7.3 JMF API support for capturing media.

JMF API provides the following classes and interfaces to support capturing the media.

(a) javax.media.protocol.CaptureDevice
(b) javax.media.CaptureDeviceInfo
(c) javax.media.CaptureDeviceManager

7.3.1 CaptureDevice interface

All DataSources that represent media-capturing devices implement the interface javax.media.protocol.CaptureDevice. The CaptureDevice interface defines the following methods: start(), stop(), connect(), and disconnect(). These methods override the corresponding methods belonging to the class DataSource. This overriding allows the implementers to incorporate certain device specific operations in these methods.

DataSources representing capture devices has an array of SourceStream one for each captured media stream (note: a capture device may output both audio and video streams.) The format of the captured media stream can be manipulated using its FormatControl object. The method

public FormatControl[ ] getFormatControls()

Each capture device has an object called CaptureDeviceInfo object that describes that capture device. The captureDeviceInfo object can be obtained from the DataSource representing a CaptureDevice through its method,

public CaptureDeviceInfo getCaptureDeviceInfo().

of the CaptureDevice returns an array of FormatControl objects, one for each captured media stream.

7.3.2 Class CaptureDeviceInfo

The CaptureDeviceInfo class has the following three fields for storing the information about the capture device it represents. The class also has three corresponding methods to access these fields.

(i) protected java.lang.String name

The field “name” represents the name of the capture device. The name can be accessed by using the method,

public java.lang.String getName()

(ii) protected MediaLocator locator

The "locator" field is a MediaLocator object that can be used to locate the capture device. You can use this MediaLocator to construct the DataSource representing the CaptureDevice. The construction can be achieved with the help of the Manager class.

The following method of the CaptureDeviceInfo can be used to access the locator,

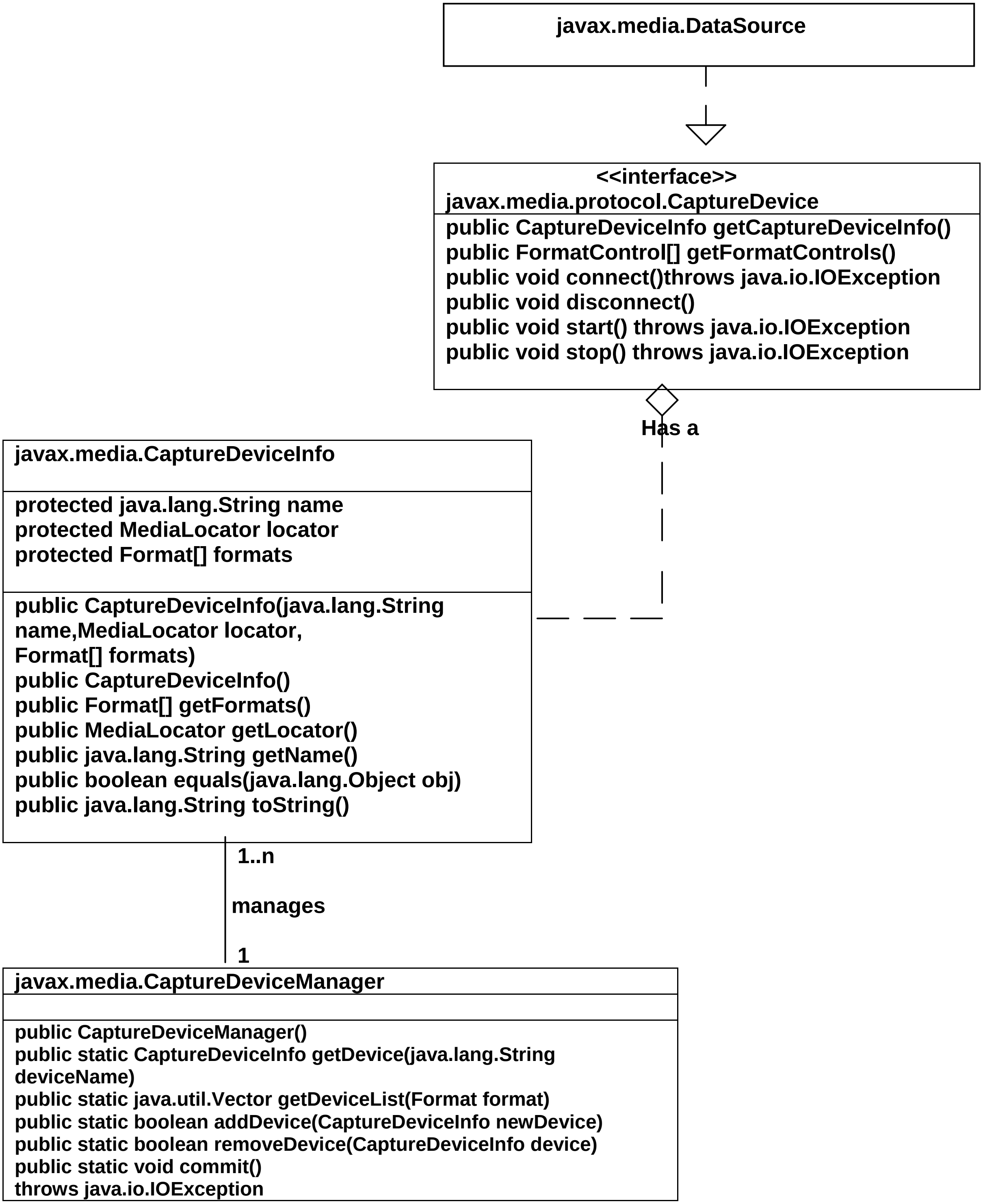
javax.media.DataSource
<<interface>>
javax.media.protocol.CaptureDevice
public CaptureDeviceInfo getCaptureDeviceInfo()
public FormatControl[] getFormatControls()
public void connect()throws java.io.IOException
public void disconnect()
public void start() throws java.io.IOException
public void stop() throws java.io.IOException
Has a
javax.media.CaptureDeviceInfo
protected java.lang.String name
protected MediaLocator locator
protected Format[] formats
public CaptureDeviceInfo(java.lang.String name,MediaLocator locator, Format[] formats)
public CaptureDeviceInfo()
public Format[] getFormats()
public MediaLocator getLocator()
public java.lang.String getName()
public boolean equals(java.lang.Object obj)
public java.lang.String toString()
1..n
manages
1
javax.media.CaptureDeviceManager
public CaptureDeviceManager()
public static CaptureDeviceInfo getDevice(java.lang.String deviceName)
public static java.util.Vector getDeviceList(Format format)
public static boolean addDevice(CaptureDeviceInfo newDevice)
public static boolean removeDevice(CaptureDeviceInfo device)
public static void commit()
throws java.io.IOException

Figure 7.2 : JMF classes and interfaces related to capturing media

public MediaLocator getLocator().

(iii) protected Format[] formats

This field is an array of Format objects that represent the output formats supported by the capture device. The array formats can be obtained through the method,
public Format[ ] getFormats()

7.3.3 The class CaptureDeviceManager

The class CaptureDeviceManager is a registry for the capture devices. It has many methods to access the registry. Use the following method of the CaptureDeviceManager,

public static java.util.Vector getDeviceList(Format format),

to find the list of all capture devices registered with the CaptureDeviceManager that support a given output format. The method returns a Vector of objects of the type CaptureDeviceInfo. Each element in that array represents a capture device that supports the specified Format "format". The name and the locatorString of the capture device can then be obtained from its CaptureDeviceInfo object.

If you know the name of a capture device its CaptureDeviceInfo object can be obtained through the method,

public static CaptureDeviceInfo getDevice(java.lang.String deviceName).

Methods addDevice and removeDevice of the captureDeviceManager can be used to add or remove capture devices by passing as parameter the CaptureDeviceInfo object of the device to be added or removed. The addDevice and removeDevice method has to be followed by the commit method in order to make the change in the captureDEviceManager registry a permanent one.

The following program lists out all the capture devices that are registered with the capture device manager. For each capture device the program prints a list of all the supported output formats.