


Programación de Dispositivos Móviles



Sesión 15:
Multimedia

Java y Dispositivos Móviles © 2007-2009 Depto. Ciencia Computación e IA Multimedia-1


Índice



- Reproductor de medios
- Reproducción de sonido
- Reproducción de video
- Captura

Java y Dispositivos Móviles © 2007-2009 Depto. Ciencia Computación e IA Multimedia-2

Sonido y multimedia



- Reproductor de medios
- Reproducción de sonido
- Reproducción de video
- Captura

Java y Dispositivos Móviles © 2007-2009 Depto. Ciencia Computación e IA Multimedia-3

Multimedia en J2ME



- MIDP 1.0 no soporta la reproducción de sonidos
- MIDP 2.0 permite reproducir audio
 - Incorpora subconjunto de MMAPI para audio
 - Secuencias de tonos
 - Ficheros WAV, MIDI, etc
- MMAPI permite
 - Reproducir audio
 - Reproducir video
 - Capturar audio y video
- Los dispositivos MIDP 1.0 y MIDP 2.0 que incorporen MMAPI permitirán realizar todas estas funciones

Reproductor de medios



- Nos permitirá reproducir distintos tipos de medios
- Será un objeto de la clase **Player**

```
Player player = Manager.createPlayer(  
    "http://jee.ua.es/pdm/sonido.wav");
```

- Para crearlo a partir de un recurso del JAR

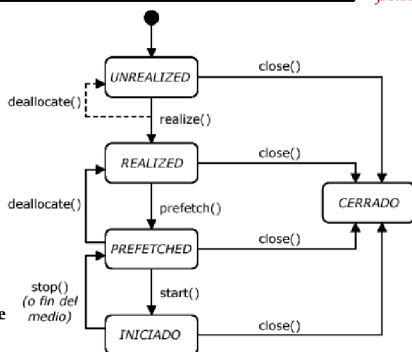
```
InputStream in =  
    getClass().getResourceAsStream("/musica.mid");  
Player player = Manager.createPlayer(in, "audio/midi");
```

- En este caso debemos proporcionar el tipo MIME

Estados



- Nos permiten ajustar la reserva de recursos
- **Unrealized**
 - No ha reservado recursos
 - Tardará en comenzar la reproducción
- **Prefetched**
 - Tiene reservados todos los recursos necesarios para comenzar
 - Puede comenzar de forma instantánea



Controles



- El reproductor de medios es genérico
 - Se podrá utilizar para cualquier tipo de medio
 - Para controlar las características concretas de un determinado tipo de medio utilizaremos controles

- Obtendremos un control con

```
Control control = player.getControl(nombre);
```

- Para poder obtener un control el reproductor deberá estar al menos en estado *realized*

- Por ejemplo, tenemos los controles

```
"VolumeControl"  
"ToneControl"  
"VideoControl"  
"RecordControl"
```

Sonido y multimedia



- Reproductor de medios
- Reproducción de sonido
- Reproducción de video
- Captura

Reproducción de tonos



- Tono simple

```
Manager.playTone(nota, duracion, volumen);
```

- Secuencia de tonos

- Crear reproductor de tonos

```
Player player =  
Manager.createPlayer(Manager.TONE_DEVICE_LOCATOR);
```

- Obtener control de tonos

```
player.realize();  
ToneControl tc =  
(ToneControl)player.getControl("ToneControl");
```

Reproducción de tonos (II)



➤ Establecer secuencia de tonos

```
tc.setSequence(new byte[] {  
    ToneControl.VERSION, 1,  
    ToneControl.TEMPO, 30,  
    ToneControl.C4, 16,  
    ToneControl.C4+2, 16,  
    ToneControl.C4+4, 16, //E4  
    ToneControl.C4+5, 16, //F4  
    ...  
});
```

➤ Comenzar la reproducción

```
pLayer.start();
```

Reproducción de ficheros



▪ Podemos reproducir músicas y sonidos de ficheros

- Los formatos soportados dependen del dispositivo
 - WAV, MIDI, MP3, etc

▪ Crear reproductor a partir de URL

```
Player player = Manager.createPlayer(  
    "http://jtech.ua.es/ctj/musica.mid");
```

▪ Crear reproductor a partir de recurso en el JAR

```
InputStream in =  
    getClass().getResourceAsStream("/musica.mid");  
Player player = Manager.createPlayer(in, "audio/midi");
```

▪ Comenzar la reproducción

```
player.start();
```

Control de volumen



▪ Podemos obtener un control de volumen

- Estará disponible en los reproductores de audio

```
player.realize();  
VolumeControl vol =  
    (VolumeControl)player.getControl("VolumeControl");
```

▪ Con este control podemos

```
vol.setLevel(volumen);  
vol.setMute(true);
```

▪ El volumen será un valor entero de 0 a 100

Sonido y multimedia



- Reproductor de medios
- Reproducción de sonido
- Reproducción de video
- Captura

Reproductor de video



- Podemos reproducir videos de distintos formatos
 - Los formatos reconocidos dependen del dispositivo
 - 3GPP, MPEG, etc



- Creamos el reproductor de video con

```
InputStream in =  
    getClass().getResourceAsStream("/video.3gp");  
Player player = Manager.createPlayer(in, "video/3gpp");
```

- Necesitaremos además un control de video
 - Nos permitirá vincular el video a la pantalla

```
player.realize();  
VideoControl vc =  
    (VideoControl)player.getControl("VideoControl");
```

Vincular el video a la pantalla



- Podemos vincular el video de dos formas
 - A un item de un formulario

```
Item item = (Item)vc.initDisplayMode(  
    VideoControl.USE_GUI_PRIMITIVE, null);  
formulario.append(item);
```

- A una región de un canvas

```
vc.initDisplayMode(VideoControl.USE_DIRECT_VIDEO, canvas);  
vc.setVisible(true);
```

- Comenzar la reproducción

```
player.start();
```

Sonido y multimedia



- Reproductor de medios
- Reproducción de sonido
- Reproducción de video
- **Captura**

Captura de medios



- **Podemos capturar audio y/o video**
 - Audio del micrófono
 - Video de la cámara
- **Crear un reproductor de medios con una URL como**
`capture://dispositivo`
- **Por ejemplo podemos utilizar**
`capture://audio`
`capture://video`
`capture://audio_video`
- **Creamos el reproductor con la URL adecuada**

```
Player player = Manager.createPlayer("capture://video");
```

Mostrar captura



- **En el caso de capturar video, podemos mostrarlo mientras se captura**
- **Esto nos permitirá ver en la pantalla el video que se está capturando**
 - Podremos tomar fotografías
- **Para mostrar el video capturado haremos lo mismo que para mostrar video de un fichero**

```
player.realize();  
VideoControl vc =  
    (VideoControl)player.getControl("VideoControl");  
vc.initDisplayMode(  
    VideoControl.USE_DIRECT_VIDEO, canvas);  
vc.setVisible(true);  
player.start();
```

Grabación de video



- Para grabar video utilizaremos un control de grabación

```
RecordControl rc =  
(RecordControl)player.getControl("RecordControl");
```

- Establecemos el flujo de salida donde grabar el video

```
ByteArrayOutputStream out = new ByteArrayOutputStream();  
rc.setRecordStream(out);
```

- Comenzamos/reanudamos la grabación

```
rc.startRecord();
```

- Detenemos la grabación

```
rc.stopRecord();
```

- Finalizamos la grabación

```
rc.commit();
```

Captura de imágenes



- Mientras se reproduce un video podemos capturar imágenes de dicho video

- Capturamos el frame actual con

```
byte [] img_png = vc.getSnapshot(null);
```

- Nos devolverá la imagen codificada en PNG

- Podemos crear la imagen con

```
Image img = Image.createImage(img_png, 0,  
img_png.length);
```

- Tomar fotografías

- Reproducir video capturado por la cámara
- Capturar imágenes de dicho video
