1998: création du SIG Bluetooth (Ericsson, Nokia, Toshiba, ...)

USB sans fil

PAN: Personnal Area Network: Ordinateurs personnels, PDA, téléphones, imprimantes, claviers, souris, oreillettes ...

Portée ~ 10m, au délà, WiFi

Intégration dans les périphériques nomades

- => Batterie
- => Gestion d'énergie

Sécurité

- => Code PIN
- => Clé de liaison

Bande ISM: libre, 2.4-2.4835GHz, encombrée

Problèmes de collisions (WiFi, HomeRF, ...)

=> Frequency Hopping

Réémission de paquets sur 79 canaux de 1MHz de largeur

Similarité USB: Profils

- Oreillette
- Accès Réseau
- => Drivers fournis pour le protocole, pas pour l'application

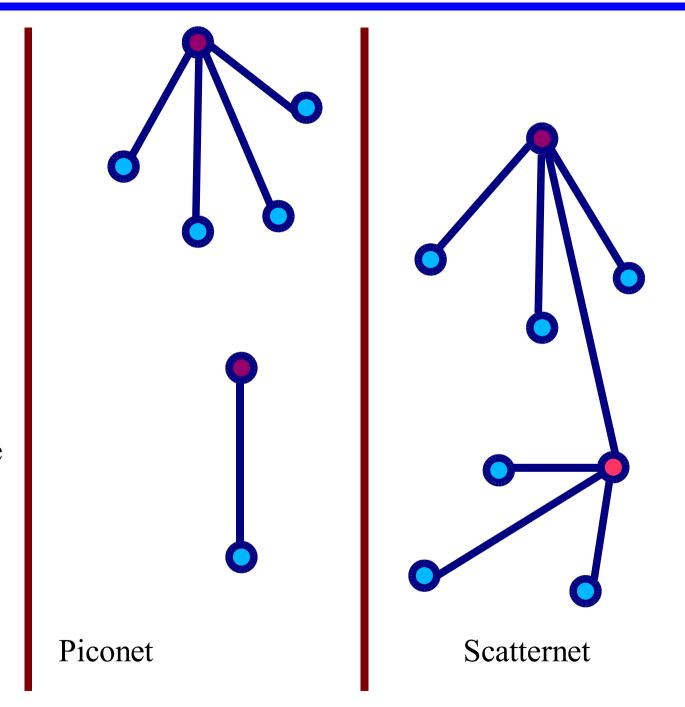
Définition d'un service de recherche de services offerts : SDP

Connexions maître-esclaves

Un maître par piconet

Un maître d'un piconet peut être esclave d'un autre piconet

=> Scatternet



| Couplage (Code PIN, Cle)     |
|------------------------------|
| Connexion                    |
| Utilisation du service       |
| Déconnexion                  |
|                              |
|                              |
| Présentation et développemen |

Périphériques étrangers :

Recherche de périphériques

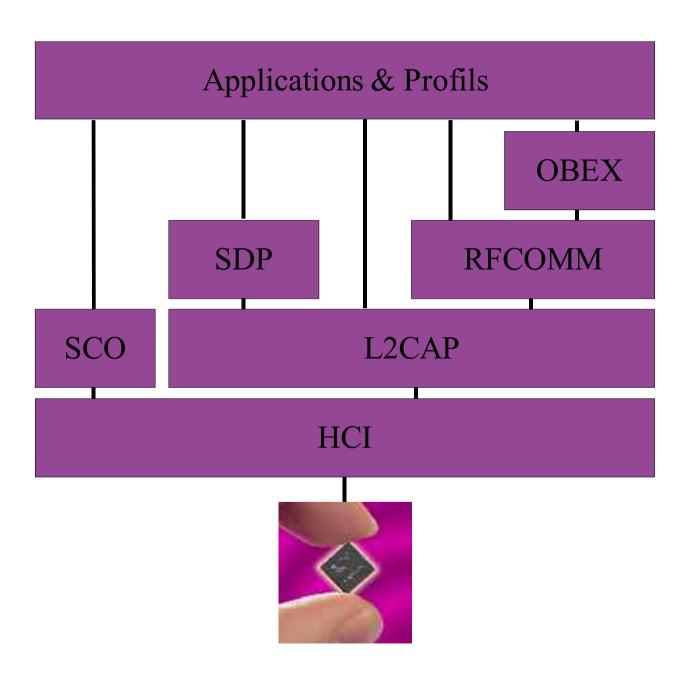
Couplege (Code DIN C16)

Recherche de services

| Connexion              |
|------------------------|
| Utilisation du service |
| Déconnexion            |
|                        |
|                        |
|                        |
|                        |

Périphérique connus:

- Adresse physique sur 6 octets => 00:A0:96:1F:B6:93
- Class de périphérique sur 3 octets
  - => identification somainre du type de périphérque
  - => information du type de service fourni
- Communiquer avec la puce :
  - => Pilote de périphérique
  - => Interface de commande définie par la norme : HCI (Host Controller Interface)



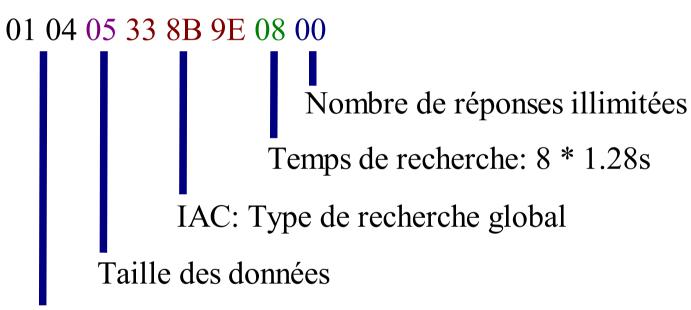
Décodage : recherche

```
< 01 01 04 05 33 8B 9E 08 00
```

- > 04 0F 04 00 01 01 04
- > 04 02 0F 01 3D 86 14 D9 0A 00 01 00 00 0C 22 10 59 43
- > 04 01 01 00
- < 01 : Envoi de commande
- > 04 : Réception d'évènement

Notation conformes au protocole H:4 de la norme

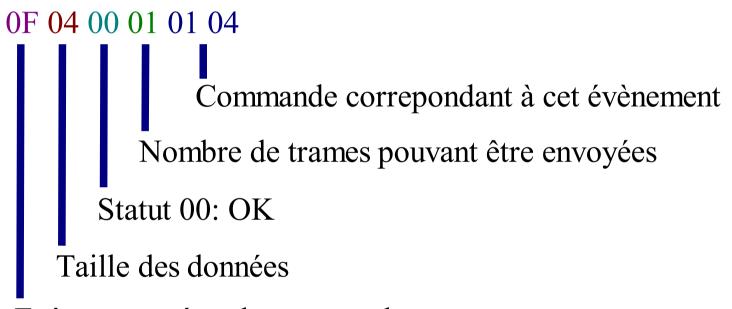
Commande: recherche



Opcode: OGF (famille) sur 6 bits et OCF (commande) sur 10 bits

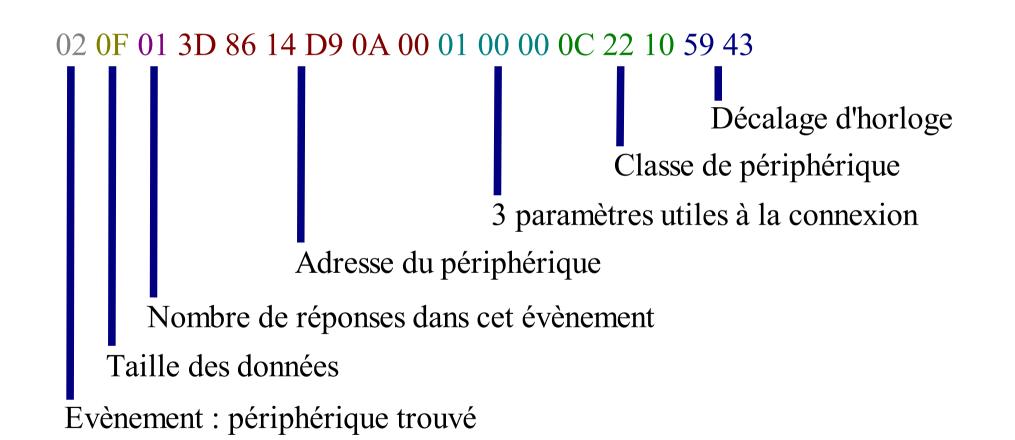
Opcode 01 04 constitué d'OGF 1 et OCF 1 : 00000100 00000001

Evènement : commande en cours

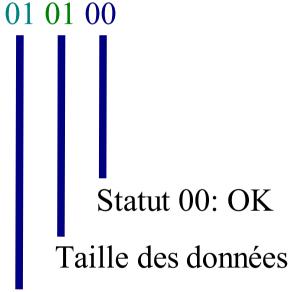


Evènement : état de commande

Evènement : périphérique trouvé



Evènement : fin de la recherche



Evènement : fin de recherche

# Décodage : Connexion / Déconnexion

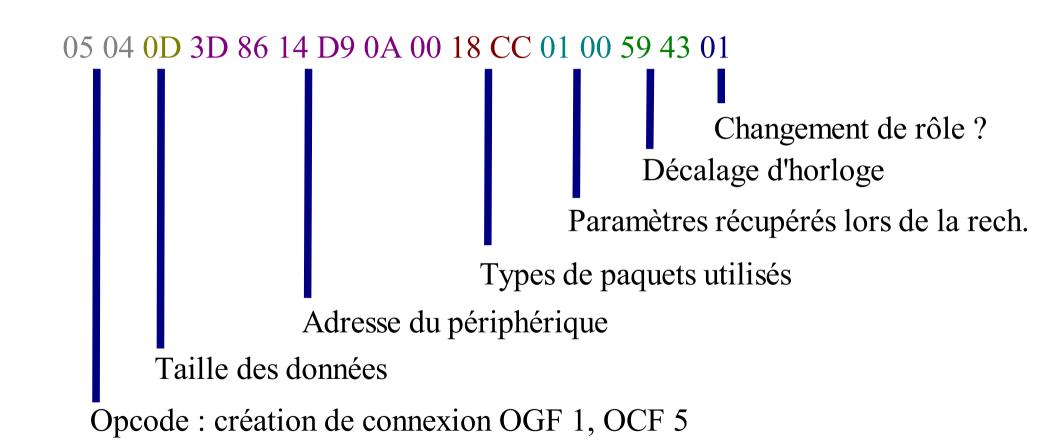
```
< 01 05 04 0D 3D 86 14 D9 0A 00 18 CC 01 00 59 43 01
```

- > 04 0F 04 00 01 05 04
- > 04 03 0B 00 28 00 3D 86 14 D9 0A 00 01 00

. . .

> 04 05 04 00 28 00 13

Commande : demande de connexion





Commande : demande de connexion

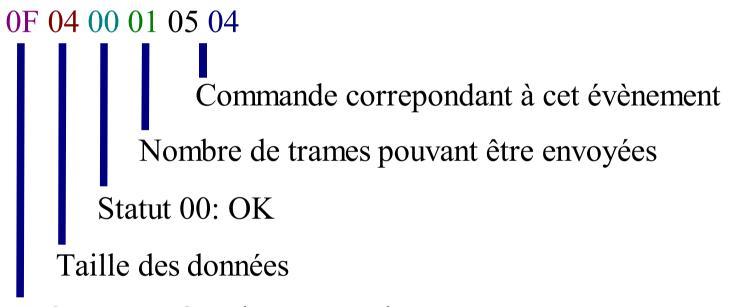
Types de paquets :

| DM1 | 0x0008 |
|-----|--------|
| DH1 | 0x0010 |
| HV1 | 0x0020 |
| HV2 | 0x0040 |
| HV3 | 0x0080 |
| DM3 | 0x0400 |
| DH3 | 0x0800 |
| DM5 | 0x4000 |
| DH5 | 0x8000 |

OU binaire pour spécifier le type de paquet à utiliser ...

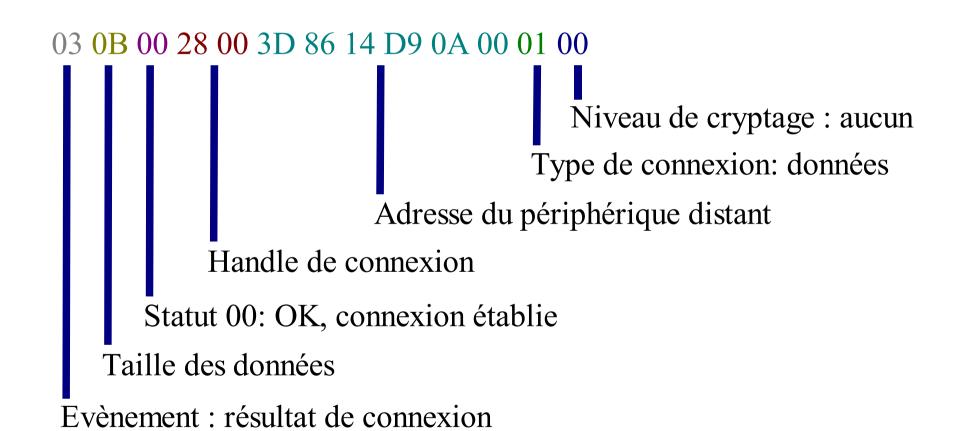
18 CC = DM1 DH1 DM3 DH3 DM5 DH5

Evènement : commande en cours

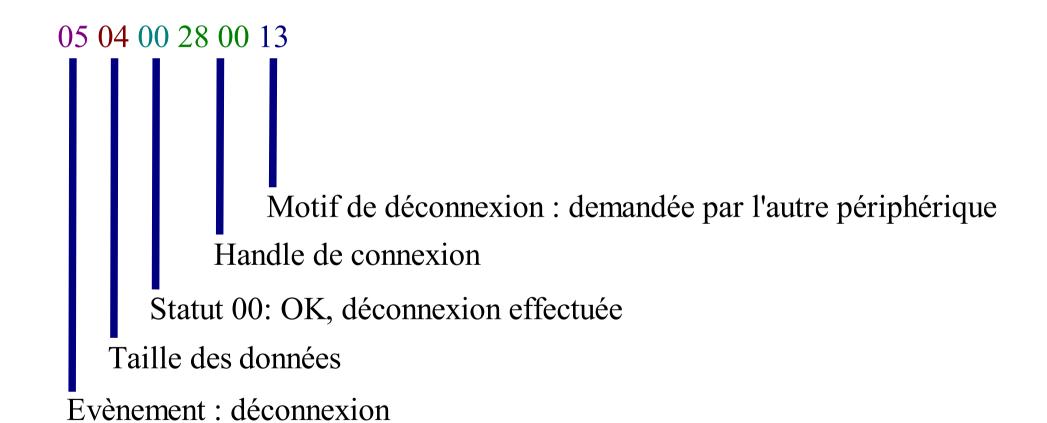


Evènement : état de commande

Evènement : résultat de connexion



Evènement: déconnexion



Premier protocole de la pile BT permettant l'échange de données

### Identifié par :

- Handle ACL
- PSM
- CID Source et CID Destination

### Paramétrage:

- Time out
- MTU
- QoS

### Cycle de vie :

- Connexion
- Configuration
- Echanges de données
- Déconnexion

## Connexion, negotiation:

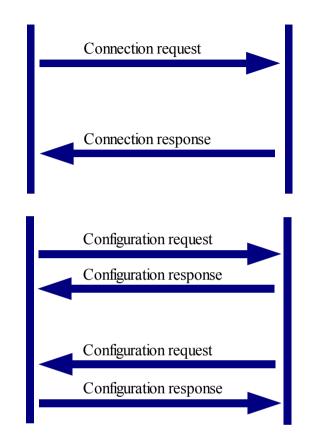
- du PSM
- des cid source et destination

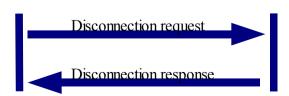
# Configuration, négotiation:

- des MTU
- des timeouts
- de la QoS

Echanges de données

Déconnexion:







### Trames de données HCI:

| Handle ACL | flags  | Longueur des données | Données |
|------------|--------|----------------------|---------|
| 12 bits    | 4 bits | 16 bits              | n bits  |

### Flags:

- BC: Broadcast flag sur 2bits

- PB: Packet boundary flag sur 2bits

#### Trames L2CAP:

| Longueur | Canal   | Données |
|----------|---------|---------|
| 16 bits  | 16 bits | n bits  |

### Canal:

- canal 1: canal de contrôle
- supérieur à 64 pour les données



- < 08 00 01 00 02 01 04 00 cc ee 40 00
- > 0c 00 01 00 03 01 08 00 50 00 40 00 00 00 00

### Connexion (demande = 02)

- identifiant de commande 1
- 4 octets de données
- PSM cc ee
- canal source 40 00

## Connexion (réponse = 03)

- identifiant de commande 1
- 8 octets de données
- canal destination 50 00
- rappel canal source 40 00
- connexion acceptée



```
< 08 00 01 00 04 02 04 00 50 00 00 00
```

- > 16 00 01 00 05 02 06 00 40 00 00 00 00 04 03 08 00 40 00 00 01 02 4e 00
- < 0a 00 01 00 05 03 06 00 50 00 00 00 00 00

### Configuration (demande = 04)

- identifiant de commande 2
- 4 octets de données
- pour canal 50 00
- configuration par défaut sans fragmentation 00 00

#### Configuration (réponse = 05)

- identifiant de commande 2
- 6 octets de données
- pour canal 40 00
- pas de fragmentation
- configuration par défaut acceptée
- Configuration (réponse = 05)
  - identifiant de commande 3
  - 6 octets de données
  - pour canal 50 00
  - pas de fragmentation
  - configuration acceptée

#### Configuration (demande = 04)

- identifiant de commande 3
- 8 octets de données
- pour canal 40 00
- pas de fragmentation
- configuration MTU = 78



- < 03 00 50 00 02 04 05
- > 03 00 40 00 02 05 04

#### Trame 1:

- 3 octets de données
- pour canal 50 00
- données "libres" 02 04 05

#### Trame 2:

- 3 octets de données
- pour canal 40 00
- données "libres" 02 05 04



- < 08 00 01 00 06 04 04 00 50 00 40 00
- > 08 00 01 00 07 04 04 00 50 00 40 00

## Déconnexion (demande = 06)

- identifiant de commande 4
- 4 octets de données
- canal distant 50 00
- canal local 40 00

## Déconnexion (réponse = 07)

- identifiant de commande 4
- 4 octets de données
- canal local 50 00
- canal distant 40 00

# SDP (Service Discovery Protocol)

- protocole d'échange de données sur les services mis à disposition par un périphérique bluetooth
- utilise une connexion L2CAP, PSM 1

## Une réponse SDP contient :

- infos sur le service (nom, identifiant)
- protocoles utilisés

Un serveur SDP stocke des "enregistrements" définissant les "services"

Un enregistrement est défini par un "handle" interne et des "attributs"

Le format utilisé pour les attributs est défini par le SDP : Attribut = "Identifiant" + "Valeur"

## Exemples d'identifiants:

- ServiceClassIDList (id 1)
- ServiceID (id 3)
- ProtocolDescriptorList (id 4)
- LanguageBaseAttributeIDList (id 6)
- ServiceName (id variable selon la langue utilisée, id de base pour la langue)
- ServiceDescription (id variable selon la langue utilisée, id de base pour la langue
- +1)



# Codage des identifiants et valeurs :

| Type   | Type de taille | Données |
|--------|----------------|---------|
| 5 bits | 3 bits         | n bits  |

### Types

| • 1 |                                   |
|-----|-----------------------------------|
| 0   | Nil, type NUL                     |
| 1   | Entier non signé                  |
| 2   | Entier signé                      |
| 3   | UUID                              |
| 4   | Chaîne de caractères              |
| 5   | Valeur booléenne                  |
| 6   | Séquence d'éléments               |
| 7   | Liste d'éléments (un seul valide) |
| 8   | URL                               |
|     |                                   |

## Types de taille

| 0 | 1 octet de données (ou 0 dans le cas du type nil)       |
|---|---|
| 1 | 2 octets de données                                     |
| 2 | 4 octets de données                                     |
| 3 | 8 octets de données                                     |
| 4 | 16 octets de données                                    |
| 5 | La taille est contenue dans l'octet qui suit            |
| 6 | La taille est contenue dans les deux octets qui suivent |
| 7 | La taille est contenue dans les 4 octets qui suivent    |



### Communication via des PDU (Protocol Data Unit):

| ID PDU | ID Transaction | Longueur | Paramètres |
|--------|----------------|----------|------------|
| 8 bits | 16 bits        | 16 bits  | n bits     |

### Types de PDU SDP:

- 1 : SDP\_ErrorResponse
- 2 : SDP ServiceSearchRequest
- 3 : SDP ServiceSearchResponse
- 4 : SDP ServiceAttributeRequest
- 5 : SDP\_ServiceAttributeResponse
- 6 : SDP\_ServiceSearchAttributeRequest
- 7 : SDP\_ServiceSearchAttributeResponse

Protocole d'émulation de ports séries (norme ETSI TS 101 368 ou TS 07.10)

Transit des données et des signaux de contrôle

Multiplexage de 60 liaisons max par session

Liaison identifiée par un dlci ou canal RFCOMM dlci 0 réservé pour signaux de contrôle et gestion de session

Credit based flow control : affectation de "jetons" pour limiter le flux de données

Utilise une connexion L2CAP, PSM 3

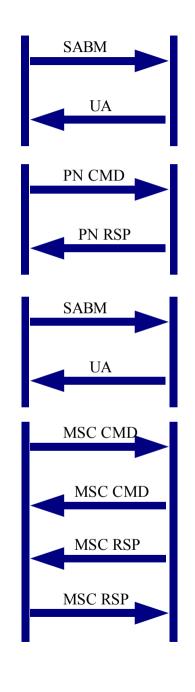


Initiation de session - création du dlci 0 SABM UA

Négotiation de paramètres (crédits) PN CMD PN RSP

Création d'un canal de communication SABM UA

Initialisation du canal MSC CMD (2) MSC RSP (2)

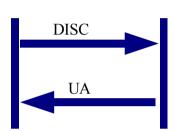




Transmission de données et/ou de crédits UIH



Fermeture de canaux et déconnexion DISC UA



## Implémentation RFCOMM:

- possibilité d'ajouter des ports virtuels au système d'exploitation
- possibilité d'utiliser les protocoles fonctionnant sur liaison série (terminaux, ppp/ip/udp/tcp/..., obex, ...)

### Profils lors de la norme 1.1:

- GENERIC ACCESS PROFILE
- SERVICE DISCOVERY APPLICATION PROFILE
- CORDLESS TELEPHONY PROFILE
- INTERCOM PROFILE
- SERIAL PORT PROFILE
- HEADSET PROFILE
- DIAL-UP NETWORKING PROFILE
- FAX PROFILE
- LAN ACCESS PROFILE
- GENERIC OBJECT EXCHANGE PROFILE
- OBJECT PUSH PROFILE
- FILE TRANSFER PROFILE
- SYNCHRONIZATION PROFILE

Aujourd'hui, 56 profils définis.