3.3.1.7 Graphics state

One may wonder how we'd be getting color and transformation values back to its original state if we would need to. So far we built our samples in a way that they didn't require any rollbacks, being fairly straightforward. But what if we would have to draw multiple objects on PDF page each having its own transforms and other settings? As you remember from previous articles, drawing commands are being added in sequences and commands that change the state of drawing context often affect subsequent drawings.

In order to avoid this you may perform drawings as isolated sequences and do a quick rollback when it's done. Here the concept of graphics state arises. PDF specification defines it in section *8.4 "Graphics State"* . In a few words, a graphics state is an internal object that is being used to maintain the state of current graphics context: its colors, clipping, transformations etc. These objects can be enclosed into each other thus making inner state saving possible. Saving and restoring of the *current* graphics state requires a command to be added to the drawing commands sequence. Below is the code that shows it in action.

```
// create clipped content object for drawing
ClippedContent page = new ClippedContent();

// save current state
page.SaveGraphicsState();

… apply transforms, set colors, draw paths

// restore initial state
page.RestoreGraphicsState();

…continue drawing using restored state
```

The sample above shows how to work with *current* graphics state, but it's also possible to set graphics states that you have created in advance and designed for multiple usages. E.g. a graphics state describing some alpha blending operations that you'd like to share and use for many independent drawings.

There is a special type called `GraphicsState` that can be used to achieve this goal, it's located under `Apitron.PDF.Kit.FixedLayout.Resources.GraphicsStates` namespace. These state objects can be set using their identifiers and should be registered as resources first in order to be used.