**NFC summary**

NFC stands for **Near Field Communication**, and as the name implies it provides a wireless communication mechanism between two compatible devices. NFC is a short range wireless technology having a range of 4cm or less for two devices to share data.

**How It Works**

Like Bluetooth and WiFi, and all manner of other wireless signals, NFC works on the principle of sending information over radio waves. Through NFC data is send through electromagnetic induction between two devices.

NFC works on the bases of tags , it allows you to share some amount of data between an NFC tag and an android powered device or between two android powered devices. Tags have various set of complexities. The Data stored in the tag can be written in a variety of formats, but android APIs are based around a NFC standard called as **NFC Data Exchange Format(NDEF).**.

**Three Modes of Operation**

Android powered devices with NFC supports following three main modes of operations −

* **Reader/Writer Mode:**

It allows the NFC device to read or write passive NFC tags.

* **P2P mode:**

This mode allows NFC device to exchange data with other NFC peers.

* **Card emulation mode:**

It allows the NFC device itself to act as an NFC card, so it can be accessed by an external NFC reader.

**How it works with Android:**

## To get a permission to access NFC Hardware add these lines to the AndroidManifest.xml

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

<uses-feature android:name="android.hardware.nfc"

android:required="true"/>

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

## Android provides an android.nfc package for communicating with another device. This package contains following classes –

1. **NdefMessage**: It represent an immutable NDEF Message.
2. **NdefRecord**: It represents an Immutable NDEF Record.
3. **NfcEvent**: It wraps information associated with any NFC event.
4. **NfcAdapter**: It represents the local NFC adapter.
5. **NfcManager:** It is a high level manager used to obtain an instance of an NfcAdapter.
6. **Tag:** It represents an NFC tag that has been discovered.

**The Tag Dispatch System**

 Android provides a special tag dispatch system that analyzes scanned NFC tags, parses them, and tries to locate applications that are interested in the scanned data. It does this by:

1. Parsing the NFC tag and figuring out the MIME type or a URI that identifies the data payload in the tag.
2. Encapsulating the MIME type or URI and the payload into an intent.
3. Starts an activity based on the intent.

**Beaming NDEF Messages to Other Devices**

Android Beam allows simple peer-to-peer data exchange between two Android-powered devices. The application that wants to beam data to another device must be in the foreground and the device receiving the data must not be locked. When the beaming device comes in close enough contact with a receiving device, the beaming device displays the "Touch to Beam" UI. The user can then choose whether or not to beam the message to the receiving device.

You can enable Android Beam for your application by calling one of the two methods:

[setNdefPushMessage()](http://developer.android.com/reference/android/nfc/NfcAdapter.html#setNdefPushMessage(android.nfc.NdefMessage, android.app.Activity, android.app.Activity...)): Accepts an [NdefMessage](http://developer.android.com/reference/android/nfc/NdefMessage.html) to set as the message to beam. Automatically beams the message when two devices are in close enough proximity.

[setNdefPushMessageCallback()](http://developer.android.com/reference/android/nfc/NfcAdapter.html#setNdefPushMessageCallback(android.nfc.NfcAdapter.CreateNdefMessageCallback, android.app.Activity, android.app.Activity...)): Accepts a callback that contains a [createNdefMessage()](http://developer.android.com/reference/android/nfc/NfcAdapter.CreateNdefMessageCallback.html#createNdefMessage(android.nfc.NfcEvent)) which is called when a device is in range to beam data to. The callback lets you create the NDEF message only when necessary.

An activity can only push one NDEF message at a time, so [setNdefPushMessageCallback()](http://developer.android.com/reference/android/nfc/NfcAdapter.html#setNdefPushMessageCallback(android.nfc.NfcAdapter.CreateNdefMessageCallback, android.app.Activity, android.app.Activity...)) takes precedence over [setNdefPushMessage()](http://developer.android.com/reference/android/nfc/NfcAdapter.html#setNdefPushMessage(android.nfc.NdefMessage, android.app.Activity, android.app.Activity...)) if both are set. To use Android Beam, the following general guidelines must be met:

1. The activity that is beaming the data must be in the foreground. Both devices must have their screens unlocked.
2. You must encapsulate the data that you are beaming in an NdefMessage object.
3. The NFC device that is receiving the beamed data must support the com.android.npp NDEF push protocol or NFC Forum's SNEP (Simple NDEF Exchange Protocol). The com.android.npp protocol is required for devices on API level 9 (Android 2.3) to API level 13 (Android 3.2). com.android.npp and SNEP are both required on API level 14 (Android 4.0) and later.

**To enable Android Beam:**

1. Create an [NdefMessage](http://developer.android.com/reference/android/nfc/NdefMessage.html) that contains the [NdefRecord](http://developer.android.com/reference/android/nfc/NdefRecord.html)s that you want to push onto the other device.
2. Call [setNdefPushMessage()](http://developer.android.com/reference/android/nfc/NfcAdapter.html#setNdefPushMessage(android.nfc.NdefMessage, android.app.Activity, android.app.Activity...)) with a [NdefMessage](http://developer.android.com/reference/android/nfc/NdefMessage.html) or call [setNdefPushMessageCallback](http://developer.android.com/reference/android/nfc/NfcAdapter.html#setNdefPushMessageCallback(android.nfc.NfcAdapter.CreateNdefMessageCallback, android.app.Activity, android.app.Activity...)) passing in a[NfcAdapter.CreateNdefMessageCallback](http://developer.android.com/reference/android/nfc/NfcAdapter.CreateNdefMessageCallback.html) object in the onCreate() method of your activity. These methods require at least one activity that you want to enable with Android Beam, along with an optional list of other activities to activate.