

# hDavinci Server Application

## 常见问题及解决办法

### 目录

1.版本历史 .....	2
2.概述 .....	3
3.Davinci-server 文件结构及各自详细定义.....	4
3.1 文件层次结构.....	4
3.2 orginfile.pdf.....	4
3.3 Filename.xml.....	4
3.4 acid_gapsystems.pdf .....	4
3.5 acid_subfolder.....	4
3.6 acid_pagenum_files 文件夹 .....	6
3.7 separations_folder .....	7
4.davinci-process-service 错误跟踪及恢复 .....	8
4.1 log 位置.....	8
4.2 错误分类.....	8
4.2.1 系统定义的错误.....	8
4.2.2 未定义错误.....	9
4.3 解决办法.....	10
4.3.1 图像转换方面问题.....	10
4.3.2 数据库连接出错的情况.....	13
4.3.3 对象序列化的问题.....	14

## 1.版本历史

1.0	Echo Lee	2011-07-21
-----	----------	------------

## 2.概述

本文档主要阐述了 Davinci-server 的日记结构,Davinci 生成的文件结构,以及如何通过日记监控错误,当监控到错误发生后,如何手工恢复错误.

因为 Davinci-server 的服务器端分为两个 module(dvview 和 dvprocess).因此在进行错误判断和日记监控后,如果能准确的判断错误发生的 module,将会减少很多麻烦.

从错误的 module 进行划分判断:

Davinci-process-service 主要是图片转换类的错误,比如图片转换的某个环节出错,比如后台做切片处理的时候因为内存不够的原因导致的内存溢出,这一类错误在客户端都有明确的 errorCode 显示.

Davinci-view-service 主要是图片的一些辅助信息获取的问题,比如添加 annotation,添加 approval,获取图片的整个 annotation 信息,获取图片整个 approval 信息,或者是 startprocess,createSession 相关的问题.

### 3.Davinci-server 文件结构及各自详细定义.

#### 3.1 文件层次结构

```
..... /orginfile.pdf
..... /Filename.xml
..... /acid\_gapsystems.pdf
..... /acid\_subfolder/
..... /acid_subfolder /thumb.jpg
..... /acid_pagenum.xml
..... /acid_pagenum_slice_folder/
..... /separations_folder/
..... /separations_folder/acid\_pagenum\_separatecolour.xml
..... /acid_pagenum_separatecolour_slice_folder/
```

#### 3.2 orginfile.pdf

这个是由用户上传的源文件.有可能是 pdf 也可以是 jpg 格式.这里只是为了方便说明因此使用 pdf.

#### 3.3 Filename.xml

这个文件主要保存了上传文件的 annotations 和 approvals 信息,以及上传文件的 acid 等信息.

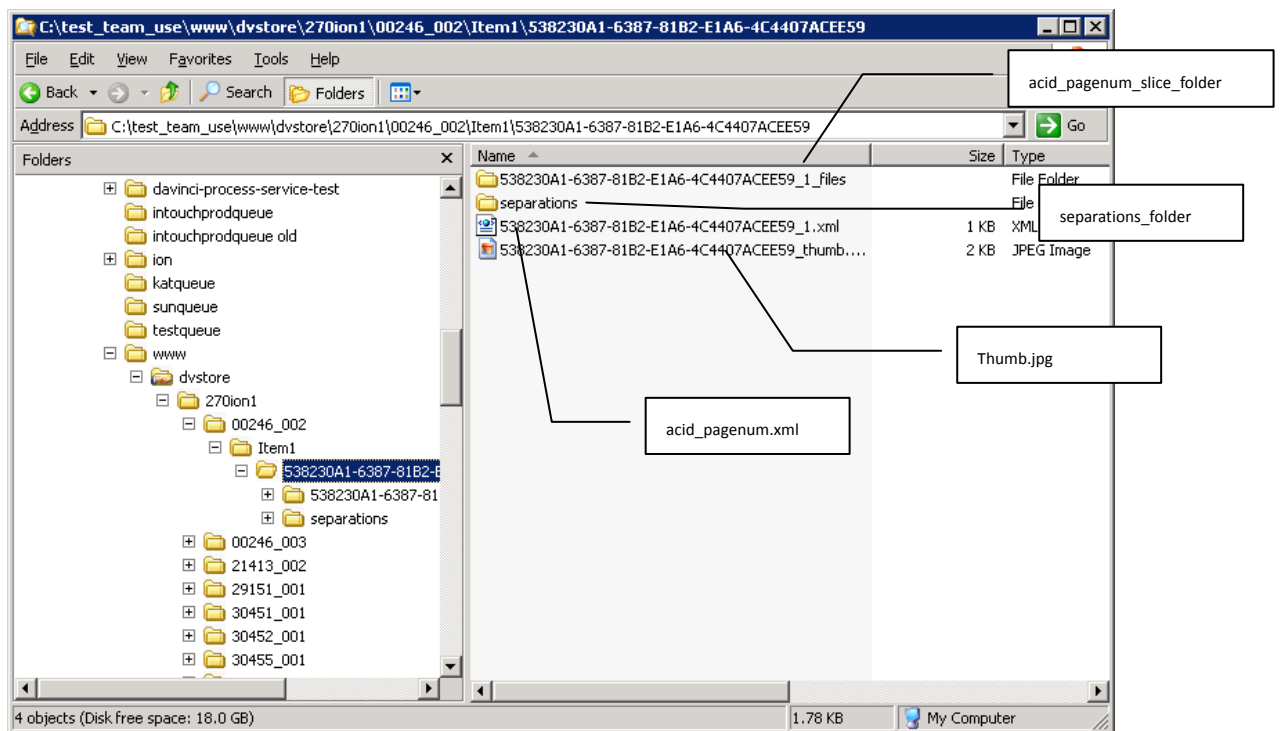
#### 3.4 acid\_gapsystems.pdf

这个是 orginfile.pdf 的一个完整拷贝,重新使用 acid+“\_gapsystems”的方式进行命名,目的是解决因为特殊字符的文件名的缘故,使得 gs 或者 imagemagick 不能正常工作.

#### 3.5 acid\_subfolder

是保存了 dvprocess 处理过的每一页图像的切片文件夹(acid\_pagenum\_slice\_folder),每一页图像处理状态文件 (acid\_pagenum.xml), 第一页的缩略图 (thumb.jpg), 分离颜色的文件夹 (separations\_folder),

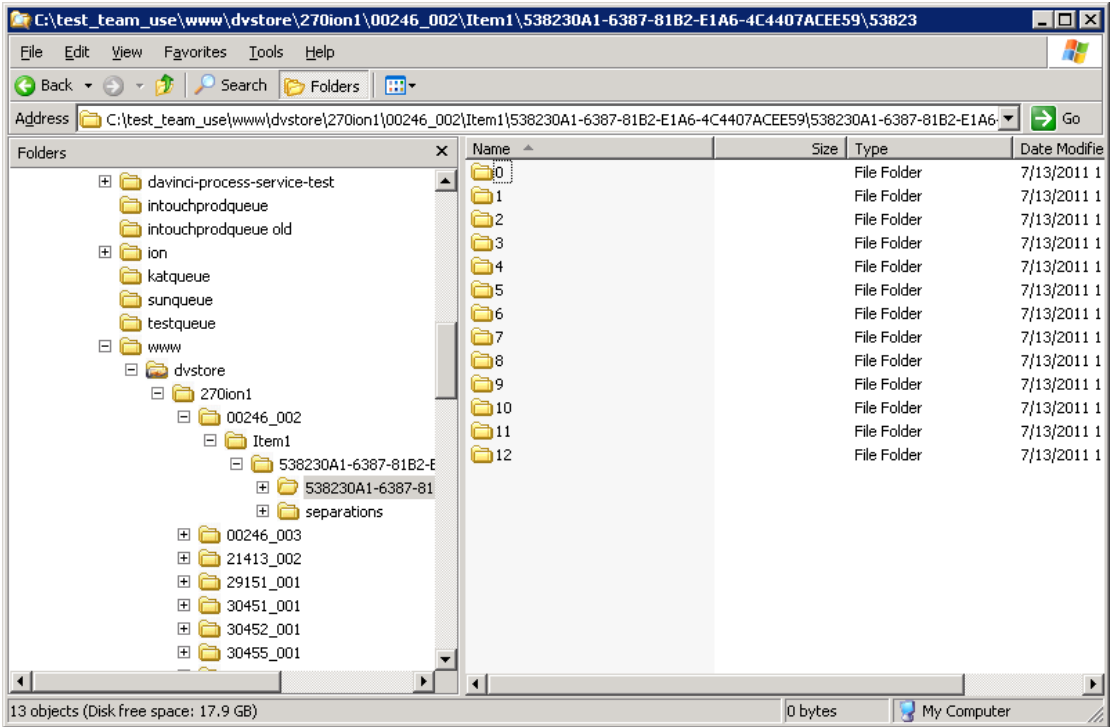
如下图所示:



- a) acid\_pagenum\_slice\_folder 保存的是对应 pagenum 的页码中的图像,转换成 jpg 格式的图  
片后,经过 dvprocess 生成的切片切片
- b) acid\_pagenum.xml 保存的是每一页的图片处理的状态信息,以及长宽信息等.
- c) thumb.jpg 源文件的缩略图
- d) separations\_folder 根据 CMYK 色彩空间,分离的颜色的图片切片文件夹

3.6 acid\_pagenum\_files 文件夹

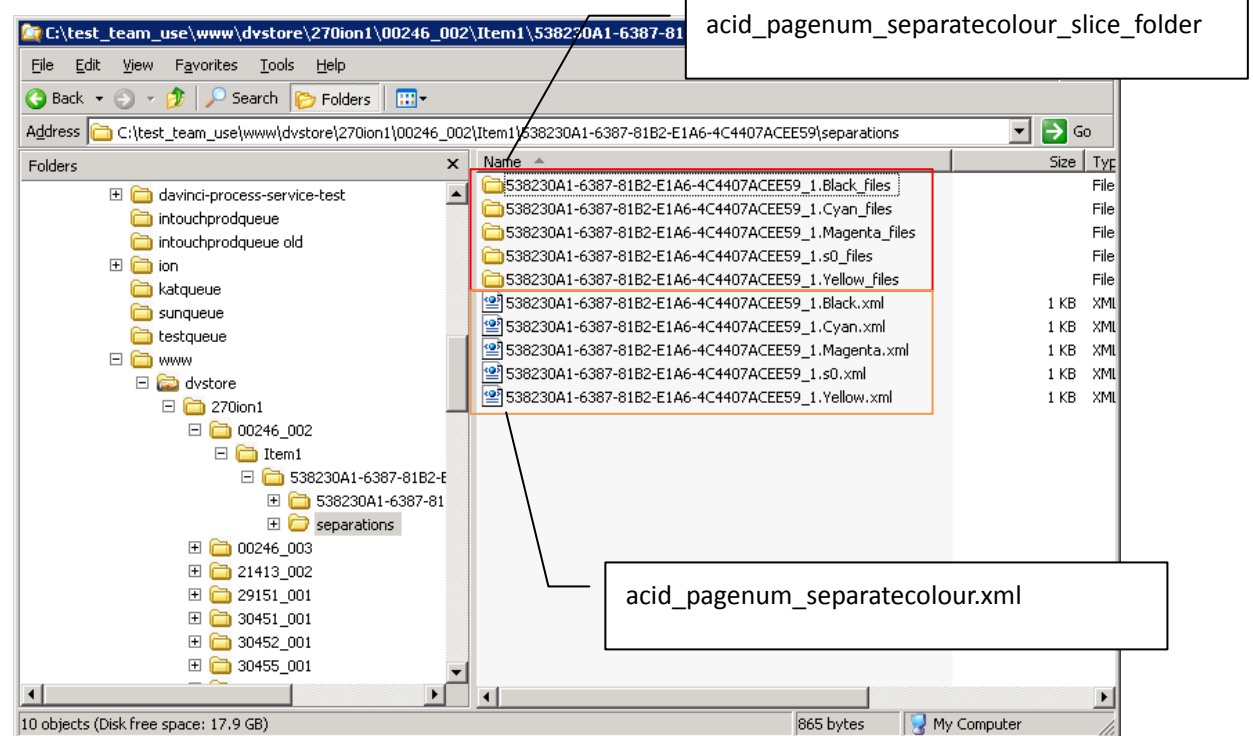
acid\_pagenum\_slice\_folder 的文件组织情况:



按照序号 0~12 分别是不同放大级别的图像切片.最大的序号数字是原图的级别的切片,也是最清晰级别的图像切片.12 不是最大的层级,随着图片的尺寸越大,最大等级也越大.

### 3.7 separations\_folder

separations\_folder 文件夹下的文件组织:



- a) acid\_pagenum\_separatecolour\_slice\_folder 对应页码对应分离的颜色的图层图像的切片文件夹(至少有 Black,Cyan,Magenta,Yello 四种颜色分离层)
- b) acid\_pagenum\_separatecolour.xml 对应颜色对应页码的切片状态信息文件.

4.davinci-process-service 错误跟踪及恢复

4.1 log 位置

Dvprocess 的 log 路径是:  
\${davinci-process-service}/logs/davinci.process.log  
或  
\${davinci-process-service}/logs/ davinci.process.log. yyyy-MM-dd  
或  
\${davinci-process-service}/logs/stdout.log  
-

这三个区别分别是:davinci.process.log 是当天的 dvprocess 的 log 文件, davinci.process.log. yyyy-MM-dd 是对应日期的 log 文件,stdout.log,是所有 log 记录的文件.

4.2 错误分类

4.2.1 系统定义的错误

errorCode	错误含义
1	Pdf 生成 thumbnail 发生错误
2	Pdf 转换 tiff 发生错误
3	(CMYK full color) Tiff 转换 jpeg 发生错误
4	(separate color) tiff 转换 jpeg 发生错误
5	Jpg 生成 thumbnail 发生错误
6	Jpg 转换成 RGB jpeg 发生错误
7	Convert 后的 Slice (切片)操作出错,一般是由内存不够造成的 Outofmemory
8	Separate 后的 Slice (切片)操作出错,一般是由内存不够造成的 Outofmemory
9	数据库出现了死锁的情况.

上述这类系统定义的错误:一般情况下都是因为 ghostscript 和 imagemagick 不支持对应的 pdf 文件,或者是因为 ghostscript 和 imagemagick 因为在执行转换命令过程中,添加了某些参数导致的异常.



#### 4.2.2 未定义错误

目前来说一些未定义的错误包括:

- 1.数据库的连接出现异常
- 2.系统的 bug 造成的空指针的错误
- 3.dvview 或者 dvprocess 添加了新的功能后,对 filename.xml 文件添加了新的类型的节点,但是未在代码在同步这个对象的修改,导致了对象序列化或者逆序列化的错误.

对于这一类未定义的错误,就特别需要跟踪查看 log,来进行调试和查看.

## 4.3 解决办法

当一个上传文件因为转换过程出错后,图片的状态也变为错误状态,那么对于 smartflow 而言就不能完成这个 `uploadtask`.这会影响用户的正常工作.那么,如何恢复这个错误,使得上传的图片可以正常转换呢?下面是我将会一步一步的演示恢复的方式.

### 4.3.1 图像转换方面问题

#### 1) 确定出错的命令

这个很简单,可以直接从 `errorCode` 知道发生错误的命令.

命令所在的文件夹

在 `${davinci-process-service}/bin/net/gapsystems/davinci /shell`

`errorCode 1~3` 是发生在: `convertpdf.bat` 文件中

`errorCode 4` 是发生在: `separate.bat` 文件中

`errorCode 5~6` 是发生在: `convertjpg.bat` 文件中

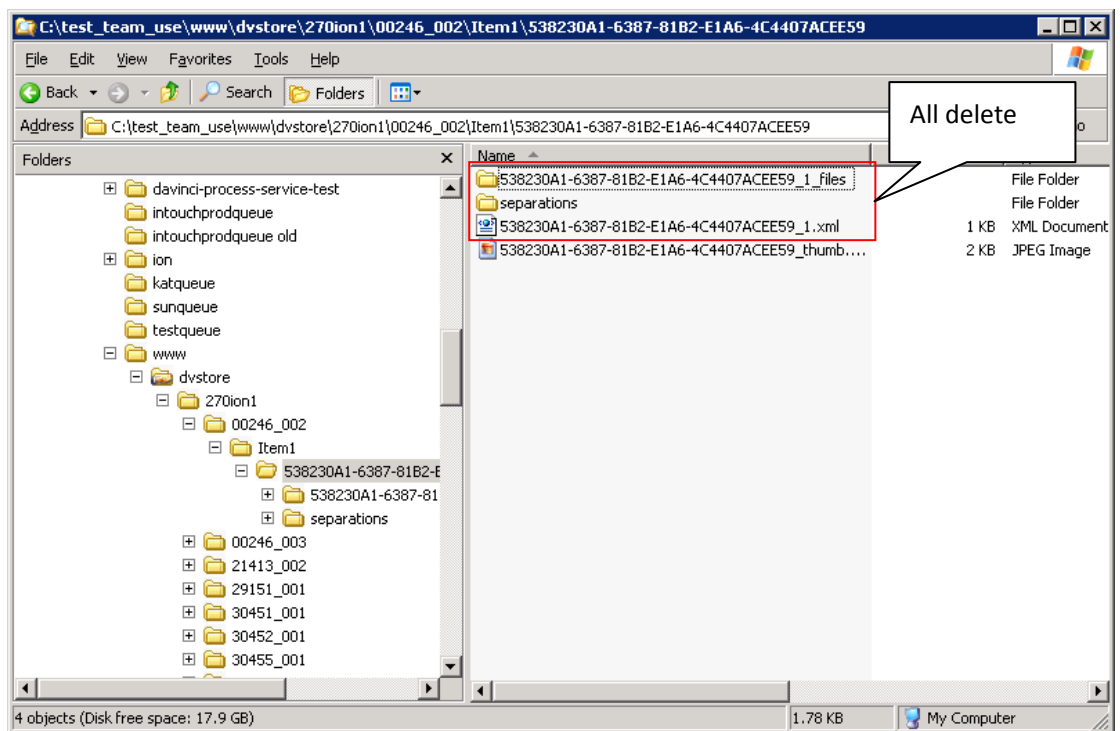
#### 2) 在本地使用控制台命令,使用相同的 `gs` 或者 `imagemagick` 命令进行处理,如果仍然是失败,最大可能是是 `gs` 或者 `imagemagick` 在转换中使用的命令的问题,这时候需要调整命令.(假如与参数无关,可能是跟 `gs` 或 `imagemagick` 的版本有关,更换不同版本尝试).直到在控制台可以使用命令正常处理转换.

举个例子:

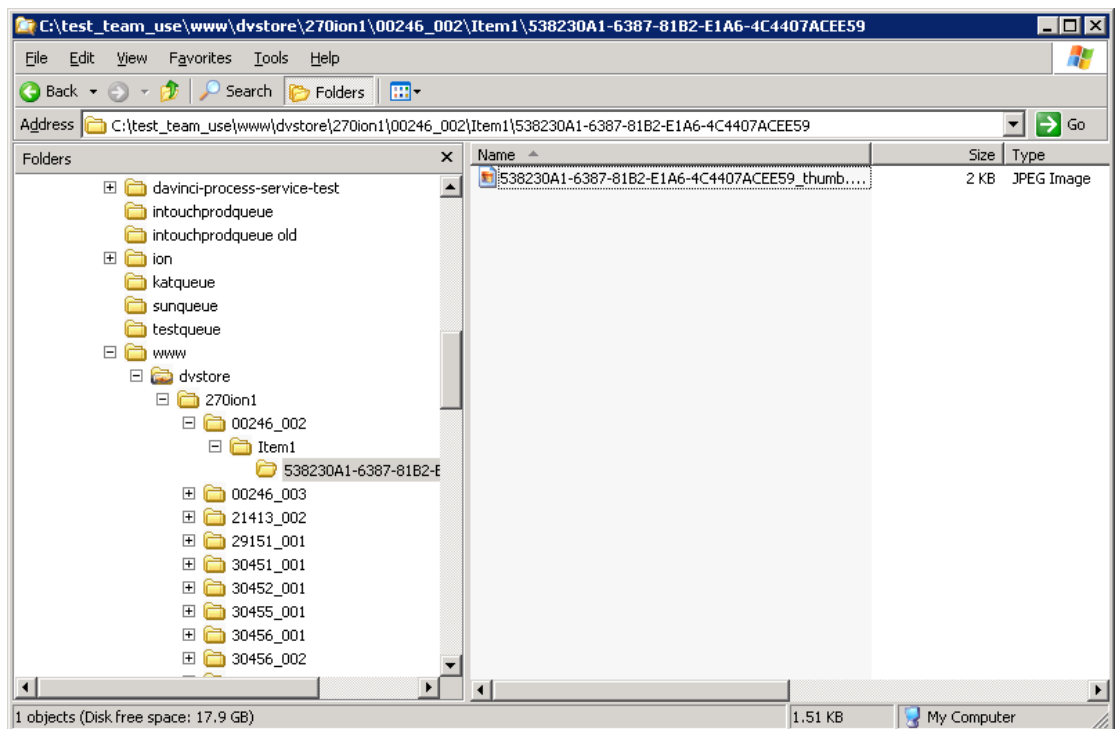
当出现 `errorCode=3` 的错误的时候,主要就是图片的尺寸过大,而 `convert.exe` 命令允许使用的最大内存有限,这个时候为了使得转换能顺利进行,就必须减少 `convert.exe` 命令的内存使用,可以通过降低 `quality` 参数的值. 将 `quality` 从 100 降低到 85 或者更低,直到转换可以正常进行.

#### 3) 删除 `acid` 文件夹中的所有文件除了 `thumbnail`.

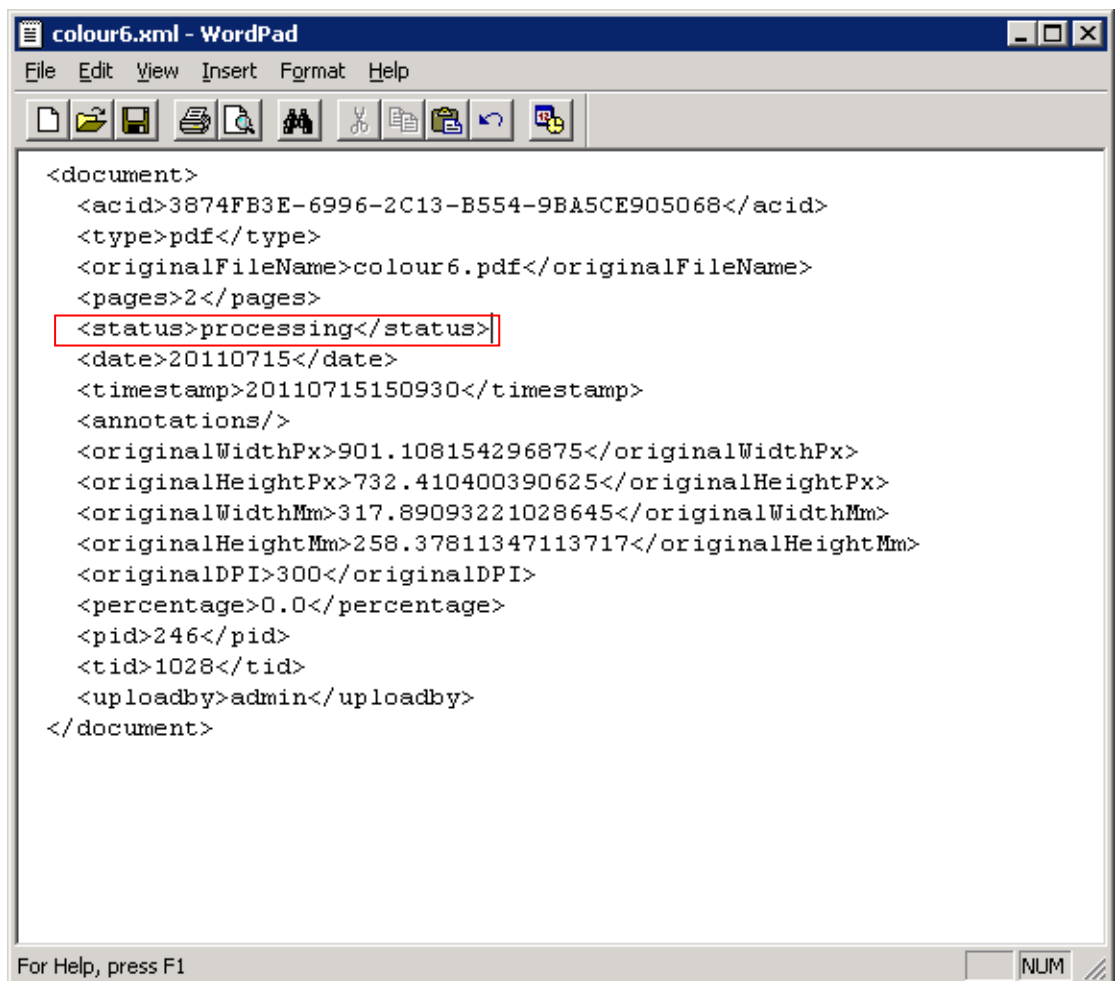
**Before**

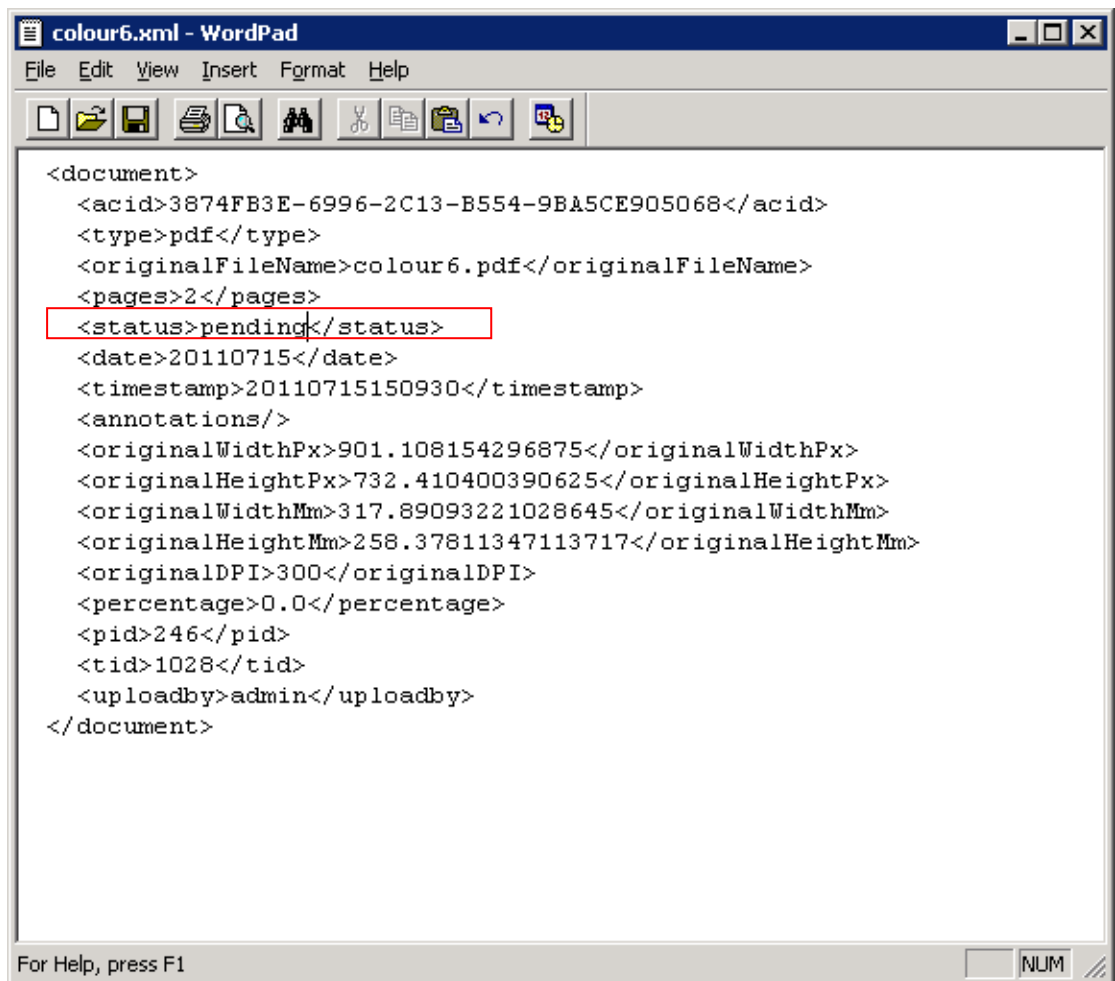


after



- 4) 修改 filename.xml 的状态信息为上传后的初始状态

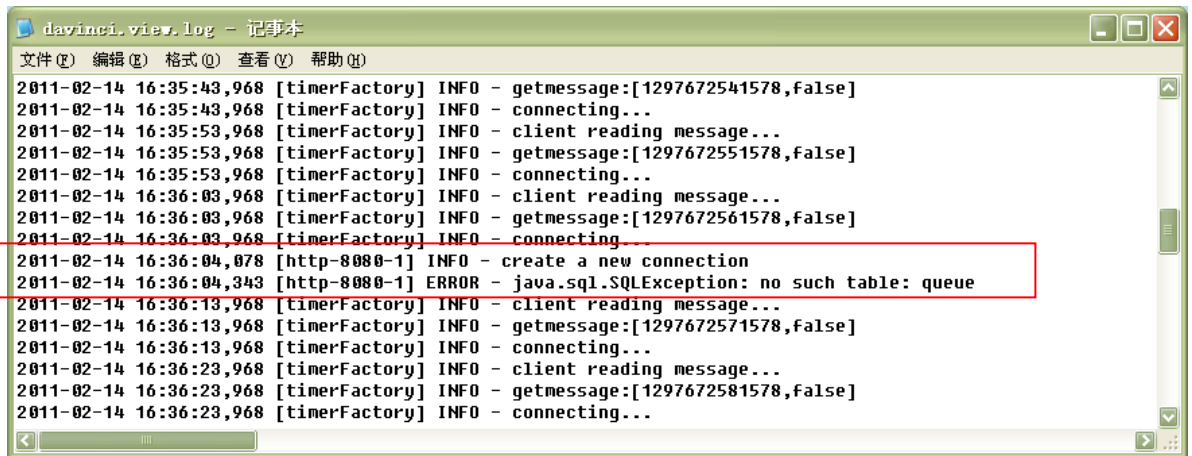




- 5) 将调整后的参数更新到对应的 shell 中(先 backup 一次 shell)
- 6) 在 smartflow 中通过点击连接重新打开客户端(这种方式可以重新调用 createSession 方法,使得 dvprocess 会重新处理这个图片)
- 7) 如果这个参数的调整仅仅是对于这个图片,则需要将 shell 重新回滚到修改之前.

#### 4.3.2 数据库连接出错的情况。

比如在 database-config.xml 文件中设置的 url,username,password 出了不正确,导致 dvprocess 无法正常工作,这种情况下,需要修改 database-config.xml。然后重新启动 dvprocess 服务,下图是出现错误的一种情况。



### 4.3.3 对象序列化的问题.

很多时候我们开发新的功能,需要在 filename.xml 文件中添加新的节点保存信息,或者在 acid-pagenum.xml 文件中添加新的节点.而这两个文件在 dvserver 的后台程序中,都会被转换成 Java 的对象.但是这种转换不是智能的,需要程序员手动的修改该 dvserver 的后台代码才可以实现,否则会出现错误.

错误信息如下:

```
[14:36:13.921] ERROR net.gapsystems.davinci.web.image.ImageService com.thoughtworks
---- Debugging information ----
message           : dependId : dependId
cause-exception    : com.thoughtworks.xstream.mapper.CannotResolveClassException
cause-message     : dependId : dependId
class             : net.gapsystems.davinci.web.beans.Document
required-type     : net.gapsystems.davinci.web.beans.Annotation
line number       : 29
```

如上图所示,我在 filename.xml 中的 annotation 结点下添加了一个子节点叫做 dependId.但是在后台代码中,没有在对应的 Annotation 类中找到有关 dependid 的域.这就会报错.解决办法是在 annotation 类中添加一个域为 dependid.同时要注意域的类型最好跟 filename.xml 文件中的节点类型一致.