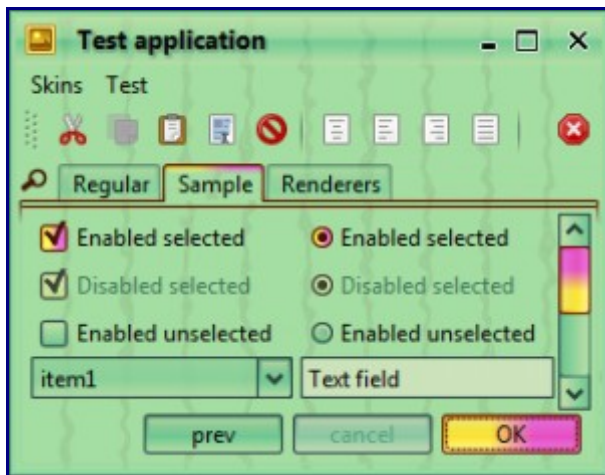


A quienes han buscado la manera de cambiar la apariencia la interfaz gráfica de una aplicación hecha con swing de java, es decir, cambiarle el skin/tema a su aplicación, dejo el siguiente tutorial que hace uso de la libreria “Substance”, un proyecto relativamente famoso y muy utilizado para el cambio de “Look and feel” pero con poca documentación en Español, es así que nos dimos un poco de tiempo para entender como aplicar algunas de sus funciones y publicárselos de la manera más sencilla con el afán de que les sea de utilidad.



Ejemplo de uso de Substance

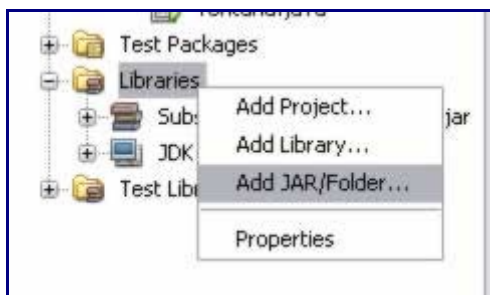
Las versiones con las que se trabaja son: Substance 4.3, JDK 6 update 10 y Netbeans 6.5

1. Descargar Libreria y agregar al proyecto.

El primer paso es descargar la libreria del Substance 4.3 de <https://substance.dev.java.net/servlets/ProjectDocumentList?folderID=9911&expandFolder=9911&folderID=9100> . Es un archivo .jar que pesa alrededor de 1.7Mb y que debemos de agregar a nuestro proyecto.

Name	Status	Modified by	Size	Reservations	Description
Substance 4.3 distribution	Stable	hmlcool on Friday, October 3, 2008 at 6:25:23 PM	15.27 mb		Contains the complete sources, documentation, and the build script
Substance 4.3 library	Stable	hmlcool on Friday, October 3, 2008 at 6:25:17 PM	1.75 mb		Runtime library (full functionality)
Substance 4.3 library lite	Stable	hmlcool on Friday, October 3, 2008 at 6:25:20 PM	1.24 mb		Runtime library with basic functionality (no color localization and additional widgets)

Descargamos el archivo .jar de la pagina del proyecto: <https://substance.dev.java.net>



En Netbeans agregamos el archivo a nuestro proyecto como libreria

2. Aplicar los Skins de Substance

Existen varias formas de aplicar un Skin de substance. La manera mas sencilla de aplicarlo es desde la clase main antes de hacer la llamada a cualquier JFrame con las 2 siguientes sentencias:

JFrame.setDefaultLookAndFeelDecorated(true); //que nos permite dejar a Substance la decoracion (por asi decirlo)

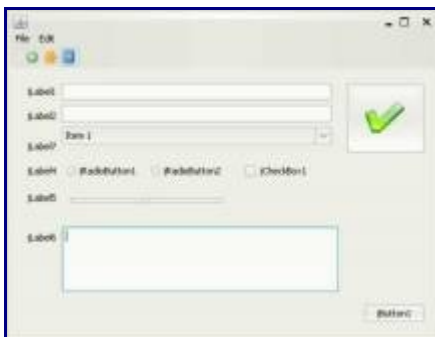
SubstanceLookAndFeel.setSkin("org.jvnet.substance.skin.CremeSkin"); // Setencia que aplica el skin Creme de Substance

```
package com.mprado;  
  
import javax.swing.JFrame;  
import org.jvnet.substance.SubstanceLookAndFeel;  
  
public class Main {  
    public static void main(String[] args) {  
        JFrame.setDefaultLookAndFeelDecorated(true);  
        SubstanceLookAndFeel.setSkin("org.jvnet.substance.skin.CremeSkin");  
        Ventana ventana = new Ventana();  
    }  
}
```

La manera mas facil de aplicar Substance



La clase Ventana que es un JFrame con diversos componentes. Con el Look and feel por default en java: Metal

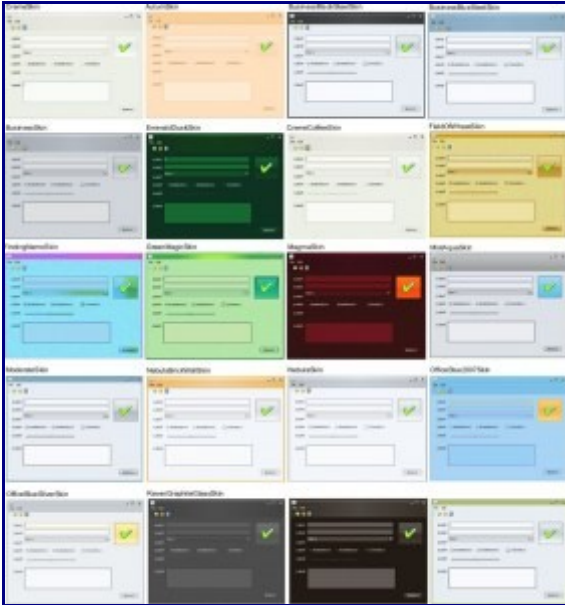


Ventana con el skin "CremeSkin" de Substance

La lista de Skins para Substance 4.3 esta dada a continuaci3n, as3 solo se debe cambiar el parametro a la funcion *SubstanceLookAndFeel.setSkin()* para aplicar el skin deseado:

- org.jvnet.substance.skin.AutumnSkin
- org.jvnet.substance.skin.BusinessBlackSteelSkin
- org.jvnet.substance.skin.BusinessBlueSteelSkin
- org.jvnet.substance.skin.BusinessSkin
- org.jvnet.substance.skin.CremeCoffeeSkin
- org.jvnet.substance.skin.CremeSkin
- org.jvnet.substance.skin.EmeraldDuskSkin
- org.jvnet.substance.skin.FieldOfWheatSkin
- org.jvnet.substance.skin.FindingNemoSkin
- org.jvnet.substance.skin.GreenMagicSkin
- org.jvnet.substance.skin.MagmaSkin
- org.jvnet.substance.skin.MangoSkin

- org.jvnet.substance.skin.MistAquaSkin
- org.jvnet.substance.skin.ModerateSkin
- org.jvnet.substance.skin.NebulaBrickWallSkin
- org.jvnet.substance.skin.NebulaSkin
- org.jvnet.substance.skin.OfficeBlue2007Skin
- org.jvnet.substance.skin.OfficeSilver2007Skin
- org.jvnet.substance.skin.RavenGraphiteGlassSkin
- org.jvnet.substance.skin.RavenGraphiteSkin
- org.jvnet.substance.skin.RavenSkin
- org.jvnet.substance.skin.SaharaSkin



Apariencia de los Skins de Substance

3. Cambiar el Tema

El cambio de tema es como cambiar el skin pero Substance los maneja en otro paquete el cual nos proporciona otra lista de themes con una muy buena variedad de colores. La diferencia en su implementación es que va después de haber aplicado un skin o Look and feel y la sentencia que se debe aplicar es:

SubstanceLookAndFeel.setCurrentTheme("org.jvnet.substance.theme.SubstanceAquaTheme"); // Se aplica el tema Aqui de Substance

```
package substance;

import java.awt.*;
import org.jvnet.substance.SubstanceLookAndFeel;

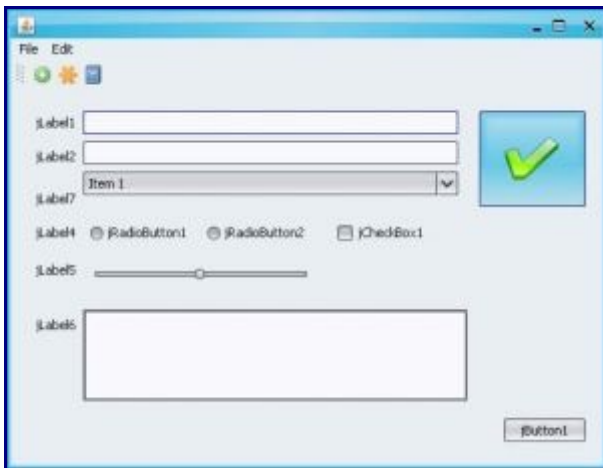
/**
 * Simple Abstracting
 */
public class Main {

    /**
     * Program will be covered like this:
     */
    public static void main(String[] args) {

        JFrame.setDefaultLookAndFeelDecorated(true);
        SubstanceLookAndFeel.setCurrentTheme(new SubstanceTheme());
        SubstanceLookAndFeel.setCurrentTheme("org.jvnet.substance.theme.SubstanceAquaTheme");

        Window window = new Window();
    }
}
```

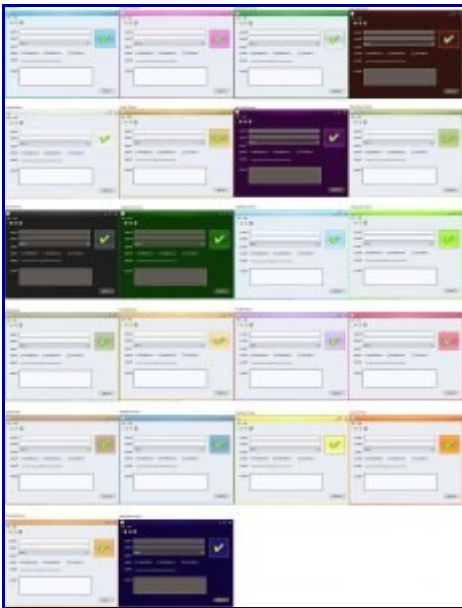
Es requisito haber aplicado primero un Skin pero la apariencia sera del theme declarado



Ejemplo de la aplicacion del theme Aqua de Substance

La lista de temas en Substance 4.3 es dada a continuación, de igual manera para cambiar de tema tan solo cambiar el String de parametro en la funcion *SubstanceLookAndFeel.setCurrentTheme()*:

- org.jvnet.substance.theme.SubstanceAquaTheme
- org.jvnet.substance.theme.SubstanceBarbyPinkTheme
- org.jvnet.substance.theme.SubstanceBottleGreenTheme
- org.jvnet.substance.theme.SubstanceBrownTheme
- org.jvnet.substance.theme.SubstanceCharcoalTheme
- org.jvnet.substance.theme.SubstanceCremeTheme
- org.jvnet.substance.theme.SubstanceDarkVioletTheme
- org.jvnet.substance.theme.SubstanceDesertSandTheme
- org.jvnet.substance.theme.SubstanceEbonyTheme
- org.jvnet.substance.theme.SubstanceJadeForestTheme
- org.jvnet.substance.theme.SubstanceLightAquaTheme
- org.jvnet.substance.theme.SubstanceLimeGreenTheme
- org.jvnet.substance.theme.SubstanceNegatedTheme
- org.jvnet.substance.theme.SubstanceOliveTheme
- org.jvnet.substance.theme.SubstanceOrangeTheme
- org.jvnet.substance.theme.SubstancePurpleTheme
- org.jvnet.substance.theme.SubstanceRaspberryTheme
- org.jvnet.substance.theme.SubstanceSaturatedTheme
- org.jvnet.substance.theme.SubstanceSepiaTheme
- org.jvnet.substance.theme.SubstanceSteelBlueTheme
- org.jvnet.substance.theme.SubstanceSunGlareTheme
- org.jvnet.substance.theme.SubstanceSunsetTheme
- org.jvnet.substance.theme.SubstanceTerracottaTheme



Apariencia de los themes de Substance

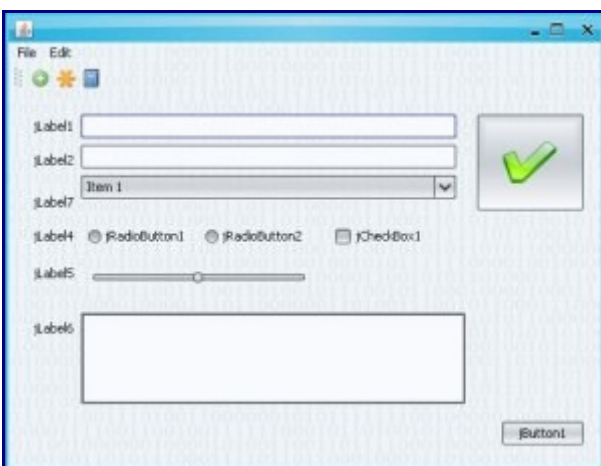
4. Cambiar el Watermark

En el Watermark se permite cambiar el fondo/Background de los componentes contenedores en un JFrame o similares, desde el uso de watermarks predeterminados en Substance hasta colocar una imagen y cambiar los valores de transparencia de ésta. La Sentencia para aplicarlo es con la funcion *setCurrentWatermark()* de la clase *SubstanceLookAndFeel*:

SubstanceLookAndFeel.setCurrentWatermark("org.jvnet.substance.watermark.SubstanceBinaryWatermark");//Ejemplo de aplicacion de un watermark de Substance



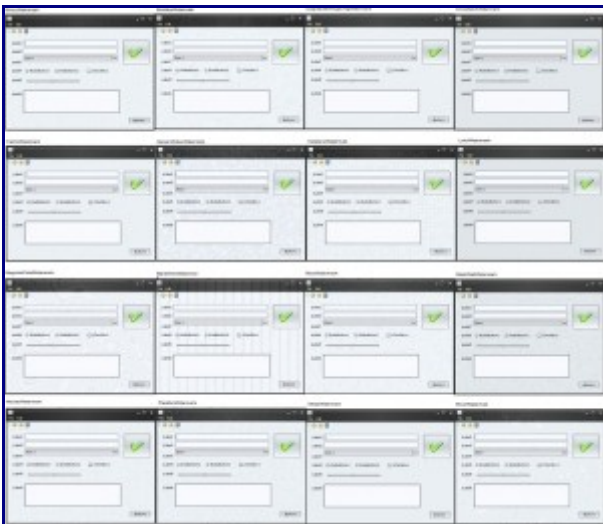
Implementacion de Watermark en main



Observa el fondo de la ventana compuesto por unos y ceros. Se aplico BinaryWatermark de Substance

La lista de Watermarks de Substance 4.3 es:

- org.jvnet.substance.watermark. SubstanceBubblesWatermark
- org.jvnet.substance.watermark. SubstanceBinaryWatermark
- org.jvnet.substance.watermark. SubstanceCopperplateEngravingWatermark
- org.jvnet.substance.watermark. SubstanceCrosshatchWatermark
- org.jvnet.substance.watermark. SubstanceFabricWatermark
- org.jvnet.substance.watermark. SubstanceGenericNoiseWatermark
- org.jvnet.substance.watermark. SubstanceImageWatermark
- org.jvnet.substance.watermark. SubstanceKatakanaWatermark
- org.jvnet.substance.watermark. SubstanceLatchWatermark
- org.jvnet.substance.watermark. SubstanceMagneticFieldWatermark
- org.jvnet.substance.watermark. SubstanceMarbleVeinWatermark
- org.jvnet.substance.watermark. SubstanceMazeWatermark
- org.jvnet.substance.watermark. SubstanceMetalWallWatermark
- org.jvnet.substance.watermark. SubstanceNoneWatermark
- org.jvnet.substance.watermark. SubstanceNullWatermark
- org.jvnet.substance.watermark. SubstancePlanktonWatermark
- org.jvnet.substance.watermark. SubstanceStripeWatermark
- org.jvnet.substance.watermark. SubstanceWoodWatermark



Apariencia de los Watermarks de Substance

4.1 Watermak de imagen

Una función interesante es proporcionar una imagen propia para colocarlo de watermark en la aplicación; para esto utilizaremos el *SubstanceImageWatermark* instaciándolo y proporcionando la ruta de la imagen como parámetro.

SubstanceLookAndFeel.setCurrentWatermark(new SubstanceImageWatermark("c:\\tuxx.jpg"));

```
package javax.swing;

import org.jdesktop.swingx.JXFrame;
import org.jdesktop.swingx.SubstanceLookAndFeel;
import org.jdesktop.swingx.SubstanceImageWatermark;

/**
 * Demostración de la interfaz de usuario
 */
public class Main {

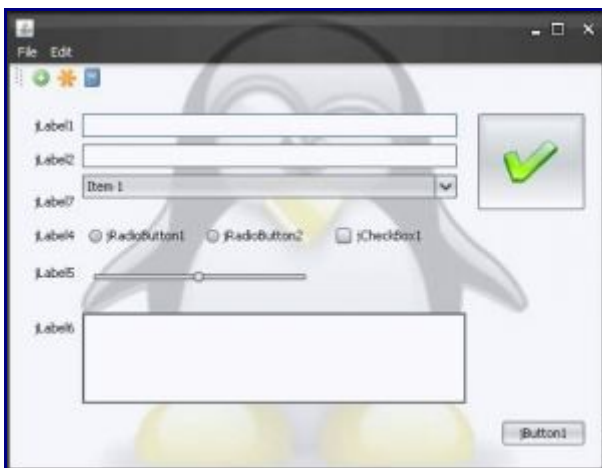
    /**
     * Método principal de la aplicación
     */
    public static void main(String[] args) {
        JXFrame.setDefaultLookAndFeelDecorated(true);
        SubstanceLookAndFeel.setCurrentWatermark(new SubstanceImageWatermark("c:\\tuxx.jpg"));
        //SubstanceLookAndFeel.setCurrentTheme(new org.jdesktop.swingx.theme.SubstanceLookAndFeelTheme());
        //SubstanceLookAndFeel.setCurrentWatermark(new SubstanceImageWatermark("c:\\tuxx.jpg"));

        Ventana principal = new Ventana();
    }
}
```

Se utiliza SubstanceImageWatermark con la ruta de la imagen como parámetro



Imagen a utilizar para aplicarla como watermark



Resultado de aplicar ImageWatermark

4.1.1 Opacidad de la imagen

Substance también permite cambiar la opacidad de la imagen del watermark através de la función *setImageWatermarkOpacity()* al cual se debe pasar el valor en flotante entre 0 y 1, siendo 1 la opacidad nula.

SubstanceLookAndFeel.setImageWatermarkOpacity(new Float(0.3));//valor aproximado de la opacidad por default de imageWatermark

[illegible]

implementacion de la opacidad de la imageWatermark en main



Opacidad de la imagen Watermark a un valor de 0.6



Opacidad de la ventana a un valor de 0.9

Opciones para Componentes.

Muchas opciones de Substance son para su aplicación a Componentes Swing específicos, aquí se verán algunas de ellas para Botones, Combobox, Tabpane y otras funciones que se pueden aplicarse a varios.

Botones.

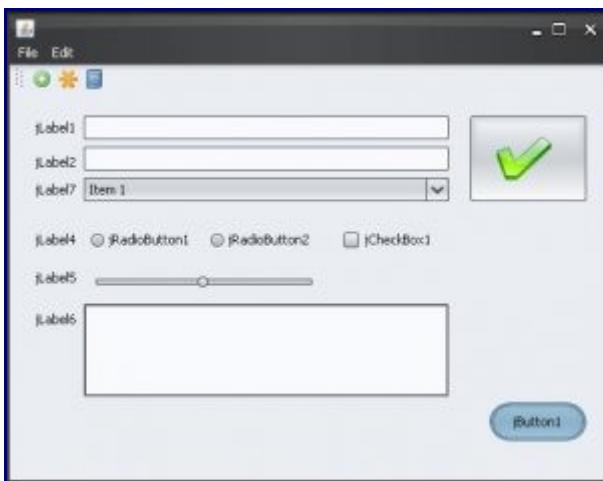
5. Redondear los botones.

La propiedad de cliente *BUTTON_SHAPER_PROPERTY* de Substance permite cambiar la clásica forma rectangular del botón a una más redondeada que puede hacerlo resaltar más o hacerlo mas amigable, por ejemplo. La implementación se realiza desde la clase del JFrame (en mi caso la clase Ventana) en el constructor. Se debe pasar un objeto Shape que es la forma que tomará el botón y la cual proporcionara Substance: *StandardButtonShaper* que es un Shape que da redondeo al botón.

```
this.jButton1.putClientProperty( SubstanceLookAndFeel.BUTTON_SHAPER_PROPERTY, new  
StandardButtonShaper());
```



Se aplica una propiedad de cliente al boton que deseamos cambiar su forma



Aplicacion de la forma StandardButtonShaper de Substance al jButton1

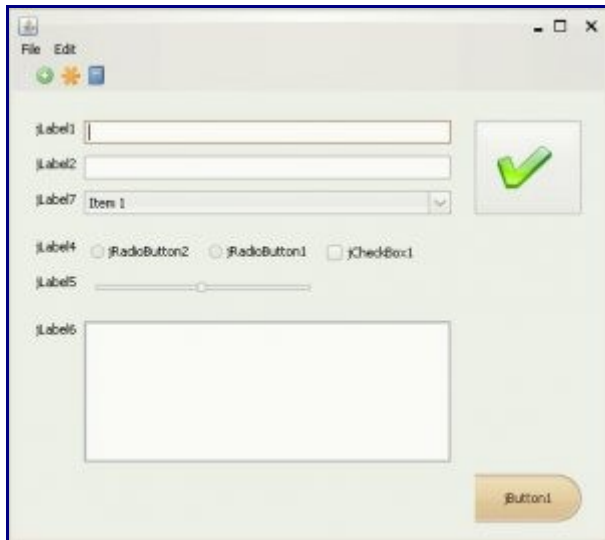
6. Lado de abertura de un botón

Permite que un cualquier lado de un boton esté abierto, es decir, acotado en espacio y borde de lado seleccionado: *bottom*, *left*, *up*, *right*. La propiedad *BUTTON_SIDE_PROPERTY* acota el area de lado y *BUTTON_OPEN_SIDE_PROPERTY* elimina el borde de tal costado.

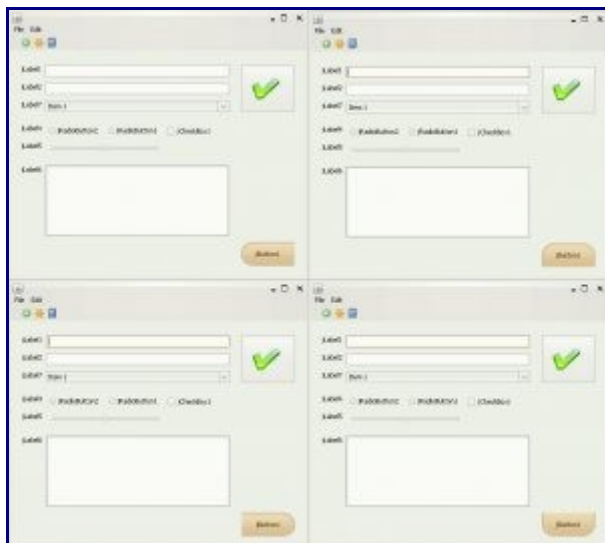
```
this.jButton1.putClientProperty( SubstanceLookAndFeel.BUTTON_SIDE_PROPERTY,  
SubstanceConstants.Side.LEFT);  
this.jButton1.putClientProperty( SubstanceLookAndFeel.BUTTON_OPEN_SIDE_PROPERTY,  
SubstanceConstants.Side.LEFT);
```

[illegible]

Los parametros Substance los proporciona con SubstanceConstants



Aplicacion de la propiedad a JButton1 con el cremeCoffee como skin



Los 4 tipos de aberturas de lado de un boton