# 在线文档预览用户手册

I Doc View

2014年10月

## 第一章、概述

I Doc View在线文档预览支持以下格式文件的在线预览：

文档：doc,docx,wps,odt,rtf,xls,xlsx,et,ods,csv,ppt,pptx,dps,odp,pdf,txt

图片：jpg, jpeg, gif, png, bmp, tif

音频：mp3, m4a, mid, wma

压缩文件：zip, rar, tar, 7z

图纸：dwg, dxf, dwf

编程：html,htm,xml,js,css,java,php,sql

I Doc View在线文档预览分为单机版和集群版。

单机版适用场景

用户量5万以下

日预览量百万级以下

上传并发平均150个文档/秒

预览并发平均1000个文档/秒

单个文档转换平均2秒

注：以上参数根据带宽和机器配置有所不同

硬件配置：

内存：8G或以上（中小规模预览）

如果预览量大，建议配置SSD，并且将Office安装在SSD硬盘中。

如果超过以上访问量，建议配置集群版。

系统环境：

Windows Server 2016 64位、2012 64位、2008 R2 64位(最低版本，不推荐)

**特别提示：请自行进行数据冗余备份**

## 第二章、特色

### 1. 预览本地文档

您可以上传并在线查看您计算机的文档。

### 2. 预览网络文档

您可以对网络文档进行预览。

### 3. 阅读进度条

doc, docx, ppt, pptx和txt文档大于约3页时，预览页面底部会出现阅读进度条。

### 4. 页面定位

doc, docx和txt文档大于约3页时，预览页面底部会出现一个页面定位区域，区域内每个圆点“·”代表一个页面，从左到右依次增加，圆点“·”越多，文档越大。鼠标移动到页面定位区域后会变为手形，点击即可跳转到对应页面；

ppt, pptx文档预览时，电脑端可使用“↑”、“←”箭头向上翻页，“→”、“↓”或空格键向下翻页，或直接点击左侧缩略图跳转到对应页面，手机端则既可通过页面选择器定位页面，也可左右滑动来切换幻灯片，pad端则既可通过左侧缩略图定位页面，也可在右侧预览图上左右滑动来切换幻灯片。

### 5. 安全等级

每个文档都可以设置两种访问方式：

公开：任何人可以访问并预览，适用于公开文档

私有：只能通过会话ID预览，会话过期后就需要重新获取预览会话ID，该预览方式适用于私有文档

### 6. 浏览器兼容

在线文档预览兼容主流浏览器，如：Chrome浏览器（推荐）、IE8及以上版本、火狐浏览器、Safari浏览器、360浏览器、Opera浏览器、搜狗浏览器等。

### 7. 适应不同终端

文档预览采用响应式设计，在不同的终端（如笔记本、平板电脑或手机等）上预览时，文档会自动调整页面布局达到最佳效果。例如：在预览ppt或pptx文档时，手机、平板和电脑预览效果不同，左侧缩略图只在电脑或平板预览时生效，在屏幕较小的手机端则隐藏缩略图，但会出现手机端专有的页面选择器来定位页面。

### 8. 加载优化

doc, docx和txt文档预览按需获取、分页加载，减少服务器、带宽和终端压力；

ppt, pptx预览时，左侧缩略图每张大约5k，假如有20页的PPT，所有缩略图总和也只有100k左右，右侧一张大图60k左右，初次页面加载图片总量只有160k左右，大大减少服务器、带宽和终端压力，手机端预览更是隐藏缩略图，大大节省网络流量。

### 9. 易扩展

对于非默认预览格式的文件，提供方便的扩展方式以供客户自定义预览页面。

### 10. 集群部署

对于高并发环境，可方便地进行集群部署，并可线性扩展。

## 第三章、系统架构

### 1. 单机版架构

I Doc View在线文档预览服务

Tomcat/Jetty

Nginx/Apache

数据库

I Doc View文档转换器

网页浏览器：手机、平板或电脑

原始文档

转换文档

I Doc View在线文档预览服务所需软件分为两类：

1. 私有软件：包括“在线文档预览服务WEB应用”和“在线文档预览转换器”，这两种软件的获取与授权详情请联系I Doc View客服；

2. 公有软件：公有软件为I Doc View在线文档预览服务正常运行所需的其他软件，大部分开源免费，根据自己实际情况从其官网获取，也可从I Doc View提供的统一下载地址来获取。

图中绿色部分为私有软件。

### 2. 集群版架构

...



数据库

原始文档

**服务层**

**用户层**

**存储层**

**代理层**



分布式云存储



转换文档

上传

预览

T3

T2

T1

N1

N2

应用系统

预览流程：

1. 用户通过第三方应用系统（以下简称“应用系统”）将所有文件上传到“分布式云存储”（或者现有的文件存储系统），并在应用系统中记录文件基本信息如文件名、大小、MD5、下载链接等；

2. 用户需要预览上传的文件时，应用系统在预览链接后面添加文件MD5值参数（或文档唯一标识uuid，或者应用系统为文档生成的唯一ID等，后续用作一致性哈希），预览链接格式如：

http://api.idocv.com/view/url?url=<UTF-8 Encoded URL>&name=<可选文件名>&md5=<file md5>

或者：

http://api.idocv.com/view/<uuid>?uuid=<uuid>（这种方式需要在下面的Nginx配置中修改$arg\_md5为$arg\_uuid）；

3. 用户点击预览链接时候，访问请求到达代理层。代理层有两台互为热备的基于Linux系统的Nginx代理（图中N1和N2），Nginx配置示例：

upstream docview\_backend\_cluster {

hash $arg\_md5 consistent;

server docview1;

server docview2;

server docview3;

}

server {

server\_name www.idocv.com;

location / {

proxy\_pass http://docview\_backend\_cluster;

}

}

参考：

https://www.nginx.com/blog/choosing-nginx-plus-load-balancing-techniques/

http://nginx.org/en/docs/http/ngx\_http\_upstream\_module.html#hash

http://nginx.org/en/docs/varindex.html

以上配置中，$arg\_md5为预览链接中的MD5参数值（可为其他参数如uuid、文件唯一标识id等），Nginx根据该参数值做一致性哈希，将预览请求分发给服务层的后端Tomcat预览服务docview1, docview2或docview3（对应图中T1, T2或T3）；

4. 服务层的Tomcat预览服务接收到请求后，首先根据MD5值判断是否成功预览过相同MD5的文件，如果预览过，则直接展示给用户，否则，根据url参数再次判断是否预览过相同链接的文档（可配置），如果预览过则直接展示给用户，否则，预览服务根据url参数去分布式云存储中获取原始文档，并在存储层的数据库和磁盘中分别存储文件元数据和文件实体，然后做转换处理，如果转换成功则直接预览，如果转换失败则提示失败信息。

说明

a). 文件存储：所有文件均统一存储在分布式云存储或已有的文件存储系统中；

b). 有效转换：因为预览链接中有文件MD5参数（或uuid、应用系统文件唯一标识ID等），代理层的Nginx做一致性哈希，每次都会将同一个文件的预览请求分发到相同的后端预览服务，避免重复转换；

c). 线性扩展：当预览服务器宕机时，Nginx会自动探测并摘除，当预览服务器需要扩展时，只需要水平扩展添加预览服务器即可，不管是摘除或添加预览服务器，根据一致性哈希原理，只会影响少量数据的自动调整，保证整体预览集群的稳定高速；

d). 高可用性：代理层的两台Nginx做了双击热备做到高可用，后端预览服务宕机会被Nginx自动感知并临时摘除，落到宕机服务器的请求会自动分发到附近预览服务器。任何服务器的宕机都不会影响整个预览服务；

e). 清理数据：当预览服务器的磁盘空间不足时，可以开启自动删除旧数据功能，只保留X天的数据，具体配置参见预览服务配置文件。

## 第四章、接入方式

### 方式1：普通接入

如果原始文档需要用户先上传，然后再预览，可以使用该接入方式。

接入时，可以先调用预览服务的上传接口：

http://api.idocv.com/doc/upload

该上传接口可以同时支持本地文档上传（file）参数和互联网文档上传（url），但两个参数必须选择其一。详细说明参见接口文档。

上传后，预览服务会返回文档的唯一标识uuid，可以将此uuid存入您的数据库，之后的预览和其他操作需要传入该参数。

预览的时候，可以调用预览接口：

http://api.idocv.com/view/<UUID>

其中，<UUID>就是调用上传接口返回的uuid。

### 方式2：互联网文档接入

如果您的应用系统中已经有文档的“下载”链接，需要在旁边添加“预览”链接，可以直接调用URL预览接口，而不需要调用上传接口。注意：该下载链接可以被预览服务器直接（不需要登录）访问到。

URL预览接口如下：  
http://api.idocv.com/view/url?url=<UTF-8 Encoded URL>&name=<可选文件名>

其中，<UTF-8 Encoded URL>是您文档的下载url，需要进行URLEncode编码，如果您使用Java语言，可以使用

java.net.URLEncoder.encode("文档下载链接", "UTF-8");

来生成编码url，如果您使用javascript，可以使用

encodeURIComponent(‘文档下载链接’);

来生成编码url。

name参数为需要显示的文件名。预览服务获取文件名方式如下：

如果传入name参数，则以name参数为准；

如果不传name参数，程序会自动从名称为“Content-Disposition”的header里获取filename作为文件名；

如果header里获取不到filename，则根据文档的下载链接来截取文件名（如下载链接为http://\*\*\*/test.docx，则自动获取文件名为test.docx），如果下载链接为动态链接，末尾不是“.docx”（docx可以是预览服务器支持的任何格式后缀）这样的形式，则获取文件名失败，返回失败信息。

示例：

a) 网络文档下载url为：

http://api.idocv.com/data/doc/在线文档预览用户手册.docx

b) Encode(UTF-8编码)后的URL为：

http%3a%2f%2fapi.idocv.com%2fdata%2fdoc%2f%e5%9c%a8%e7%ba%bf%e6%96%87%e6%a1%a3%e9%a2%84%e8%a7%88%e7%94%a8%e6%88%b7%e6%89%8b%e5%86%8c.docx

c) 最终预览URL为：

http://api.idocv.com/view/url?url=http%3a%2f%2fapi.idocv.com%2fdata%2fdoc%2f%e5%9c%a8%e7%ba%bf%e6%96%87%e6%a1%a3%e9%a2%84%e8%a7%88%e7%94%a8%e6%88%b7%e6%89%8b%e5%86%8c.docx

### 方式3：预览服务器文档接入

如果原始文档在预览服务器的某个磁盘路径下，您也可以直接使用URL预览方式，无需调用上传接口。

调用接口跟预览网络文档接口相同：

http://api.idocv.com/view/url?url=<UTF-8 Encoded URL>&name=<可选文件名>

只是<UTF-8 Encoded URL>部分为Encode后的您预览服务器本地文档路径。

示例：

a) 本地文档路径：

D:/测试 目录/测试 文件.txt

b) 本地文档路径URL：

file:///D:/测试 目录/测试 文件.txt

c) Encode(UTF-8编码)后的本地文档URL：

file%3A%2F%2F%2FD%3A%2F%E6%B5%8B%E8%AF%95+%E7%9B%AE%E5%BD%95%2F%E6%B5%8B%E8%AF%95+%E6%96%87%E4%BB%B6.txt

d) 最终预览URL为：

http://api.idocv.com/view/url?url=file%3A%2F%2F%2FD%3A%2F%E6%B5%8B%E8%AF%95+%E7%9B%AE%E5%BD%95%2F%E6%B5%8B%E8%AF%95+%E6%96%87%E4%BB%B6.txt

如果原始文档存储在预览服务器上，建议使用该预览方式，服务器会直接读取本地文件，不通过网络，速度更快。

### 展示

大部分情况下，第三方系统需要在自己的文档列表页的文档“下载”链接旁边添加“预览”链接的方式来集成在线文档预览服务，这时候只需要在“下载”链接旁边添加如下代码：

<a href="<view url>" target="\_blank">预览</a>

其中，<view url>是实际预览地址，例如：

<a href="http://api.idocv.com/view/url?url=http%3A%2F%2Fdata.idocv.com%2Fidocv-manual.docx" target="\_blank">预览</a>

但有时候，客户需要直接将预览页面嵌入自己的模板中，这时可以使用iFrame方式来集成嵌入。

首先需要在要嵌入预览页面的地方添加：

<iframe src="http://api.idocv.com/view/HwIqnTw.html" width="100%" scrolling="no" style="min-height: 600px;"></iframe>

其次需要添加如下js：

<script src="http://api.idocv.com/static/iframe-resizer/js/iframeResizer.min.js"></script>

<script type="text/javascript">

iFrameResize();

</script>

2015年04月

I Doc View技术部