

## 10.4 Storing both audio and video in the same file.

The Example 10.2 given below illustrates storage and both audio and video in a single file.

---

*/\* Example 10.2. Program AVStore.java capture and stores both audio and video into a file.  
\*/*

```
import javax.media.*;
import javax.media.format.*;
import javax.media.protocol.*;
import java.io.*;
import javax.media.datasink.*;
import javax.media.control.*;

public class AVStore {
    Format formats[] = new Format[2];
    DataSink fileWriter;
    Processor p = null;

public void dothis() {

    formats[0] = new AudioFormat( AudioFormat.GSM);
    //formats[1] = new VideoFormat(VideoFormat.CINEPAK);
    formats[1] = new VideoFormat(VideoFormat.H263);

    FileTypeDescriptor outputType = new
        FileTypeDescriptor( FileTypeDescriptor.QUICKTIME );

    //formats[0] = new AudioFormat( AudioFormat.LINEAR);
    //formats[1] = new VideoFormat(VideoFormat.RGB);

    //FileTypeDescriptor outputType = new
    //    FileTypeDescriptor ( FileTypeDescriptor.MSVIDEO );

    try {
        p = Manager.createRealizedProcessor(
            new ProcessorModel( formats, outputType ) );

    } catch( NoProcessorException e ) { System.out.println(e );
    } catch( CannotRealizeException e ) { System.out.println(e );
    } catch( IOException e ) { System.out.println( e );
    }
    TrackControl track[] = p.getTrackControls();
    System.out.println("No. of tracks="+track.length);
    for (int i = 0; i < track.length; i++) {
        System.out.print("Track #"+i+": " +track[i].getFormat() + " \n" );
    }
}
```

---

```

DataSource source = p.getDataOutput();

String filename = "movie.mov" ; // movie.avi
String locator = "file://" + System.getProperty( "user.dir" )
+System.getProperty( "file.separator" ) + filename ;
MediaLocator dest = new MediaLocator(locator);

try{
    fileWriter = Manager.createDataSink( source, dest );
} catch( NoDataSinkException e ) { System.out.println("The DataSink
    Exception tgv : " + e);

} catch( SecurityException e ) { System.out.println(e); }

fileWriter.addDataSinkListener( new DataSinkListener() {

    public void dataSinkUpdate( DataSinkEvent event ) {
        if ( event instanceof EndOfStreamEvent ) {
            fileWriter.close();
        }
    }
});
*/
StreamWriterControl swc = ( StreamWriterControl)p.getControl(
    "javax.media.control.StreamWriterControl");
if( swc != null ) {
    swc.setStreamSizeLimit(1024 * 1024 );
}
*/
try{
    fileWriter.open();
    fileWriter.start();
    source.connect();
    source.start();
    p.start();

} catch( IOException e ) { System.out.println(e);
}
System.out.println( "File storage starts now \n " );
try{
    Thread.sleep(10000);
} catch( Exception e ) { System.out.println( "Error:" + e ); }
System.out.println( "File storage stops now \n " );
try{
    p.stop();
    p.close();
    fileWriter.close();
    source.disconnect();
    source.stop();
    System.exit(0);
}

```

---

```
    } catch( IOException ioe ) { System.out.println( ioe ); }  
}  
  
public static void main( String args[] ) {  
    AVStore store = new AVStore();  
    store.dothis();  
}  
}
```

---

Example 10.2 is very similar to the example 10.1. However the construction of the processor has been made very simple. We use a ProcessorModel to create a realized Processor. The ProcessorModel is defined such that it supports an input audio format of type “GSM”, and an input video format of type “H263”. As no DataSource is specified in the ProcessorModel, a Processor will be created for suitable audio and video capture devices.

The output content type descriptor is specified to be of the type fileTypeDescriptor.QUICKTIME in the ProcessorModel. The file that stores the media is given the extension “.mov”. The media is stored in the file for 10 seconds. The StreamWriterControl will be discussed in the Chapter on JMF Control.

You may use the JMStudio in order to find the allowed encoding schemes for a given file format. You can try storing the media (audio or video or both audio and video) using JMStudio. Run JMStudio and use its “capture” option followed by the “Export” option. You will be given certain options for the allowed file formats. Further for the chosen file format the valid encoding schemes for audio and /or video will also be given. For example in JMF 2.1.1 Windows performance pack you can use the MSVIDEO format (an .avi file) or QuickTime format (a .mov file) to store both audio and video.