## 4.4 Applet Life Cycle

Java applet inherits features from the class Applet. Thus, whenever an applet is created, it undergoes a series of changes from initialization to destruction. Various stages of an applet life cycle are shown in the figure :

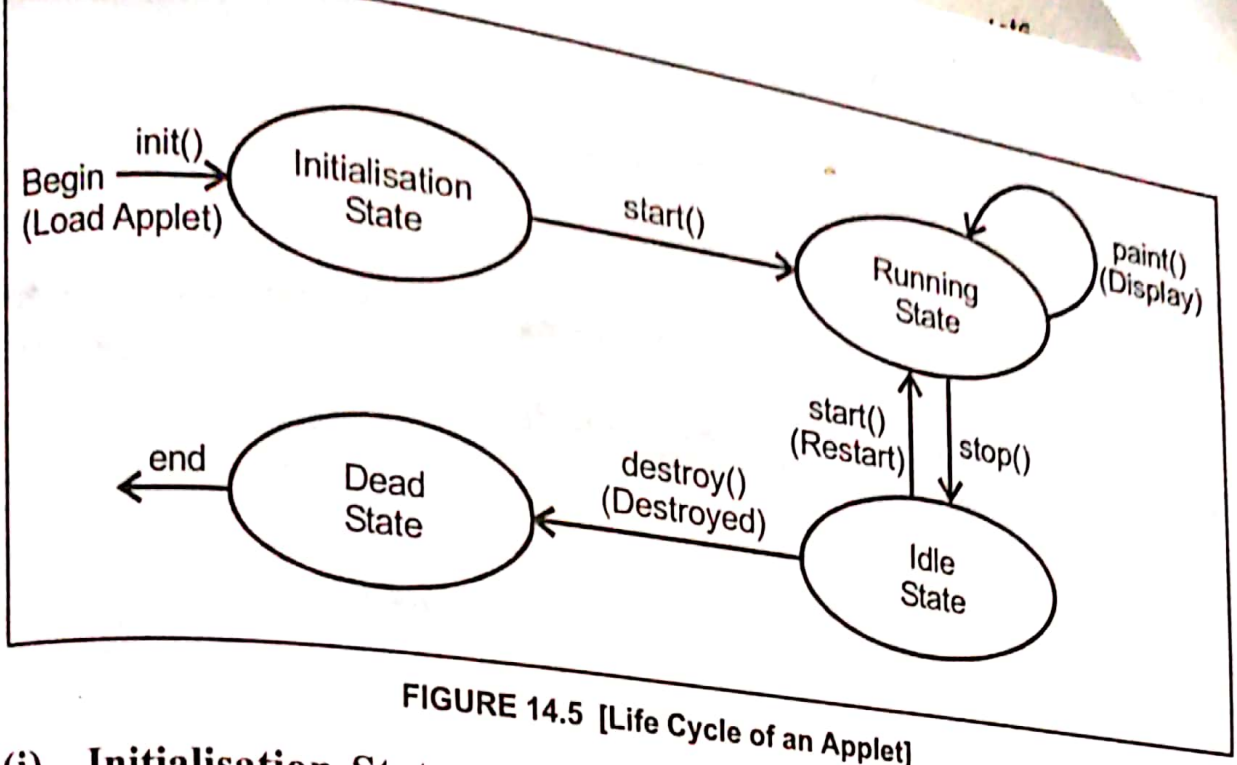


FIGURE 14.5 [Life Cycle of an Applet]

### (i) Initialisation State

When a new applet is born or created, it is activated by calling **init()** method. At this stage, new objects to the applet are created, initial values are set, images are loaded and the colors of the images are set. An applet is initialized only once in its lifetime. It's general form is:

```
public void init( )
{
        //Action to be performed
}
```

### (ii) Running State

An applet achieves the running state when the system calls the **start()** method. This occurs as soon as the applet is initialized. An applet may also start when it is in idle state. At that time, the start() method is overridden. It's general form is:

```
public void start( )
{
        //Action to be performed
}
```

### (iii) Idle or Stopped State

An applet comes in idle state when its execution has been stopped either implicitly or explicitly. An applet is implicitly stopped when we leave the page containing the currently running applet. An applet is explicitly stopped when we call **stop()** method to stop its execution. It's general form is :

```
public void stop( )
{
        //Action to be performed
}
```

### (iv) Dead State

An applet is in dead state when it has been removed from the memory. This can be done by using destroy() method. This stage is also occur only once in the applet's life cycle. It's general form is :

```
public void destroy( )
{
        //Action to be performed
}
```

### (v) Display State (Running state)

Apart from the above stages, Java applet also possess paint( ) method. This method helps in drawing, writing and creating colored backgrounds of the applet. It takes an argument of the graphics class. To use the graphics, it imports the package **java.awt.Graphics**. It's general form is :

```
public void paint(Graphics g )
{
        //Display Statements
}
```

}

...y method

Figure 7.7 summarizes the work mechanism of an Applet:

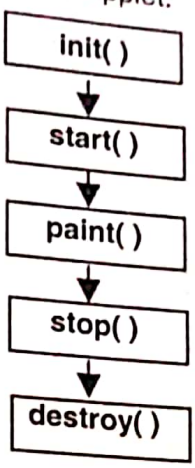


Fig. 7.7
Life cycle of an applet

Consider the following example, which illustrates the life cycle of an applet. This is the source code for an applet called Simple.

```
import java.applet.Applet;
import java.awt.Graphics;
public class Simple extends Applet
{
   String s="The";
   public void init()
   {
         s = s+"New";
         System.out.println(s);

   }
   public void start()
   {
         s=s+"Applet";
         System.out.println(s);
   }
   public void stop()
   {
         s=s+"stops";
         System.out.println("Stops");
   }
   public void destroy()
   {
         s=s+"here";
      System.out.println("Here");
   }
   public void paint(Graphics g)
```

```
    {
        //Draw a rectangle around the applet's display area
        g.drawRect(25, 25, 100,100);

        //Draw the current string inside the rectangle
        g.drawString(s, 25, 40);
    }
}
```

Save this file as **Simple.java**. Create an HTML file called **Simple.html** as follows:

```
<html>
<body>
<applet code=Simple width=200 height=200>
</applet>
</body>
</html>
```