

HOWTO: Replace a single page of a PDF document

A relatively common situation for our customers to run across is that they only need to alter individual pages of a PDF document.

There are several reasons to replace individual pages within the original PDF instead of editing the whole document. Among them are preserving vector formatting on pages you don't intend to edit, memory constraints of working with large PDF documents and reducing run-time from having to open and then re-encode all the pages you aren't actively modifying.

This method will take a filepath to the original PDF (although it can be altered to take a Stream easily), the page number you intend to replace (zero-indexed) and the AtalaImage containing your altered page. In this demonstration we save back to the file system after appending "_tmp.pdf" to the original name and return a string with your newly created file. Again, it's a simple matter to modify it to use any Stream implementation to save the updated PDF.

[C#]

```
private string replacePdfPage(string originalPdf, int page, AtalaImage replaceImage) { /* Save
our altered AtalaImage to a MemoryStream. */ MemoryStream ms = new MemoryStream();
replaceImage.Save(ms, new PdfEncoder(), null); /* Make a PdfDocument in memory with the
altered AtalaImage */ PdfDocument newDoc = new PdfDocument(ms); /* Get our original
PdfDocument. Note: You can also pass this a FileStream or MemoryStream directly if you have
it available. */ PdfDocument pDoc = new PdfDocument(originalPdf); /* Remove page (x) and
insert our new page in place. */ pDoc.Pages.RemoveAt(page); pDoc.Pages.Insert(page,
newDoc.Pages[0]); /* Note: We've hard coded Pages[0] here as newDoc will only have the single
altered page. */ pDoc.Save(originalPdf + "_tmp.pdf"); newDoc.Close(); ms.Close();
ms.Dispose(); return originalPdf + "_tmp.pdf"; }
```

[VB.NET]

```
private Function replacePdfPage(ByVal originalPdf As String, ByVal page As Integer, ByVal
replaceImage As AtalaImage) As String 'Save our altered AtalaImage to a MemoryStream. Dim ms
As New MemoryStream() replaceImage.Save(ms, New PdfEncoder(), Nothing) 'Make a PdfDocument in
memory with the altered AtalaImage Dim newDoc As New PdfDocument(ms) 'Get our original
PdfDocument. Note: You can also pass this a 'FileStream or MemoryStream directly if you have
it available. Dim pDoc As New PdfDocument(originalPdf) 'Remove page (x) and insert our new
page in place. pDoc.Pages.RemoveAt(page) pDoc.Pages.Insert(page, newDoc.Pages(0)) 'Note:
We've hard coded Pages(0) here as newDoc will 'only have the single altered page.
newDoc.Close() ms.Close() ms.Dispose() pDoc.Save(originalPdf & "_tmp.pdf") Return originalPdf
& "_tmp.pdf" End Function
```

HOWTO: Replace a single page of a PDF document

Original Article:

Q10343 - HOWTO: Replace a single page of a PDF document

Atalasoft Knowledge Base

<https://www.atalasoft.com/kb2/KB/50143/HOWTO-Replace-a-single-page-of-a-PDF...>