

2011

XFA 简介

简单了解 xfa

简单了解 xfa 是什么 ; 什么样的文件是 xfa 文件 ; xfa 文件打开方式控制的标准、数据的存储及导入导出等五方面的内容。

第一部分 什么是 XFA？

XFA：XML Forms Architecture

XML: extensible markup language(可扩展标记语言)

XFA is similar to PDF interactive forms introduced in PDF 1.2, which is also known as AcroForm, with the following differences:

- XFA can be used in XML-based workflows.

- XFA separates data from the XFA template, which allows greater flexibility in the structure of the data supported and which allows data to be packaged separately from the form.

- XFA can specify dynamically-growing forms.

- XFA can specify Web interactions, such as HTTP and Web Services (WS-DL). Such interactions can be used to submit data to a server or to request a server perform a calculation and return the result.

- XFA works with other XML grammars.

第二部分 判定文档为 xfa文档的标准

经分析：PDF字典项中 /catalog 中的 /AcroForm 中有 /XFA，说明这是一个 xfa文档，如果没有 /XFA，说明该文档为普通的 PDF文档。

/XFA	Result
有	XFA文档
无	普通 PDF文档

表 1 是否为 xfa文档判定表

第三部分 控制文档是以动态还是静态打开的判定标准

一、控制是以动态还是静态打开的判定标准的前提条件

文件必须是 xfa文档，才能开始判断文档在打开的时候是以静态还是动态的形式打开，这个可以根据“表 1 是否为 xfa文档判定表”判断。

二、控制是以动态还是静态打开的判定标准

经过分析 Acrobat在 config有特有的关键字，控制是以动态还是静态打开 xfa，如下：

```
<config>
  <acrobat>
    <acrobat7>
      <dynamicRender>forbidden ( 默认值 ) |required</dynamicRender>
    </acrobat7>
  </acrobat>
</config>
```

下面表格是根据测试文档验证得到的验证结论表：

dynamicRender	NeedsRendering	Result
required	True	Dynamic
	False	Dynamic
	不存在	Dynamic
forbidden	True	Static
	False	Static
	不存在	Static
不存在	True	Static
	False	Static
	不存在	Static

表 2 静态和动态 xfa文档判定表

说明：上面表格中 `dynamicRender` `NeedsRendering` 分别为 PDF 文档中的关键字，`Result` 为测试结果（`Dynamic` 说明文档以动态 xfa 打开，`Static` 说明文档以静态 xfa 打开）。

由上面表格可以看到，xfa 文档的打开方式和 `NeedsRendering` 关键字没有关系，只和 `<dynamicRender></dynamicRender>` 的值有关。

三、结论

综上所述，首先，PDF 字典项 `/Catalog` 中的 `/AcroForm` 中有 `/XFA` 项。

再者，如果关键字 `dynamicRender` 的值为 `required`，无论 `NeedsRendering` 的值为何值，则文档以动态 xfa 方式打开；如果关键字 `dynamicRender` 的值为 `forb idder` 或者不存在，无论 `NeedsRendering` 的值为何值，则文档以静态 xfa 方式打开。

第四部分 XFA 文档中的数据存储

对于文档中的数据，有些是默认数据，有些是用户填写数据，下面就简单说一下这两种数据在文档中的存储情况。

一、默认值的存储

xfa文档中的一些表单是有默认值的，动态和静态的默认值的存储并不相同。

动态表单文件

默认值只存在一个地方，那就是存储在模板中，我们可以通过字典项信息找到这些值，如下：

```
/XFA [(preamble) 1 0 R (config) 2 0 R (template) 3 0 R (localeSet) 4 0 R (xmpmeta) 5 0 R (postamble) 6 0 R]
```

我们可以通过查找 template对应的信息来找到默认值：

```
3 0 obj
<< /Filter /FlateDecode /Length 813 >> stream
< template xlns= "http://www.xfa.org/schemas/xfa-template/2.8/"
>
.....
< field name= "TextField1" y= "47.625mm" x= "53.975mm" w= "62mm" h= "9mm"
>
.....
< caption reserve= "25mm"
>
< value
>< text
> test1</text
></value
></caption
>< value
>< text
> thank</text
></value
></field
>
.....
</template>
endstream
endobj
```

静态表单文件

默认值在 3 个地方有存储，分别在：

1. 模板

模板里的存储和动态表单存储保持一致。

2. Datasets

datasets里的存储可以通过查找字典项 /AcroForm 中的 /XFA 中的 “

[(xdp xdp) 54 0 R (config) 2 0 R (template) 3 0 R (localSet) 4 0 R (xmpmeta) 5 0 R (form) 55 0 R (datasets) 56 0 R (< /xdp xdp>) 57 0 R]“ 中的 (datasets) 56 0 R 来查看数据的存储情况。

3. /DV

/DV 在字典项中是存储默认值的。

可以通过查找字典项中的 /AcroForm 中的 /Fields 中的 /Kids.. 中的字典项

“ << /DA (MyriadPro-Regular10.00 Tf0 g) /DV (thank) /F 4 /FT /Tx MK<<>> /P 36 0 R /Parent 17 0 R /Q 0 /Rect [262.700989 605.835022 366.165009 645.164978] /StructParent 0 /Sub type /Widget /T (TextField1[0]) /TU (test1) /Type /Annot>> ” 中的 /DV 来查看数据的存储情况。

上面三个地方的存储数据是一致的。

二、非默认值的存储

动态 xfa 文档中的数据存储

动态 xfa 文档填写数据后，在字典项中的存储位置为 datasets 中；

可以通过查找字典项 /AcroForm 中的 /XFA 中的 “

[(xdp xdp) 54 0 R (config) 2 0 R (template) 3 0 R (localSet) 4 0 R (xmpmeta) 5 0 R (form) 55 0 R (datasets) 56 0 R (< /xdp xdp>) 57 0 R]“ 中的 (datasets) 56 0 R 来查看数据的存储情况。

即便是修改后的数据仍然是保存到这。

静态 xfa 文档中的数据存储

静态 xfa 文档填写数据后，在字典项中有两个位置进行了保存，分别为 datasets 和 /v 中，且两个位置的数据存储应该是一致的。

(1) 数据在 datasets 中的存储

可以通过查找字典项中的 /AcroForm 中的 /XFA 中的 “ [(xdp xdp) 1 0 R (config) 2 0 R (template) 3 0 R (localSet) 4 0 R (xmpmeta) 5 0 R (form) 6 0 R (datasets) 7 0 R (< /xdp xdp>) 8 0 R] “ 中的 (datasets) 7 0 R 来查看数据的存储情况。

(2) 数据在 /V 中的存储

可以通过查找字典项中的 /AcroForm 中的 /Fields 中的 /Kids.. 中的字典项 “ << /AP << /N 36 0 R >> /DA (MyriadPro-Regular 10.00 Tf 0 g) /F 4 /FT /Tx MK << >> /P 35 0 R /Parent 16 0 R /Q 0 /Rect [244.701004 643.322998 343.912994 663.164978] /StructParent 0 /Subtype /Widget /T (TextField1[0]) /TU (test1) /Type /Annot /V (good) >> ” 中的 /V 来查看数据的存储情况。

综上所述，动态表单的非默认值存储只存在 datasets 中；但是静态表单的存储相对特殊，存在两个地方，datasets 和 /V 中，且这两处数据必须保持一致，由于这样的存储结构，对于静态表单来说容易出现对象查找不到的情况。

举例：

静态表单和普通表单中遇到的特殊情况：显示数据与编辑状态的数据不同

显示状态的数据显示的是外观流，正常情况下 /V 中的数据是和 /AP 外观流一致的，但某些情况下导致了外观了和编辑状态的数据不一致，比如，一个表单外观显示有数据，但是当鼠标焦点放在域中的时候，并没有数据显示，当鼠标离开时发现又有数据显示，这是为什么呢？

答案是这样的：

对于静态表单

这是因为 datasets 中并没有数据，而 /AP 中确有外观流显示，这一点的现象是和普通表单保持一致的。

对于普通表单

这是因为 /V 中并没有数据，而 /AP 中确有外观流显示。

第五部分 数据的导入导出

一、数据的导出

XFA 表单的数据是存储在 XML 数据集里的，导入导出的数据和普通的表单导入导出数据并不相同。
先简单说一下普通表单的导入导出：
导出的数据是来自字典项 IV 中的数据；自然导入数据后也是对应这部分数据。
而 XFA 文件的导入导出是数据集中存储的数据。

数据的导出格式

XFA 文件数据导出格式有两种：XML 和 XDP 格式。

导出数据的源

对于静态表单而言

无论文件是否有默认值，它导出数据的来源都是数据集 (datasets) 里的数据。

对于动态表单而言

无论文件是否有默认值，它导出数据的来源都是数据集 (datasets) 里的数据。

动态表单和静态表单的数据存储虽然不同，但是，导出数据的源是相同的。

这里要说明一点 含有默认值的动态 xfa 文档，如果没有对文档做任何修改，它的 datasets 是在文档打开的时候生成，关闭的时候消失，所以用 PDFSpy 工具是不能查到该部分内容的。
不过我们可以通过 js 脚本来测试这种情况，如下：

假如动态文件中只含有一个含有默认值的 TextField，其 name: TextField1, caption: test1；
含有一个 TextField2，其 name: TextField2, caption: value，但是没有任何默认值，创建该域是为了测试 test1 的 datasets 在打开时是否创建了；Button 按钮是设置响应的。

test1

thank

value

Button

我们设置 Button的响应是 点击 Button按钮，设置 value的值为 datasets中 TextField1的 value，其脚本如下：

```
TextField2.rawValue = xfa.datasets.data.form1.TextField1.value;
```

如果我们点击 Button，value对应的 field的值显示和 test1的不一致，说明 datasets在打开文档的时候没有生成；

假如我们点击 Button，value对应的 field的值显示和 test1的一致，说明 datasets在打开文档的时候生成了。

大家不要误认为：含有默认值的没有修改过的动态文档，导出的数据来自模板，从上面分析来看，无论是否有默认值，导出数据始终来自 datasets

二、数据的导入

由数据的导出源为 datasets，我们推想，可以得数据应该是对应的导入到 datasets中，用下面例子，让大家了解一下。

假如，动态文件中只含有一个含有默认值的 TextField，其 name: TextField1，caption: test1，含有一个 TextField2，其 name: TextField2，caption value，但是没有任何默认值；含有一个 Button按钮，如下显示：

test1	<input type="text" value="thank"/>
value	<input type="text"/>
<input type="button" value="Button"/>	

此时字典项中是没有 datasets，如下：

```
/XFA [(preamble)1 0 R (config)2 0 R (template)3 0 R (localeSet)4 0 R (xmpmeta)5 0 R (xpdf)6 0 R (postamble)7 0 R]
```

观察执行下述操作后，字典项中的变化。

1. 修改 test1对应的值为：thankyou；
2. 导出数据到 xmp 或 xdp 格式；
3. 对表单文件执行 reset操作；
4. 导入步骤 2导出的文件，然后保存文档；
5. 用 PDFSpy查看该 PDF文档，查看字典项中对应的域，如下：

```
/XFA [(preamble)1 0 R (config)2 0 R (template)3 0 R (localeSet)4 0 R (xmpmeta)5 0 R (xpdf)6 0 R (datasets)44 0 R (form)45 0 R (postamble)7 0 R]
```

(datasets) 44 0 R对应的信息如下：

```
44 0 obj
<< /Length 123 /Filter /FlateDecode >> stream
<xfa datasets xmlnsxfa="http://www.xfa.org/schema/xfa-data/1.0/"
><xfa data xmlnsxfa="http://www.xfa.org/schema/xfa-data/1.0/"
><form 1
><TextField1
>thankyou</TextField1><TextField2
/></form 1></xfa data></xfa datasets>
endstream
endobj
```

对比操作前后的变化，字典项中多了 (datasets) 44 0 R这部分及这部分对应的数据。

由于上面的例子是以动态文件中含有默认值文档进行比较的，所以会有这样的变化，如果文档中已经写入了数据，那么变化的只是 (datasets) 44 0 R这部分对应的数据，即 44 0 obj这部分对应域的数据；如果是静态文件，那么变化也只是 (datasets) 44 0 R这部分对应的数据，即 44 0 obj这部分对应域的数据大家不要混淆。

由上述分析可看到，xfa文档，数据导入到对应的数据集 (datasets) 中。

文献

XML Forms Architecture (XFA) Specification Version 3.1 pdf.
pdf_reference_17 pdf.