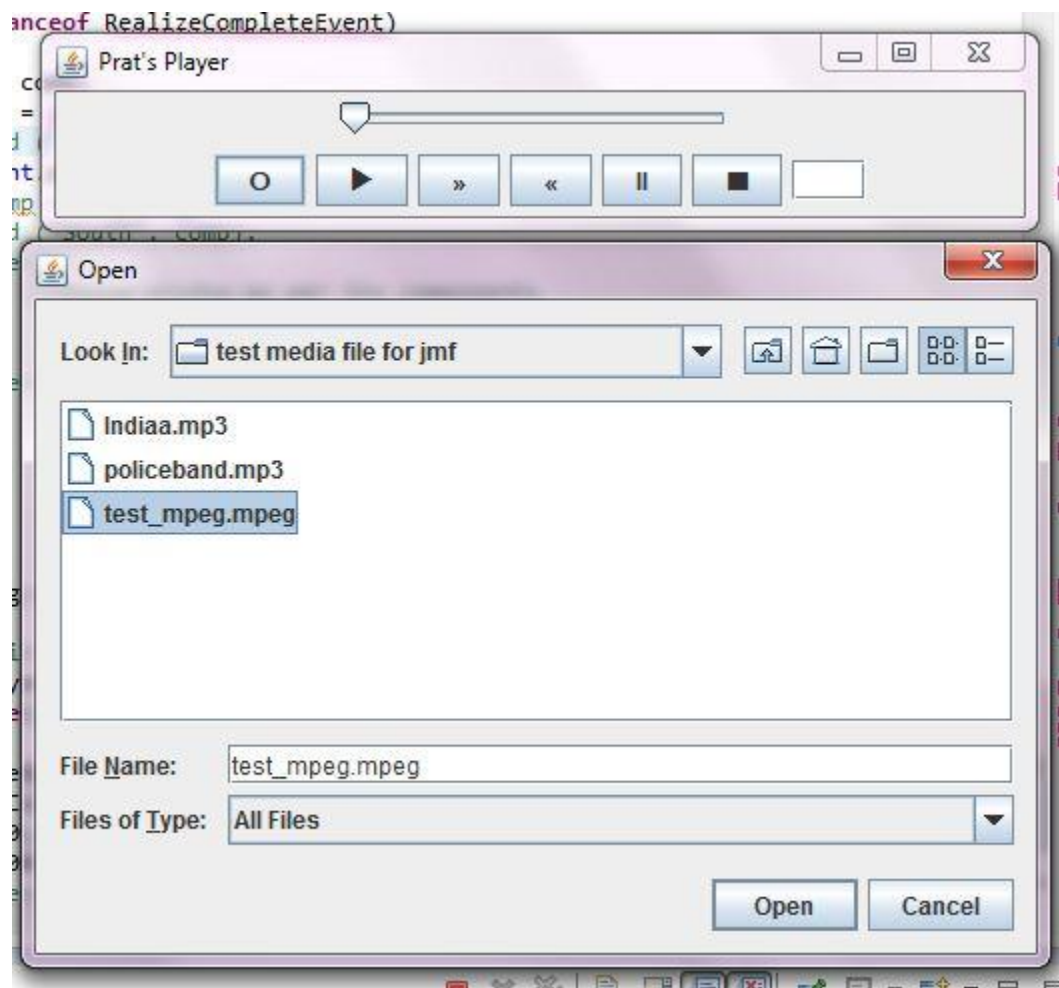


Prat's Player Documentation

Introduction- My player name is Prat's player. It is made in java swing and JMF. I designed my own GUI rather than starting from skeleton file provided to us. Player can play audio/video file supported by JMF. User can stop, pause, fast forward media. It also shows a slider and duration of file being played.

Open File- JFileChooser helps choose a file. We set the selectionmode as filesonly as we want to select the audio/video file to be played. Showopendialog opens a dialog box for choosing a file. getselectedfile returns the selected file. We don't create any player when we open a file.



Play - manager.createplayer creates a player but it is in unrealized state. I am implementing controllerListener interface and using addcontrollerlistener method to register its interest in controllerevents. ControllerUpdate method is called whenever an event is generated by a controller. So,

controllerupdate is called when we have transitionevent, realizecompleteevent, prefetchcompleteevent, startevent, stopevent,etc.

I am using player.start method which changes the state of the player from any state. So after calling player.start the state of the player changes from unrealized => realizing=> realized => prefetching=> prefetched => started (controllerupdate method gets called again and again during events) and our media file begins to play. Player goes into prefetched state when media playback is complete

In controllerupdate method, I am checking if player is realized. I get the visualcomponent of the player and add it to the contentpane. pack method resizes the window according to the size of its components.



Stop - when we call player.stop, state of player changes from started to prefetched state.
player.deallocate changes the state to realized state

Pause- To implement pause functionality, I am storing the duration of media already played in a variable and then calling player.stop method which changes the state of the player to prefetched state. So when we again hit play button, I use setmediatime and then player.start which changes the state of the player to started and our media begins to play from the position it was paused.

Fast Forward- I am just calling the setrate method and making sure that the player rate increases by .5 whenever we hit the fast forward button.

Slider and Passing time- I have created a timer which calls a displayrate method after every one second. We can get the total duration of media by using player.getduration.getseconds method. We can also get the amount of time already played using getmediananoseconds. This helped me to set the position of slider and value of a textfield by using basic arithmetic operations.



I used `player.getstate` method to check state of player at different points in my code. `Getstate` method returns an int which is as follows

Unrealized = 100;

Realizing = 200;

Realized = 300;

Prefetching = 400;

Prefetched = 500;

Started = 600;