1 Embeded Graphics

In this chapter we will introduce the interface between *ConTEXt* and *MetaPost* and demonstrate how the definitions of the graphics can be embedded in the document source.

1.1 External graphics

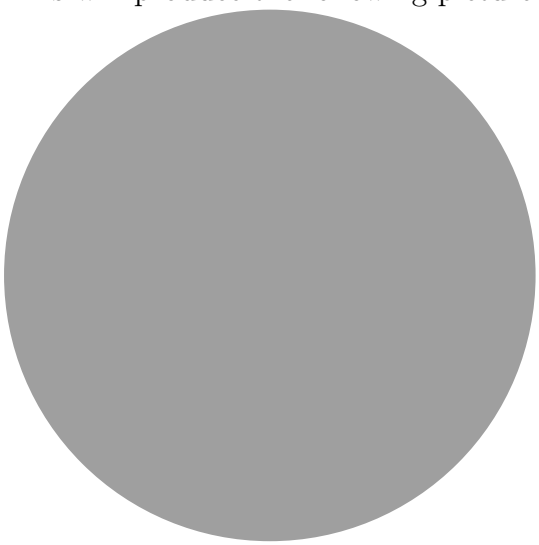
1.2 Integrated graphics

1.2.1 Primitive method

An integrated graphic is defined in the document source or in a style definition file. The most primitive way of doing this is just inserting the code:

```
\startMPcode
fill fullcircle scaled 200pt withcolor .625white ;
\stopMPcode
```

This will produce the following picture:



Such a graphic is used once at the spot where it is defined.

1.2.2 Usable graphics

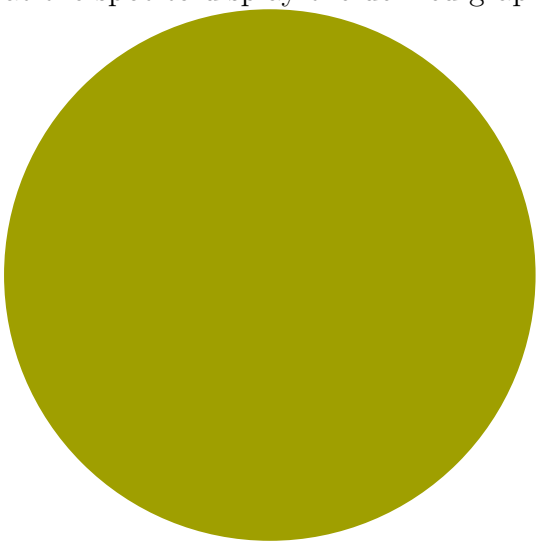
A *usable* graphic is calculated anew each time it is used. An example of defining a usable graphic:

```
\startuseMPgraphic{name}
fill fullcircle scaled 200pt withcolor .625yellow ;
\stopuseMPgraphic
```

To use it, we place the the command

```
\useMPgraphic{name}
```

at the spot to display the defined graphic.



Note that this graphic is calculated each time it is placed, which can be time consuming.

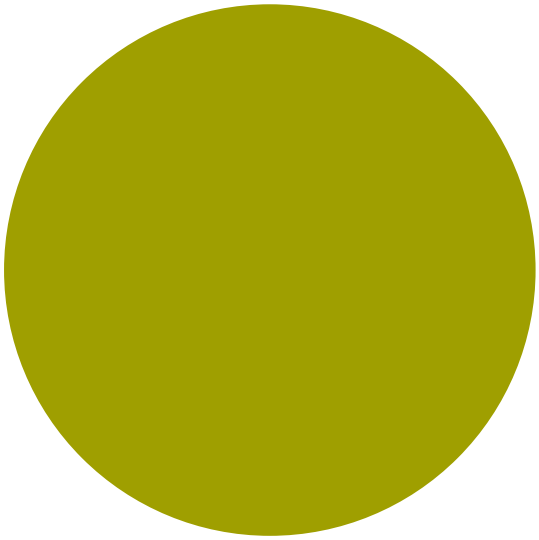
1.2.3 Reusable graphics

For graphics that don't change, *ConTEXt* provides *reusable* graphics:

```
\startreusableMPgraphic{name}
fill fullcircle scaled 200pt withcolor .625yellow;
\stopreusableMPgraphic
```

This definition is accompanied by:

```
\reuseMPgraphic{name}
```



Imagine that we use a graphic as a background for a button. We can create a unique and reusable graphic by saying:

```
\def\MyGraphic{
  \startreusableMPgraphic{name:\overlaywidth:\overlayheight}
  path p ; p := unitsquare
  xscaled OverlayWidth yscaled OverlayHeight ;
  fill p withcolor .625yellow ;
  draw p withcolor .625red ;
  \stopreusableMPgraphic
  \reuseMPgraphic{name:\overlaywidth:\overlayheight}
}
```

Notice the use of `OverlayWidth` and `OverlayHeight`. These variables are set for each call to *MetaPost*. After this we can say:

```
\defineoverlay[my graphic][\MyGraphic]
\button[background=my graphic,frame=off]{Go Home}[firstpage]
```

Say that we have a 30pt by 20pt button, then the identifier will be name:30pt:20pt. Different dimensions will lead to other identifiers, so this sort of makes the graphics unique.
Imagine what happens when we add some buttons to an interactive document without taking care of this side effect. All the frames would look different. Consider the following example.

```
\startuniqueMPgraphic{right or wrong}
  pickup pencircle scaled .075 ;
```

```
  fill unitsquare withcolor .8white ;
  draw unitsquare withcolor .625red ;
  currentpicture := currentpicture
    xscaled OverlayWidth yscaled OverlayHeight ;
\stopuniqueMPgraphic
```

Let's define this graphic as a background to some buttons.

```
\defineoverlay[button][\uniqueMPgraphic{right or wrong}]
\setupbuttons[background=button,frame=off]

\hbox
  {\button {previous}          [previouspage]\quad
   \button {next}              [nextpage]\quad
   \button {index}             [index]\quad
   \button {table of contents} [content]}
```

The buttons will look like:

previous next index table of contents

Compare these with:

previous next index table of contents

Here the graphic was defined as:

```
\startuniqueMPgraphic{wrong or right}
  pickup pencircle scaled 3pt ;
  path p ; p := unitsquare
    xscaled OverlayWidth yscaled OverlayHeight ;
  fill p withcolor .8white ;
  draw p withcolor .625red ;
\stopuniqueMPgraphic
```