may use a logical
link-layer address such as the logical Connection Indentifier (CID)
provided by L2CAP.

For a dynamically assigned IP address:

```
+------------+-----+------------------+
| IP Address | CID | Bluetooth Address |
+------------+-----+------------------+
```

For a permanently assigned IP address, the IP address may be used to
uniquely identify a device:

```
+------------+-----+
| IP Address | CID |
+------------+-----+
```

6.2 IP Unicast:

On a Bluetooth Piconet the primary mode of communication is
point-to-point time-slot based according to a scheduling algorithm.

Slave devices may be scheduled for communications in a time-slot
manner based on the policy negotiated by the Master and Slave device
upon connection setup to reduce power consumption.

When an IP unicast packet arrives, the Master will check its IP to
link-layer routing table to ensure that the device with the correct
link-layer address is connected, before scheduling the packet for
delivery at the next available time-slot for that particular Slave
device.

6.3 IP Multi-cast:

When a single Slave device in a Piconet is a participant of an IP
Multi-cast, the Master after checking its routing table, will route
the Multi-cast packet identical to an IP Unicast packet.

For Bluetooth devices that do not support Broadcast all packets shall
be sent as per Section 6.2 IP Unicast.

If there is more than one Slave device participating on the same
Multi-cast session, then routing is identical to an IP Broadcast packet
to multiple devices [6.4.2].

6.4 IP Broadcast:

6.4.1 IP Broadcast to single device:

When only one Slave device in a Piconet is the receiver of an IP
Broadcast, after checking the routing table, routing is identical to
that of a Unicast packet.

6.4.2 **IP Broadcast to multiple devices:**

If sufficient unused bandwidth is available, the slots required for
the session shall be allocated from available unused bandwidth,
by the Bluetooth Scheduler using the Host Controller Interface.

If insufficient bandwidth is available, then the Slots required for
the session must be subtracted equally from the slots allocated ONLY
to the participants. If there is an odd number of slots per
participants, the first participants to join will supply the
remainder.

The slots must be used for a Piconet Active Member Broadcast. If
some of the participants are Parked devices, these participants
should be made active, and given sufficient bandwidth for the
Broadcast. If it is not possible to un-Park all the participants,
then a Piconet Broadcast shall be used.

6.5 **IP Anycast:**

When only one Slave device in a Piconet is the receiver of an IP
Anycast packet, after checking the routing table, routing is identical
to that of a Unicast packet.

If more than one Slave device could be the potential receiver of an IP
Anycast packet, then the Slave device with the earliest scheduled slot
will be the receiver of the Anycast IP packet.

7. MTU

7.1 Link MTU:

The Maximum Transmission Unit for an L2CAP packet is 65535 bytes. In
a Bluetooth network all L2CAP packets are delivered in-order.
Therefore, the IP header does not need to be re-transmitted for every
IP fragment sent over L2CAP. Only the first IP fragment must have an
IP header. An implementation must set the Logical Channel Code
(L_CH Code) in the Link Manager Protocol header to 0x01 to indicate
the continuation of the last IP fragment of an IP packet greater than
the L2CAP MTU size of 65535 bytes (IP header + payload). And, set the
Logical Channel Code to 0x10, to indicate the start of the next IP
packet. It is not necessary to know the end of an IP packet at the
L2CAP level, as this can be inferred from the length field in the IP
header.

There is no Maximum Transmission Unit for IP over L2CAP, when the
Logical Channel Code is used effectively. But, an implementer may
define an IP MTU that is appropriate for the intended application.

7.2 Path MTU:

IPv4 Path MTU discovery will be done in accordance with [PATHMTU:1].

When one IP host has a large amount of data to send to another host,
the data is transmitted as a series of IP datagrams. It is usually
preferable that these datagrams be of the largest size that does not
require fragmentation anywhere along the path from the source to the
destination. This datagram size is referred to as the Path MTU
(PMTU), and it is equal to the minimum of the link MTUs of each hop
in the path.

IPv6 Path MTU discovery will be done in accordance with [PATHMTU:2].

When one IPv6 node has a large amount of data to send to another node,
the data is transmitted in a series of IPv6 packets. It is usually
preferable that these packets be of the largest size that can
successfully traverse the path from the source node to the destination
node. This packet size is referred to as the Path MTU (PMTU), and it
is equal to the minimum link MTU of all the links in a path. IPv6
defines a standard mechanism for a node to discover the PMTU of an
arbitrary path.

8. GLOSSARY OF TERMS

Active Member(AM): A device with a valid active Link.

AM_ADDR: A 3-bit physical address assigned by the Master to a Slave

device during connection setup.

K. Atwal, R. Akers   [Page 11]

Access Point(AP): A wireless node that directly connects to a wireline
network, and provides network access to other wireless nodes that are
associated to the AP. Typically, the wireline network is the INTERNET
or an IP based Intranet.

Baseband: A Physical Layer Device (PHY2).

BD_ADDR: A 48-bit physical address used by a Bluetooth device for
discovery and connection setup.

Bluetooth Radio: A Radio Interface (PHY1).

Connection Handle: A 12-bit logical address assigned to a physical
link by the Host Controller.

Frequency-Hopping Radio: A radio that uses a Frequency Hopping
Sequence for communications.

Frequency-Hopping Sequence: A set of radio frequencies that are
selected in pseudo-random order to tune a radio to a new frequency
once per Slot-Time.

Host Controller: A controller consisting of the MAC(MAC1 + MAC2) and
PHY(PHY1 + PHY2).

Host Controller Interface: An API provided by the Host Controller
Driver.

IP: Internet Protocol

L2CAP: Logical Link Control and Adaptation Protocol

Link Controller: A lower Link Layer Controller (MAC1)

Link Manager: A upper Link Layer Controller (MAC2).

Master Device(MD): The device delegated as the router on a Piconet.

Master-Slave Network: A network where a single device is delegated
the Master, and all other devices assume the role of Slave devices.
A Master must poll a Slave device by addressing a packet to a Slave
device, before a Slave device can transmit. All communications are
either Master to Slave, or Slave to Master. Since all communications
in a Master-Slave Network are between the Master and a Slave, the
Master is effectively the router. A Master-Slave Network must have
one and only one Master device. A Master-Slave network may have
broadcast capabilities.

Master/Slave Slot-Pair: The smallest integral unit of transmission
composed of a Master device Slot-Time followed immediately by a

Slave device Slot-Time

Parked Member(PM): A device with a valid dormant link.

Piconet: Two or more Bluetooth devices linked on the same
Frequency-Hopping Sequence.

Piconet Active Member Broadcast: A packet addressed to be received
by all Active Members.

Piconet Broadcast: A packet addressed to be received by all Active
and Parked Members.

PM_ADDR: A 8-bit physical address assigned by the Master to a
Parked Slave device.

RFCOMM: RS-232 Serial port emulation protocol

Scatternet: Three or more Bluetooth devices linked on at least two
Frequency-Hopping Sequences. A device may be a slave on all
sequences, or a master on one and a slave on the other.

Slave Device(SD): A device not acting as a router on a Piconet.

Slot-Time: A fixed interval of time specifying the duration that a
radio will spend tuned to one frequency before tuning to a new
frequency. For Bluetooth this is 625 microseconds with a clock
jitter of -/+ of 10 microseconds and a clock drift rate of 20
parts per million.

Un-discovered Device(UD): A device that has not been discovered.

[9]. Authors Addresses

Ronald G. Akers
Motorola Laboratories
[1301] Algonquin Rd.
Schaumburg IL. 60196
ron.akers@motorola.com

Kulwinder S. Atwal
Zucotto Wireless
[130] Slater St.
Ottawa, Ontario, Canada
kulwinder.atwal@zucotto.com

[10](#). REFERENCES

[ARP:1] RFC 826, An Ethernet Address Resolution Protocol,
        David C. Plummer, November 1982.

[HOST:1] RFC 1122, Requirements for Internet Hosts --
        Communications Layers, R. Braden (Editor), October 1989.

[HOST:2] RFC 1123, Requirements for Internet Hosts --
        Application and Support, R. Braden (Editor),October 1989.

[INTRO:1] Specification of the Bluetooth System Core Version 1.1.
(http://www.bluetooth.com/developer/specification/specification.asp)

[IANA:1] Protocol Numbers for the Internet Protocol.
(http://www.isi.edu/in-notes/iana/assignments/protocol-numbers)

[IANA:2] Port Numbers for TCP/UDP.
(http://www.iana.org/assignments/port-numbers)

[IANA:3] Parameters for the Address Resolution Protocol.
(http://www.iana.org/assignments/arp-parameters)

[IPv6:1] RFC 1884, IP Version 6 Addressing Architecture,
        R. Hinden and S. Deering, December 1995.

[PATHMTU:1] RFC 1191, Path MTU Discovery, J. Mogul and S. Deering,
        October 1989.

[PATHMTU:2] RFC 1981, Path MTU Discovery for IP version 6,
        J. McCann, S. Deering, and J. Mogul, August 1996.

[ROUTER:1] RFC 1009, Requirements for Internet Gateways,
        R. Braden and J. Postel, June 1987.

11.REFERENCE BLUETOOTH IP ARCHITECTURES

    Bluetooth devices communicate on a Master-Slave Frequency Hopping
Radio network operating within the 2.4-2.5 Giga-Hertz radio spectrum.

11.1 **Reference Piconet IP Architecture:**

  A Bluetooth Piconet can be formed between any two Bluetooth devices
engaging in a RF discovery procedure. The discovered device becomes
a "Slave" device, while the discoverer becomes the "Master".

```
                      +------+
                      |      |
                      |Master|
                      |      |
                      +------+
                         |
                         |
                      +-----+
                      |     |
                      |Slave|
                      |     |
                      +-----+
```

       Figure 11.1.1 - A simple Master-Slave Bluetooth Piconet


```
                 +------+
                 |      |
                 |Master|
                 |      |
                 +------+
                  | | | |
      +----------+ | | +------------------+ - - - - - -+
      |            | +---------+          |
      |            |           |          |            |
      |            |           |          |            |
      V            |           |          |            |
    +-----+      +-----+     +-----+     +-----+   +--------+
    |     |      |     |     |     |     |     |   | Parked |
    |Slave|      |Slave|     |Slave| o o o|Slave|   | Slaves |
    | 1   |      | 2   |     | 3   |     | 7   |   | 8-255  |
    +-----+      +-----+     +-----+     +-----+   +--------+
```
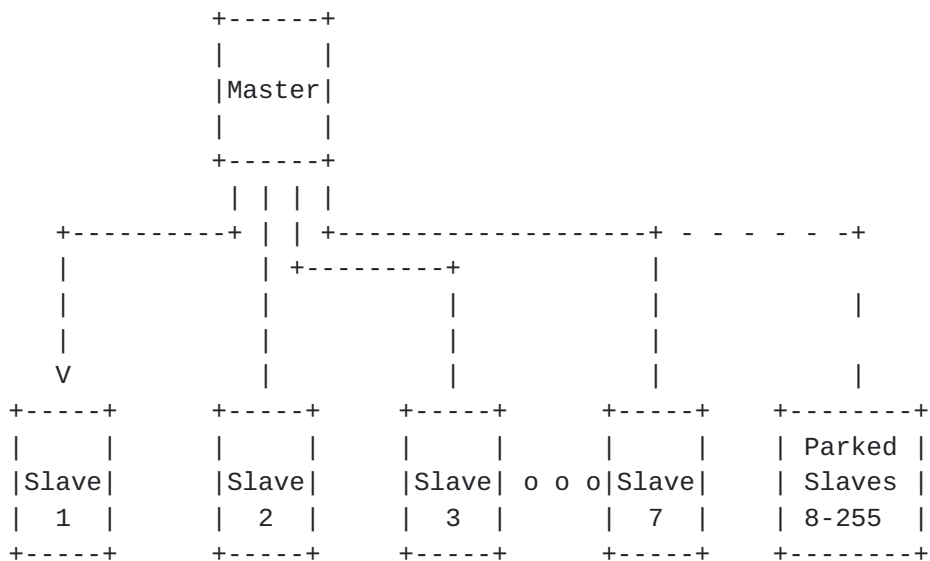
       Figure 11.1.2 - Single Bluetooth Piconet with multiple Slaves

    A Master Bluetooth node can have up to seven active slaves associated
to it at any one time. There can also be a number of "parked" slaves
associated with the Master at the same time. When a slave is in the
parked state it is not participating in normal communications. A Slave

can request to be "parked", or "un-parked". Also, a Master can "park"
any of its active nodes, and "un-park" any of it's parked nodes.

[11.2](#) **Reference Scatternet IP Architecture:**

   A Bluetooth Piconet can be formed between any two devices engaging in
a discovery procedure. The discovered device becomes the Slave device,
while the discoverer becomes the Master. A Scatternet can be formed
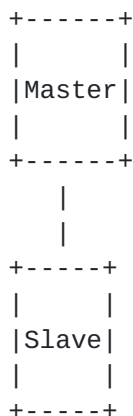between at least three devices, where at least one device is able to
switch between the hopping sequences of the other two devices.

```
                              +------+
                              |      |
                              |Master|
                              |      |
                              +------+
                                |   |
                     +----------+   |
                     |              |
                     |          +-----+
                     |          |
                 +---------+    +-------+
                 | Slave 1 |    |       |
                 | - - - - |    |Slave 2|
                 | Master  |    |       |
                 +---------+    +-------+
                   | | |
         +---------+ | +---------+
         |           |           |
         |           |           |
         |           |           |
      +-----+     +-----+     +-----+
      |     |     |     |     |     |
      |Slave|     |Slave|     |Slave|
      | 1   |     | 2   |     | 3   |
      +-----+     +-----+     +-----+
```
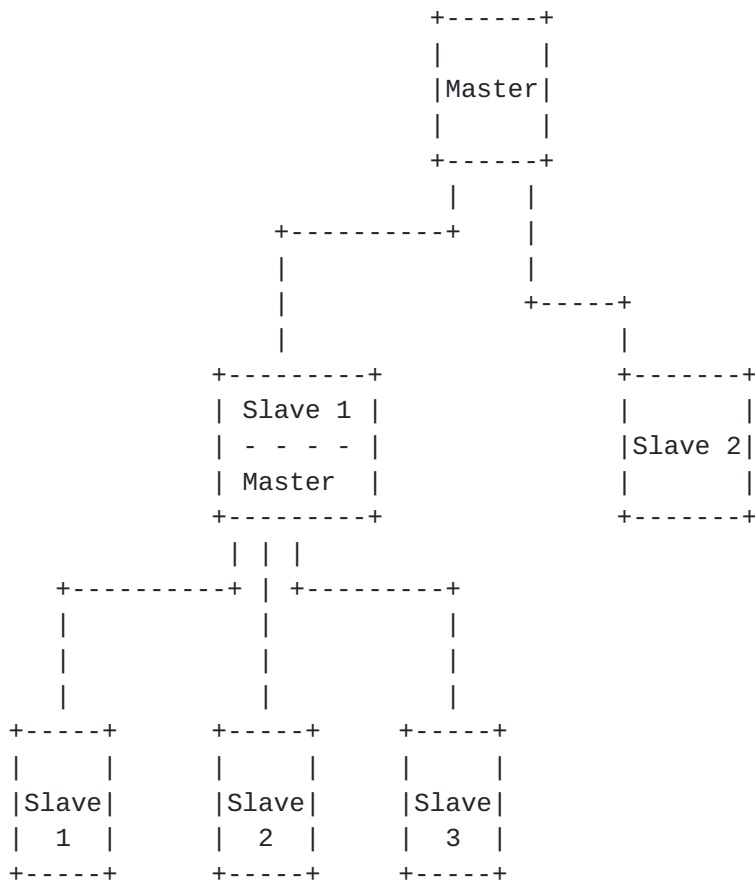
        Figure 11.2.1 - A Bluetooth Scatternet made up of two Piconets
                  by a Routing Master/Slave node

The Bluetooth node "Slave 1" is member of the upper Bluetooth Piconet.
The node is also a Master node in the lower Bluetooth Piconet. In this
way a series of Bluetooth nodes can become a multiple-hop wireless network.

11.2.2 **Scatternet Slave IP Architecture:**

```
                +-------+                    +-------+
                |Piconet|                    |Piconet|
                |   A   |                    |   B   |
                |Master |                    |Master |
                +-------+                    +-------+
                 | | |                           |
      +---------+ | +---------+   +--------+
      |           |           |   |  |
   +-----+      +-----+      +-----+
   |     |      |     |      |     |
   |Slave|      |Slave|      |Slave|
   | 1A  |      | 2A  |      |3A/1B|
   +-----+      +-----+      +-----+
```
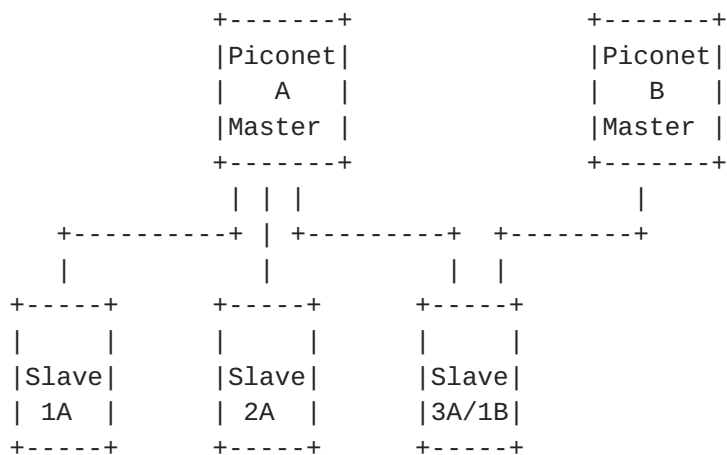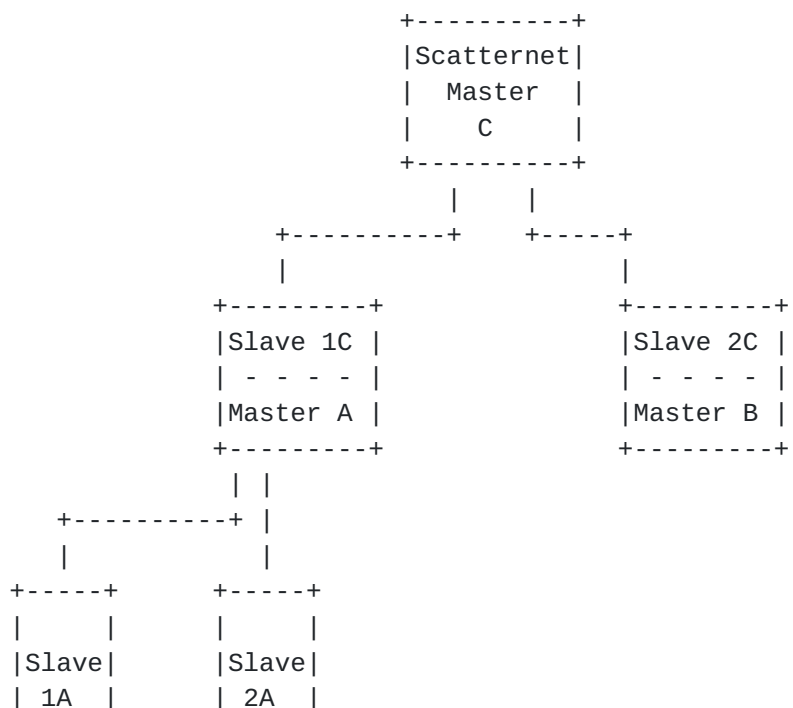
            Figure 11.2.2 - A Bluetooth Scatternet made up of two Piconets
                      by a Slave node with no Routing capability.

The Bluetooth node "Slave 3A/1B" is member of the Bluetooth Piconet A.
The node is also a member in the Bluetooth Piconet B. Since the node
is a Slave it has no active routing capability. In this way a node
can participate on two networks.

11.2.3 **Scatternet Master IP Architecture:**

If an active routing capability were desired in Slave 3A/1B it would
need to become the Scatternet Master in the above arrangement, such that:

```
                        +----------+
                        |Scatternet|
                        |  Master  |
                        |    C     |
                        +----------+
                           |    |
                  +----------+  +-----+
                  |                  |
                +---------+        +---------+
                |Slave 1C |        |Slave 2C |
                | - - - - |        | - - - - |
                |Master A |        |Master B |
                +---------+        +---------+
                   | |
         +---------+ |
         |           |
      +-----+      +-----+
      |     |      |     |
      |Slave|      |Slave|
      | 1A  |      | 2A  |
```

```
   +-----+      +-----+
```

Figure 11.2.3 - A Bluetooth Scatternet made up of two Piconets
                by a Master node with Routing capability.