试题的存储格式研究

三年一个轮回，再次来到高三，发现在网络技术的推波助澜下，试题呈现出几何级数般的增长。在数量的激增之下，试题的质量自然也有相应的显著提高，以地理试题而论，在题目的题材新颖、设计思路的巧妙、高效的融合知识要点、精美的绘图等方面，都出现了不少值得赞叹的好题。但是世界总是对称的，偏题、怪题、俗不可耐的烂题也是如潮水般涌现。如何加强对试题的甄别、管理和使用就显得非常重要了。(而另一个问题是Adaptive Testing就更重要了。)

要解决以上两个问题的基础是，你的研究对象不是试卷，不是考试，而是试题。研究尺度的微观化是提高研究层次的重要手段。而研究试题在技术实现的层面来讲，首先面临的一个绕不过去的问题是：你的试题将以何种文件格式存储？(其实这个问题可能是个伪问题，下次另文叙述)

近来主要考虑了这几种可能性：

1 IMS QTI格式

几乎每家著名的考试机构都有自己的试题文件格式，但是一直以来缺乏一个统一的标准，直到后来IMS发布了QTI试题规范。上个月刚刚发布新的v3.0版本。但是标准发布之后似乎并没有得到广泛的支持。

优点：

i) 规范本身全面而且强大，支持几乎所有常见题型

ii) 适合于网络数据交互

Iii) 常用网络教学系统，如Moodle, Edx等基本都支持或部分支持这一规范。

缺点：

i) 目前影响力不大，未来局势不明朗

ii) 试题的表达能力不够强大，我觉得它难以存储我们的高考试卷(这个不是很确定，有待继续观察)

iii) 缺乏高效的编辑工具，常用的Office软件都不支持

结论：观察

2 PDF格式

PDF文件格式是印刷行业的印前标准，现在也逐渐被国人认识。

优点：

i)是跨平台的开放标准。

ii)浏览器的支持相当好。

iii)支持矢量图形，放大缩小不影响清晰度。

iv)编程友好，现成的第三库很丰富，如PyPDF2等，估计PuMuPDF是最强的的库，当然这里说的都是python库，其他编程语言的PDF库我不熟悉，估计Java有更强的库。

缺点：

i)文件与常用字处理软件的结合不太紧密

ii)文件结构以页为单位，因此对文件的常规操作，如merge和split等操作，目前的 python第三方库都不支持。这也许是由PDF文件不欢迎修改的先天属性注定了的。

3 Word格式

Word格式，或者说docx文件，全天下的人都知道都在用的文件格式，优点就不用多说了，尤其是文件的表现力强，有强大的字处理软件配套，便于编辑。第三库支持良好，python-docx的功能完备，但是这个python库的不足是对中文的支持存在缺陷，上游已经长期沉默，众多pull request没人打理。Merge操作借助python-docxcompose，操作几乎完美。

缺点：与浏览器的融合表现不佳，利用py-mammoth可以将docx文件转换为html，但是我之前的简单试用至少发现两个问题，一是转换后中文乱码，二是图像转换后大小失去控制。我不知道是否存在更好的利用Javascript等其他的解决方案。

目前这种格式最大的好处是可以很方便地使用python操作docx文件，尤其是文件的拆分和重组。最大的问题是，docx文件不适合在浏览器中表达。一个可能的间接方案是利用docx2pdf库把docx文件转换为pdf文件再由浏览器显示，但这种解决方案必须依赖于Windows Server服务器，我不喜欢，而且估计在生产环境服务器端的压力会相当大。

4 LaTeX文件

我总是先验的觉得LaTeX文件格式是最终的解决方案，但是LaTeX文件过于复杂，一百个人里未必有一个知道LaTeX，而一百个知道LaTeX的人里未必有一个会用，所谓曲高和寡阳春白雪的，就是说它了。

这个还是姑且不论，至少眼前讨论把我们的试题试卷保存为LaTeX文件是极不现实的。