

# **PDF/A validation - overcoming limitations of validation tools**

**Written by Apitron Documentation Team**

## Introduction

PDF/A is a perfect alternative when it comes to archiving and saving documents for later use. The format guarantees that the document can be read years after creation because all resources needed to process the document are embedded into the file. Sometimes PDF/A is set as a requirement for saving documents with digital signatures, e.g. contracts, official papers and so on.

There are plenty of tools on the market that claim that they can produce PDF/A documents, and the only way to check if the tool fulfills this condition is to check it using a PDF/A validation tool.

The most popular and reliable tool from our point of view is Adobe Acrobat Professional – a paid professional version of the well-known Adobe Reader. It allows you to validate the document against many conditions including PDF/A compatibility using built-in *Preflight* tool. As Adobe is the author of PDF standard it knows all inside and out of the PDF/A as well.

There are other PDF/A validation tools produced by various software companies, but sometimes their results differ from Adobe Acrobat Professional due to double interpretation of the PDF-A specification.

We use Adobe as a gold standard and [Apitron PDF Kit for .NET](#) product produces files 100% verifiable by Adobe Acrobat Professional. If you use the same toolchain you don't have to worry, as this post describes possible warnings produced by other tools, and custom settings needed to avoid them.

One of the possible warnings issued is – “the file contains cross reference streams”, it's related to internal storage format of objects to IDs mapping in PDF document. PDF versions prior to 1.5 (released in 2003) used cross reference tables instead of cross reference stream objects. The advantages of using streams over tables are:

- A more compact representation of cross-reference information
- The ability to access compressed objects that are stored in object streams (see 7.5.7, "Object Streams" section of the specification) and to allow new cross-reference entry types to be added in the future

Current PDF version is 1.7 (updated 2011), so it's a pretty old feature and PDF/A (released in 2005) don't forbid the use of such objects. To fix the cross-reference stream warning for those who need this we introduced the new setting for the PDF export API. The code sample can be found in the next section.

## The code

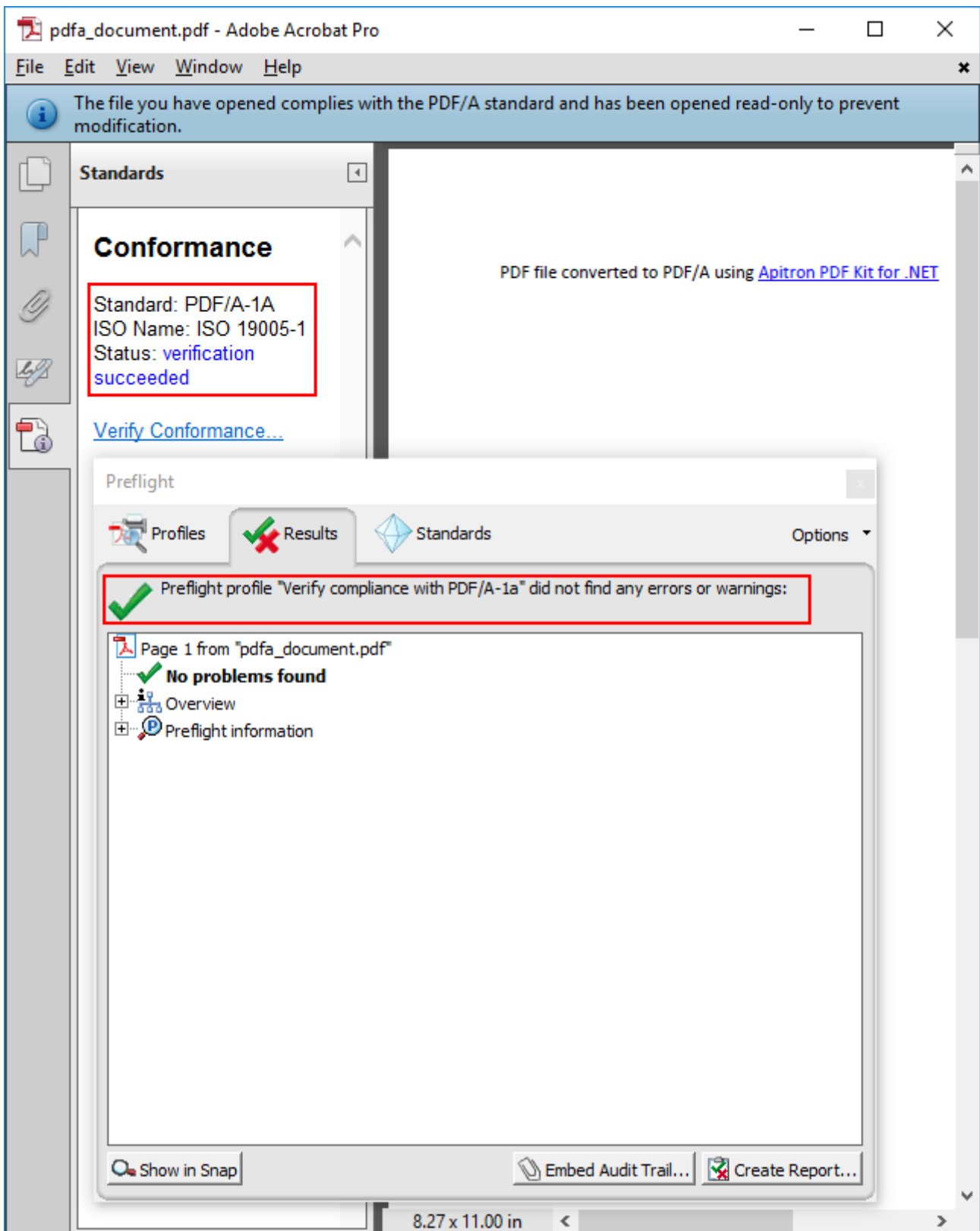
```
class Program
{
    static void Main(string[] args)
    {
        using (Stream stream = File.Open(@"../../data/document.pdf",
            FileMode.Open, FileAccess.Read))
        {
            // create document object and specify the output format
            FixedDocument doc = new FixedDocument(stream, PdfStandard.PDFA);

            // save document
            using (Stream outputStream = File.Create(@"pdfa_document.pdf"))
            {
                // turn off cross reference stream usage
                doc.IsCompressedStructure = false;
                doc.Save(outputStream);
            }
        }

        Process.Start("pdfa_document.pdf");
    }
}
```

You see that by setting the `IsCompressedStructure` property it's not possible to control cross reference streams usage. The complete code sample can be found in our [github repo](#).

The image below demonstrates PDF/A document validation using Adobe Acrobat Pro:



Pic. 1 PDF/A validation

## Summary

The [Apitron PDF Kit for .NET](#) is a powerful library for creation and manipulation of PDF and PDF/A documents. This product has many unique features, offers easy to use API and is cross-platform that means you can create apps for .net(windows, windows phone, windows store), ios & android (via xamarin) and mono targeting modern mobile, desktop and web platforms at once. Contact us and we'll be happy to answer your questions.