

Near Field Communication (NFC)

Android München

21.3.2011

anselm@garbe.us

What is NFC?

- **Short range (<4cm)** wireless technology
- Operating rates: 108 kbit/s - 848 kbit/s
- Modes:

Passive Communication: Initiator creates RF (radio frequency) field that powers target

- **Active Communication:** Initiator and target alter their own RF field depending on reading/writing

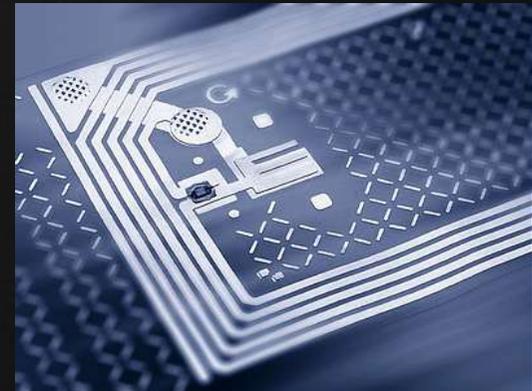
- Uses:

- **Card emulation:** device emulates contactless card
- **Reader:** device reads passive targets like RFIDs
- **P2P:** devices exchange information via a connection

RFID Tag reading



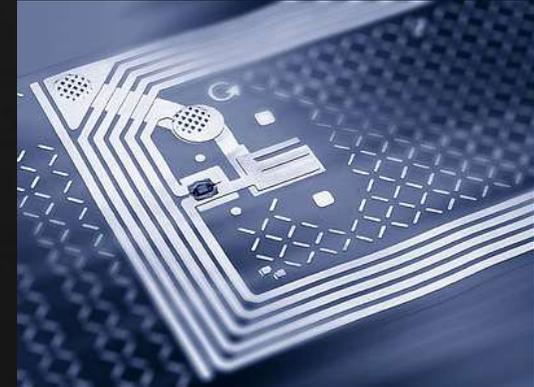
Reads



RFID Tag writing



Writes



Only some RFID tags are writable and/or killable

Card emulation

Reading an emulated RFID Tag



Reads



Bonding: Holding the devices close enough together

P2P

Exchanging data through a P2P connection



Quick history

- Very first RFID (radio frequency identification) technology appeared during **WW2**
- NFC developed by **NXP (Philips)**, and **Sony** since 2002, ISO13157, ISO18092
- NFC has been **piloted back in 2007/2008** already, Nokia produced some NFC-enabled devices, some French operators and banks experimented with mobile payments in Carrefour supermarkets
- NFC has **higher adoption rate in Asian** countries like Japan, mainly for mobile payments and smart poster use

What happened recently?

- **Nov 2010** Google announced android 2.3+ will have NFC support, and **Nexus S features NFC device**
- **Dec 2010** Gingerbread SDK 2.3[1,2], contains basic API (almost all features usable through JNI though)
- **Feb 2011** Gingerbread SDK 2.3.3 with extended API
- **Feb 2011** OpenNFC announced android API (not usable)
- Upcoming NFC smartphones announced
 - **Samsung** Galaxy S 2, Various **HTC** devices
 - **Apple** iPhone 5 (yes, no, maybe, ???)
 - Many **RIM** devices will feature NFC

What can we do with NFC?

- Mobile payments (rumors Google piloting this now)
- Electronic ticketing
- Electronic access control (doors, cars)
- Reading/Writing RFID Tags (URLs, Text, vcards, image data)
- Exchanging small data via P2P
- **Establishing a Bluetooth connection** or AdHoc WiFi connections through NFC bonding

Some application ideas

- Message in the bottle
- Geocaching tracker
- Speed dating app
- Make friends on the run (feed social networks)
- Social gaming (for example a Scoreloop challenge)
- **bu.mp killer app**

IMHO:

-> **FUN** and **convenience** apps for now...

BUT will mobile payment apps succeed? Who knows...

android.nfc

- **NfcManager**: returns NfcAdapter
- **NfcAdapter**: controls NFC lifecycle of an activity
 - enables/disables NDEF (NFC data exchange format)
Tag pushing (for card emulation and P2P issuing)
- Current (2.3.3) limitations
 - NDEF pushing **can only be performed** by foreground activities
 - Rather low level **NdefMessage/NdefRecord** handling

android.nfc.tech

- **TagTechnology**: interface to Tag properties and I/O
 - **IsoDep**: ISO 14443-4
 - **MifareClassic**: MIFARE Classic
 - **MifareUltralight**: MIFARE Ultralight
 - **Ndef**: NDEF content and operations on a Tag.
 - **NdefFormatable**: NDEF format operations on a Tag.
 - **NfcA**: NFC-A (ISO 14443-3A)
 - **NfcB**: NFC-B (ISO 14443-3B)
 - **NfcF**: NFC-F (JIS 6319-4)
 - **NfcV**: NFC-V (ISO 15693)

android.nfc.tech (cont'd)

- Usual TagTechnology methods:

- Connection related: `connect()`, `close()`, `isConnected()`

Properties related: `getTag()`

- Actual Tag (IsoDep, NfcA, etc) methods implement

TagTechnology and also provide methods for:

- Data exchange: `byte[] transceive(byte[] raw);`
- Technology specific: `getSak()`, `getAtqa()`

android.nfc where to look at first

Getting started

1. Google's NFC Demo sample
2. Gingerbread source, Tag application
3. My examples perhaps ;)

Advanced power users

1. Custom JNI code around [libnfc-nxp](#)
2. [OpenNFC](#) (if it works ;))
3. <http://rfidiot.org/> (tools, devices, non-android)

Using NFC/Receiving Tag intents

Android.xml:

```
<uses-permission android:name="android.permission.NFC" />
<uses-sdk android:minSdkVersion="9" />
<uses-feature android:name="android.hardware.nfc" android:required="true" />

<activity android:name=".ReceiverActivity" >
<intent-filter>
  <action android:name="android.nfc.action.TAG_DISCOVERED" />
<!-- <data android:mimeType="text/plain" /> -->
  <category android:name="android.intent.category.DEFAULT" />
</intent-filter>
</activity>
```

ReceiverActivity (simplified)

```
protected void onCreate(Bundle savedInstanceState) {  
    Intent intent = getIntent();  
    if (NfcAdapter.ACTION_TAG_DISCOVERED.equals(intent.getAction())) {  
        byte[] id = intent.getByteArrayExtra(NfcAdapter.EXTRA_ID); // tag id  
        Parcelable[] raw;  
        raw = intent.getParcelableArrayExtra(NfcAdapter.EXTRA_NDEF_MESSAGES);  
        NdefMessage msg = (NdefMessage) raw[0];  
        NdefRecord[] records = msg.getRecords();  
        String text = NdefHelper.parse(records[j]);  
    }  
}
```

SenderActivity (simplified)

```
public void onCreate(Bundle savedInstanceState) {  
    manager = (NfcManager) getSystemService(NFC_SERVICE);  
    adapter = manager.getDefaultAdapter();  
}  
  
protected void onResume() {  
    NdefRecord records[] = new NdefRecord[] {  
        NdefHelper.newTextRecord("Hallo Android München",  
            Locale.getDefault(), true)  
    };  
    adapter.enableForegroundNdefPush(this, new NdefMessage(records));  
}  
  
protected void onPause() {  
    adapter.disableForegroundNdefPush(this);  
}
```

NdefHelper.newTextRecord

```
public static NdefRecord newTextRecord(String text, Locale locale, boolean inUtf8) {  
    byte[] langBytes = locale.getLanguage().getBytes(Charset.forName("US-ASCII"));  
    Charset utfEncoding = inUtf8 ? Charset.forName("UTF-8") : Charset.forName("UTF-16");  
    byte[] textBytes = text.getBytes(utfEncoding);  
    int utfBit = encodeInUtf8 ? 0 : (1 << 7);  
    char status = (char) (utfBit + langBytes.length);  
    byte[] statusBytes = new byte[] { (byte) status };  
    byte[] data = new byte[statusBytes.length + langBytes.length + textBytes.length];  
    System.arraycopy(statusBytes, 0, data, 0, statusBytes.length);  
    System.arraycopy(langBytes, 0, data, statusBytes.length, langBytes.length);  
    System.arraycopy(textBytes, 0, data, statusBytes.length + langBytes.length, textBytes.length);  
    return new NdefRecord(NdefRecord.TNF_WELL_KNOWN, NdefRecord.RTD_TEXT, new byte[0], data);  
}
```

NdefHelper.parse

```
public static String parse(NdefRecord record) throws UnsupportedOperationException {
    String result = null;
    if (record.getTnf() == NdefRecord.TNF_WELL_KNOWN && record.getType() == NdefRecord.RTD_TEXT) {
        byte[] payload = record.getPayload();
        /*
         * payload[0] contains the "Status Byte Encodings" field, per the
         * NFC Forum "Text Record Type Definition" section 3.2.1.
         * if (Bit_7 == 0): text UTF-8 encoded if (Bit_7 == 1): UTF-16
         * Bit_6 is reserved for future use and must be set to zero.
         * Bits 5 to 0 are the length of the IANA language code.
         */
        String textEncoding = ((payload[0] & 0200) == 0) ? "UTF-8" : "UTF-16";
        int languageCodeLength = payload[0] & 0077;
        result = new String(payload, languageCodeLength + 1, payload.length - languageCodeLength - 1, textEncoding);
    }
    return result;
}
```

Some recommendations

- **Make receiver activity separate from UI activity** that lists received tags, call finish() when receiver activity has read the tag (if no P2P is required)
- **Filter for mime types that you actually support** (annoying side effect android's Tag activity has wildcard intent filter)
- **Limit foreground dispatching/NDEF pushing time** to the bare minimum (saves battery of course)

Demo time



Question time

Further questions?

URL: <http://garbe.us>

Email: anselm@garbe.us

OR anselm@scoreloop.com

Example code: <http://hg.suckless.org/nfc-example>

References

- <http://www.businessinsider.com/nfc-chips-2011-3>
- <http://www.businessinsider.com/iphone-5-may-have-mobile-payments-after-all-2011-3>
- <http://www.minyanville.com/dailyfeed/android-50-faster-than-iphone/>
- <http://www.dialtosave.co.uk/mobile/news/2011/03/17/foursquare-testing-nfc-for-mobile-check-ins/>
- <http://techcrunch.com/2011/03/16/mobile-wallet-nfc-overhyped/>

References II

"Google is rumoured to be planning NFC mobile payments trials in New York and San Francisco. Bloomberg has reported that sources familiar with the project say the trials will begin in the next four months and that Google will pay for the installation of "thousands" of custom-built NFC-enabled Verifone terminals at merchants across both cities."

[]<http://www.telecoms.com/25502/connected-home-google-to-launch-nfc-payments-trial/>

References III

"The Independent reports that a source at a mobile network has told them:

'The new iPhone will not have NFC, Apple told the operators it was concerned by the lack of a clear standard across the industry.'"

[]<http://www.metro.co.uk/tech/858017-apple-iphone-5-won-t-have-nfc-mobile-wallet-feature>