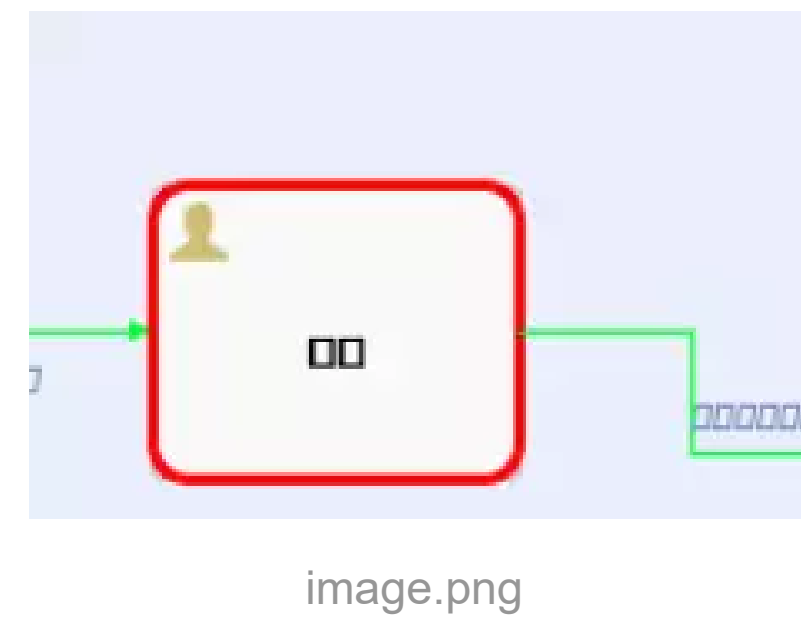


centos7下，flowable流程跟踪图出现乱码

1. 默认最简单的安装centos7，一般中文字体都是不全的，这时如果访问flowable的流程跟踪图会出现



2. 主要原理是flowable是在服务器端，用awt组件的Graphics2D画出，再通过图片返回到浏览器的，以下是flowable的源码，对应的类org.flowable.image.impl.DefaultProcessDiagramCanvas，这个类的模块是flowable-image-generator

```
public void initialize(String imageType) {
    if ("png".equalsIgnoreCase(imageType)) {
        this.processDiagram = new BufferedImage(canvasWidth, canvasHeight, BufferedImage.TYPE_INT_ARGB);
    } else {
        this.processDiagram = new BufferedImage(canvasWidth, canvasHeight, BufferedImage.TYPE_INT_RGB);
    }

    this.g = processDiagram.createGraphics();
    if (!"png".equalsIgnoreCase(imageType)) {
        this.g.setBackground(new Color(0, 0, 0, 0));
        this.g.clearRect(0, 0, canvasWidth, canvasHeight);
    }

    g.setRenderingHint(RenderingHints.KEY_ANTIALIASING, RenderingHints.VALUE_ANTIALIAS_ON);
    g.setPaint(Color.black);

    Font font = new Font(activityFontName, Font.BOLD, FONT_SIZE);
    g.setFont(font);
    this.fontMetrics = g.getFontMetrics();

    LABEL_FONT = new Font(labelFontName, Font.ITALIC, size: 10);
    ANNOTATION_FONT = new Font(annotationFontName, Font.PLAIN, FONT_SIZE);
}
```

3. 解决的办法，第一种方法是安装好centos7的中文字体；第二种jvm启动加载中文字体，和指定flowable绘图的字体。

4. 以下是在jvm下给flowable的流程跟踪图指定字体
默认情况下，flowable是和springboot部署的，所有我这边用到的是spring配置

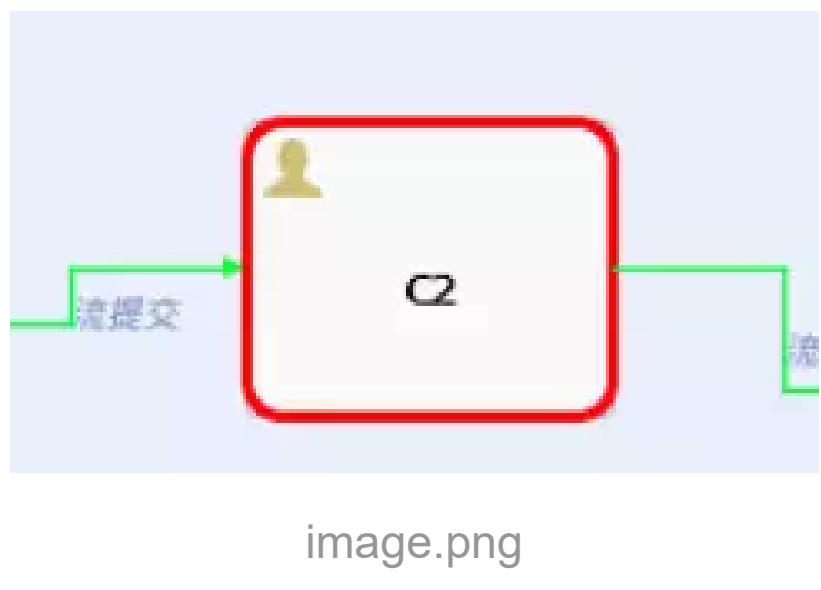
```
@Configuration
public class MyFlowableConfig implements EngineConfigurationConfigurer<SpringProcessEngine> {

    @Override
    public void configure(SpringProcessEngineConfiguration engineConfiguration) {
        engineConfiguration.setActivityFontName("WenQuanYi Micro Hei");
        engineConfiguration.setLabelFontName("WenQuanYi Micro Hei");
        engineConfiguration.setAnnotationFontName("WenQuanYi Micro Hei");
    }
}
```

5. 还有在启动spring boot的web应用时，还需要加载自己指定的字体，我这边用到的是linuxmint下的文泉驿微米黑字体/usr/share/fonts/truetype/wqy/wqy-microhei.ttc

```
@Component
public class MyLoadFont4Flowable {
    @PostConstruct
    protected void init() {
        GraphicsEnvironment ge = GraphicsEnvironment.getLocalGraphicsEnvironment();
        ge.registerFont("/你的字体路径/wqy-microhei.ttc");
    }
}
```

6. 当启动应用再看时



7. 如果在window下部署flowable出现显示跟踪图不正确时，只要设置为宋体就可以了

```
@Configuration
public class MyFlowableConfig implements EngineConfigurationConfigurer<SpringProcessEngine> {

    @Override
    public void configure(SpringProcessEngineConfiguration engineConfiguration) {
        engineConfiguration.setActivityFontName("宋体");
        engineConfiguration.setLabelFontName("宋体");
        engineConfiguration.setAnnotationFontName("宋体");
    }
}
```

主要是flowable默认的字体系是Arial
org.flowable.image.impl.DefaultProcessDiagramCanvas指定字体的源码如下

```
protected static BufferedImage CAMEL_TASK_IMAGE;
protected static BufferedImage HTTP_TASK_IMAGE;

protected static BufferedImage TIMER_IMAGE;
protected static BufferedImage COMPENSATE_THROW_IMAGE;
protected static BufferedImage COMPENSATE_CATCH_IMAGE;
protected static BufferedImage ERROR_THROW_IMAGE;
protected static BufferedImage ERROR_CATCH_IMAGE;
protected static BufferedImage MESSAGE_THROW_IMAGE;
protected static BufferedImage MESSAGE_CATCH_IMAGE;
protected static BufferedImage SIGNAL_CATCH_IMAGE;
protected static BufferedImage SIGNAL_THROW_IMAGE;

protected int canvasWidth = 1;
protected int canvasHeight = 1;
protected int minX = 1;
protected int minY = 1;
protected BufferedImage processDiagram;
protected Graphics2D g;
protected FontMetrics fontMetrics;
protected boolean closed;
protected ClassLoader customClassLoader;
protected String activityFontName = "Arial";
protected String labelFontName = "Arial";
protected String annotationFontName = "Arial";

/**
 * Creates an empty canvas with given width and height.
 *
 * Allows to specify minimal boundaries on the left and upper side of the canvas. This is useful if you want to
 * It's also possible to pass a specific font name and a class loader for the icon images.
 */
```