

HOWTO: Use an ImageSource with an AnnotatePrintDocument or ImagePrintDocument

DotImage provides two very useful classes called ImagePrintDocument and AnnotatePrintDocument which extend the built in .NET PrintDocument. However, neither of these classes take an ImageSource as a constructor. Instead, they have overloads for ImageCollection and AtalaImage[], both of which are very memory inefficient.

Here is how to simulate the function of ImagePrintDocument pDoc = new ImagePrintDocument(ImageSource);

```
ImageSource _imgsrc; public void OpenDocument(string fileName) {
    _imgsrc= new FileSystemImageSource(fileName, true); } public void
DoPrint() { ImagePrintDocument pDoc = new ImagePrintDocument();
// This is where we tell it how to acquire the next image
doc.GetImage += new PrintImageEventHandler(doc_GetImage); // This
is where to tell it to release each page once the page is done
doc.AfterPrintPage += new
PrintImageEventHandler(doc_AfterPrintPage); // get as much of the
image on the page as possible doc.ScaleMode =
PrintScaleMode.FitToEdges; doc.Center = true; // Ask the user to
select the printer using (PrintDialog printDialog = new
PrintDialog()) { printDialog.Document = doc; if
(printDialog.ShowDialog() == DialogResult.OK) { // Execute the
print job doc.Print(); } } } /// /// After a page has been
printed, we need to tell the /// FileSystemImageSource to release
the image /// /// void doc_AfterPrintPage(object sender,
PrintImageEventArgs e) { _imgsrc.Release(e.Image); } /// /// when
the PrintDocument calls for an image, we will use the ///
FileSystemImageSource to get only ONE PAGE. /// /// void
doc_GetImage(object sender, PrintImageEventArgs e) { if
(_imgsrc.HasMoreImages()) { e.Image = _imgsrc.AcquireNext();
e.HasMorePages = _imgsrc.HasMoreImages(); }
```

This is virtually the same thing that you need to do for printing with Annotations using an AnnotatePrintDocument... just need a couple of additions.

```
ImageSource _imgsrc; public void OpenDocument(string fileName) {
    _imgsrc= new FileSystemImageSource(fileName, true); } public void
DoPrint(AnnotationController annoController) { // Since there's
no constructor that takes an ImageSource, just make an empty one
AnnotatePrintDocument pDoc = new AnnotatePrintDocument(); // Set
the annotationcontroller //(which we assume you have loaded with
the annotations that match the file you're printing)
pDoc.Annotations = annoController; // This is where we tell it
how to acquire the next image doc.GetImage += new
PrintImageEventHandler(doc_GetImage); // This is where to tell it
```

HOWTO: Use an ImageSource with an AnnotatePrintDocument or ImagePrintDocument

```
to release each page once the page is done doc.AfterPrintPage +=
new PrintImageEventHandler(doc_AfterPrintPage); // get as much of
the image on the page as possible doc.ScaleMode =
PrintScaleMode.FitToEdges; doc.Center = true; // If the Units
(annotation units to be precise) are not set, annotations will
not print doc.Units = AnnotationUnit.Pixel; // Ask the user to
select the printer using (PrintDialog printDialog = new
PrintDialog()) { printDialog.Document = doc; if
(printDialog.ShowDialog() == DialogResult.OK) { // Execute the
print job doc.Print(); } } } // ... the handlers for doc_GetImage
and doc_AfterPrintPage don't change at all ...}
```

Original Article:

Q10342 - HOWTO: Use an ImageSource with an AnnotatePrintDocument or
ImagePrintDocument

Atalasoft Knowledge Base

<https://www.atalasoft.com/KB2/KB/50144/HOWTO-Use-an-ImageSource-with-an-Ann...>