

NFC & Android

Nouvelle Façon de Communiquer ?

Guillaume Gerbaud
DevFest 09.11.2012

- Consultant et formateur Android
- Responsable comité technique Mobilité

 guillaume.gerbaud@zenika.com

 @ggerbaud

 <http://blog.zenika.com>

- Qu'est ce que le NFC
- Fonctionnement
- Usages
- NFC et Android

NFC ?



- Near Field Communication
 - Communication sans contact
 - En champs proche

Nouvelle Technologie ?



- NFC spécifié depuis 2004
- Basé sur la technologie RFID (Radio Frequency IDentification)
 - Premier brevet français en 1983
- Compatible SmartCard
 - Fin 70' - début 80'

- 2004
 - Création du NFC Forum
 - Sony, Philips (NXP)
- 2006
 - Api Java pour NFC : JSR-257
- 2009
 - Spécification d'un mode P2P
- 2010
 - Premier Androphone NFC

- Fondé en 2004 (Sony, NXP, Nokia)
- 140 membres (Google, Visa, FT, Paypal)
- Définir des standards
- Assurer l'interopérabilité

- Qu'est ce que le NFC
- Fonctionnement
- Usages
- NFC et Android

- Interaction entre un lecteur et une puce
 - Un Tag
- La puce est alimentée par le champs magnétique du lecteur
 - RFID : plusieurs dizaine de mètres
 - Identification, antivols, télépéages d'autoroutes
 - NFC : une dizaine de centimètres

- L'utilisateur approche son téléphone d'un Tag
- Le téléphone doit être actif



Avantages et inconvénients



- Interaction volontaire
- Géolocalisation précise
- Échanges standardisés

- Stockage faible (8ko max)
 - Mais supérieur aux codes barres 2D

- NFC utilise une des bandes de fréquence du RFID
 - 13,56 MHz (ISO 14443)
 - Unique bande standardisée
 - Portée théorique de 1 mètre

3 modes de communication

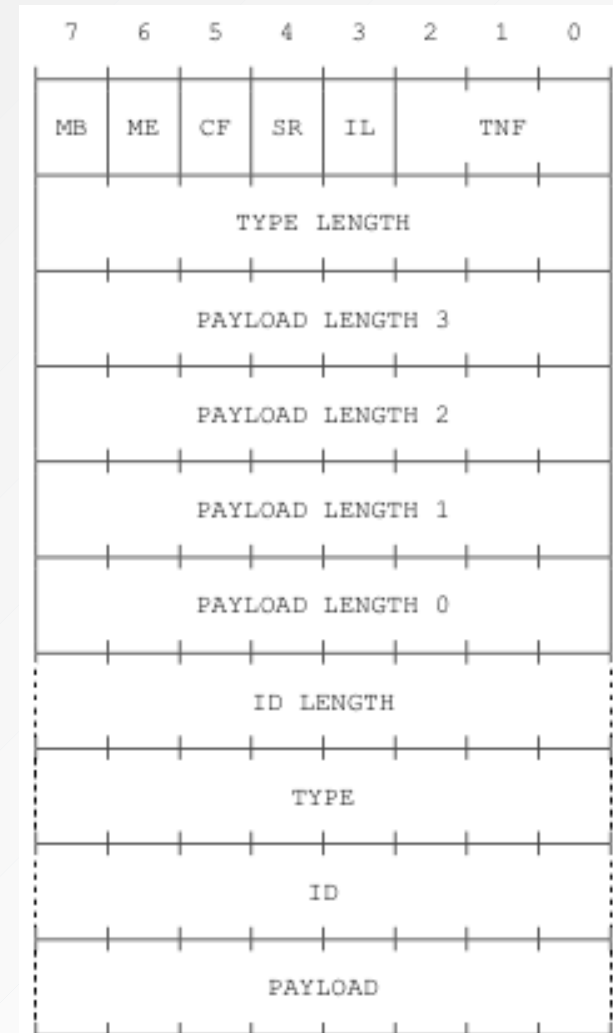


- Reader / Writer
- Card Emulation
 - Émulation d'un Tag ou d'une carte à puce
- P2P
 - Échanges pair à pair

- NFC Data Exchange Format
- Un tag contient 1 ou plusieurs messages NDEF
- Chacun contient 1 ou plusieurs enregistrements (Record)

Record

- Chaque Record possède un en-tête
 - Premier ou dernier enregistrement
 - Taille réduite
 - Tronqué
 - TNF
 - Tailles
 - Type
- Suivi de son contenu (Payload)



- Type Name Format
- Comment est décrit le type
 - 0x00 : vide
 - 0x01 : NFC Forum (Well-Known Type)
 - 0x02 : type MIME
 - 0x03 : Uri absolue
 - 0x04 : type externe
 - 0x05 : type inconnu

- Uri
 - Type : U (0x55)
 - Préfixe de l'Uri sur un octet dans le payload
 - 0x01 : http://www.
 - 0x03 : http://
 - 0x05 : tel:
 - 0x06 : mailto:
 - 0x1D : file://

```
D1 01 0B 55 01  
7A 65 6E 69 6B => zenik  
61 2E 63 6F 6D => a.com
```

```
http://www.zenika.com
```

- Text
 - Type : T (0x54)
 - Premier octet du payload
 - Encodage et taille de la langue (« fr » : 2, « en-US » : 5)
 - N octets suivants
 - La langue
 - Les octets suivants
 - Le texte (UTF-8 ou UTF-16)

- Qu'est ce que le NFC
- Fonctionnement
- Usages
- NFC et Android

- Expérience utilisateur et compléments d'information
 - Musées, Monuments
 - Quai Branly
 - Horaires de bus
 - Étiquettes de supermarché
 - Casino et E.Leclerc

- Marketing
 - Coupon réduction
 - Carte de fidélité



- Authentification et Ticketing
 - Entrée de concert et d'événements
 - Ticket de transport
 - Déverrouillage porte

- Déclenchement d'actions
 - Émission d'un appel
 - Envoi d'un sms
 - GPS
 - Appariement Bluetooth ou Wifi



- Paiement
 - Commerces
 - Ticket de bus (à l'arrêt de bus, dans le bus)

- Échanges (P2P)
 - Uri, Texte
 - Cartes de contacts
 - Fichiers



- Avantages d'un téléphone
 - 1 seul objet
 - 1 objet intelligent
 - Sur soi
 - Accès à d'autres services (Internet, GPS, capteurs, ...)

- Qu'est ce que le NFC
- Fonctionnement
- Usages
- NFC et Android

Android en chiffres



- 36% des smartphones en France (fin 2011, +16pts sur 1 an)
 - Source : comScore

- 42% de PdM en France (Q1 2012, +7pts sur 1 an)
 - Source : Kantar Worldpanel

- Lecture et écriture des Tags
 - Depuis la version 9 (Gingerbread, 12/2010)
 - Amélioré en version 10

- Beam
 - Depuis la version 14 (Ice Cream Sandwich, 10/2011)

- Handover et partage de médias
 - Depuis la version 16 (Jelly Bean, 07/2012)

- Configurer son application (AndroidManifest.xml)
 - Obtenir la permission

```
<uses-permission android:name="android.permission.NFC"/>
```

- Cibler uniquement les appareils compatibles

```
<uses-feature android:required="true" android:name="android.hardware.nfc"/>
```

- Préciser la version minimum du SDK

```
<uses-sdk android:minSdkVersion="10" />
```

- Android transforme les Tags détectés en Intent
- L'application n'a pas besoin d'être en tâche de fond

- Les applications indiquent les Tags souhaités
 - Tous les Tags formatés Ndef

```
<intent-filter>  
  <action android:name="android.nfc.action.NDEF_DISCOVERED"/>  
  <category android:name="android.intent.category.DEFAULT"/>  
</intent-filter>
```

- Les Tags url pointant vers <http://blog.zenika.com/>*

```
<intent-filter>  
  <action android:name="android.nfc.action.NDEF_DISCOVERED"/>  
  <category android:name="android.intent.category.DEFAULT"/>  
  <data android:scheme="http" android:host="blog.zenika.com"/>  
</intent-filter>
```


- Une application au premier plan peut court-circuiter le tag dispatch

```
@Override
protected void onResume() {
    super.onResume();

    [...]

    mNfcAdapter.enableForegroundDispatch(this, pendingIntent, filters, techs);
}

@Override
protected void onPause() {
    mNfcAdapter.disableForegroundDispatch(this);

    super.onPause();
}
```

MonActivityNFC.java

- Le tag et son contenu se trouvent dans l'Intent reçu
 - `onNewIntent(...)` ou `onCreate(...)`

```
Parcelable[] rawMsgs = intent.getParcelableArrayExtra(NfcAdapter.EXTRA_NDEF_MESSAGES);  
if(rawMsgs != null && rawMsgs.length > 0) {  
    NdefMessage message = (NdefMessage) rawMsgs[0];  
    NdefRecord[] records = message.getRecords();  
    if(records.length > 0) {  
        NdefRecord record = records[0];  
        [...]  
    }  
}
```

MonActivityNFC.java

- Création d'un message

```
String uri = "blog.zenika.com";  
byte[] uriField = uri.getBytes();  
byte[] payload = new byte[uriField.length + 1];  
payload[0] = 0x03;  
System.arraycopy(uriField, 0, payload, 1, uriField.length);  
  
NdefRecord record = new NdefRecord(NdefRecord.TNF_WELL_KNOWN, NdefRecord.RTD_URI, new byte[0], payload);  
  
NdefMessage msg = new NdefMessage(new NdefRecord[]{record});
```

- Raccourcis

```
String uri = "http://blog.zenika.com";  
  
// depuis API 14  
NdefRecord record = NdefRecord.createUri(uri);  
  
// depuis API 16  
NdefMessage msg = new NdefMessage(record);
```

- Après la détection du Tag

```
Tag tag = intent.getParcelableExtra(NfcAdapter.EXTRA_TAG);  
final Ndef ndef = Ndef.get(tag);  
new Thread(new Runnable() {  
  
    @Override  
    public void run() {  
        try {  
            ndef.connect();  
            try {  
                ndef.writeNdefMessage(msg);  
            }  
            catch(final FormatException e) {}  
            ndef.close();  
        }  
        catch(final IOException e) {}  
    }  
}).start();
```

- Enregistrement indiquant l'application à utiliser
- L'application est téléchargée si nécessaire
- Spécifique à Android
 - Depuis API 14

```
NdefRecord aarRecord = NdefRecord.createApplicationRecord(context.getPackageName());
```

```
NdefRecord aarRecord = NdefRecord.createApplicationRecord("com.mobile.zenika");
```

- Permet l'échange direct de messages Ndef entre 2 appareils Android
- Intervention de l'utilisateur nécessaire pour chaque message

```
public class MainBeamCallbackActivity extends Activity implements CreateNdefMessageCallback {  
  
    private NfcAdapter mNfcAdapter;  
  
    @Override  
    public void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.main);  
  
        mNfcAdapter = NfcAdapter.getDefaultAdapter(getApplicationContext());  
  
        // Register callback  
        mNfcAdapter.setNdefPushMessageCallback(this, this);  
    }  
  
    @Override  
    public NdefMessage createNdefMessage(NfcEvent event) {  
        NdefRecord record = NdefRecord.createUri("http://blog.zenika.com");  
        NdefMessage msg = new NdefMessage(new NdefRecord[] {record});  
        return msg;  
    }  
}
```

Partage de médias

- Partager de fichiers (depuis JB)
- Initié via Beam (Connection Handover)
- Puis transfert par Bluetooth ou Wifi



Card Emulation



- Le téléphone agit comme une carte à puce
- L'application doit pouvoir accéder à un élément sécurisé
- Aucune API Android ne le permet

Questions ?



 guillaume.gerbaud@zenika.com

 [@ggerbaud](https://twitter.com/ggerbaud)

 <http://blog.zenika.com>

- <http://blog.zenika.com/?tag/nfc>
 - Articles concernant NFC
- http://www.nfc-forum.org/specs/spec_list/
 - Spécifications de NFC Forum
- <http://d.android.com/guide/topics/nfc/index.html>
 - Présentation de l'API NFC
- <http://d.android.com/reference/android/nfc/package-summary.html>
 - Javadoc

- <http://nfctags.tagstand.com/>
 - Boutique Tags NFC (Starter Kit avec Reader)
- <http://blog.ribomation.com/droid-at-screen/>
 - Droid@Screen Pour afficher le téléphone sur PC