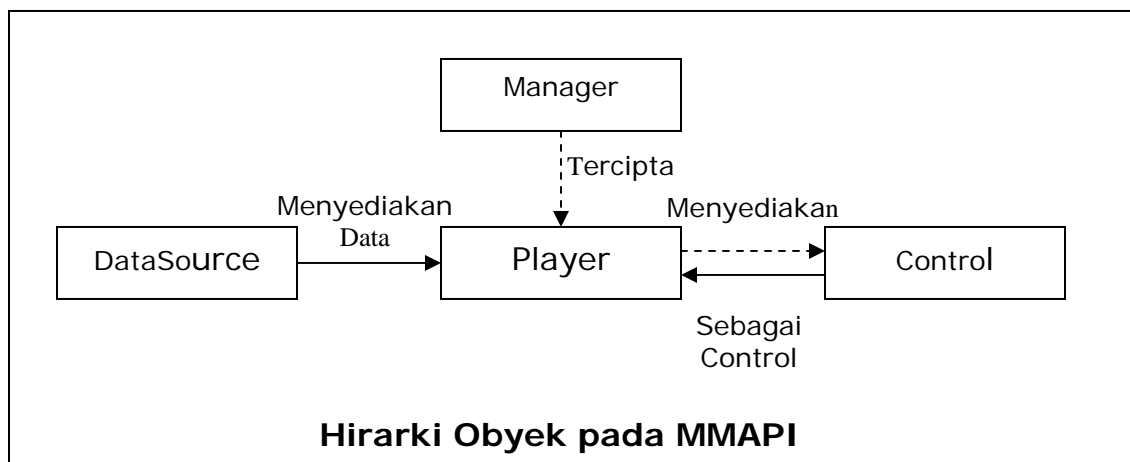


# BAB 2

## MULTIMEDIA API

### 2.1 Pendahuluan

Mobile Media API (MMAPI) memberikan dukungan lebih dari MIDlet dalam penggunaan media seperti audio dan video. Hirarki obyek pada MMAPI dapat dilihat pada gambar dibawah ini :



Manager adalah sebuah kelas yang menyediakan metode untuk menggenerasi Player dan mengakses aturan yang mendukungnya. Manager juga menyediakan metode *playTone()* untuk memainkan *tone*. Metode *createPlayer()* menggenerasi Player yang terasosiasi dengan DataSource yang menyediakan data. Player itu sendiri adalah sebuah antarmuka(*interface*). Jika player telah di generasi maka aplikasi juga dapat membuat sebuah obyek kontrol. Berikut ini adalah metode-metode yang terkait dengan obyek Player:

Metode	Deskripsi
<i>realize()</i>	Secara eksplisit mengubah kondisi Player dari kondisi <i>Unrealized</i> ke kondisi <i>Realized</i> .
<i>prefecth()</i>	Secara eksplisit mengubah kondisi Player dari kondisi <i>realized</i> ke kondisi <i>prefecth</i>
<i>start()</i>	Memberitahukan kepada Player untuk memulai memproses

	media
<i>stop()</i>	Memberitahukan kepada player untuk mengubah kondisi menjadi <i>pause</i>
<i>close()</i>	Mengubah kondisi Player menjadi <i>close</i> .
<i>deallocate()</i>	Memberitahukan player untuk melepaskan sumber daya yang digunakan.

## 2.2 Memainkan file musik

Teknologi J2ME juga dapat memainkan file musik seperti file wave (.wav) dan midi (.mid). memainkan file musik pada J2ME juga menggunakan obyek Player pada MMAPI.

```
public class PlayMusic extends MIDlet implements PlayerListener, CommandListener
{ }
```

Class Palyer :

Perhatikan kelas yang terdapat pada kelas PlayMusic, terdapat kode *implements* *PlayListener* yang berarti bahwa kelas PlayMusic adalah implementasi dari antarmuka (*interface*) *PlayListener*, oleh karena itu metoda *playerUpdate(Player player, String event, Object eventData)* harus diimplementasikan (harus ada).

Pada metode *playMedia(String file, String p)* obyek player dibuat. Obyek player ini bergantung pada file musik yang akan dimainkan, jika file musik adalah file wave maka obyek Player dibuat sebagai kode *player = Manager.createPlayer(getClass().getResourceAsStream(file), "audio/x-wav")*, jika file musik adalah midi maka obyek Player akan dibuat kode *player = Manager.createPlayer(getClass().getResourceAsStream(file), "audio/midi")*. Proses selanjutnya adalah mengeset *PlayerListener* dengan kode *player.addPlayerListener(this)* kemudian menginisialisasikan kondisi Player dengan kode :

```
player.setLoopCount(-1);
player.prefetch();
player.realize()
```

Lalu mengeset volume (nilai volume dari 0-100) dengan kode :

```
vc=(VolumeControl).player.getControl("VolumeControl");  
if(vc!=null)  
vc.setLevel(100);
```

Kemudian Player dimainkan dengan kode *player.start()*.

Untuk menghentikan Player menggunakan kode *player.stop()*.

Dan untuk menutup koneksi player gunakan kode *palyer.close()*.