

## Infrastructure - Bug #696

### D1Client failing to serialize document stream correctly in multipart form

2010-06-22 22:38 - Chad Berkley

<b>Status:</b>	Closed	<b>Start date:</b>	
<b>Priority:</b>	High	<b>Due date:</b>	
<b>Assignee:</b>	Chad Berkley	<b>% Done:</b>	100%
<b>Category:</b>	Metacat	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>	CCI-0.4	<b>Story Points:</b>	
<b>Milestone:</b>	None		
<b>Product Version:</b>	*		

#### Description

The data stream that comes from D1Client in the multipart message for create is truncated for larger documents. This causes the create to fail since some of the document does not get transferred.

#### History

##### #1 - 2010-06-22 22:38 - Chad Berkley

- Status changed from New to In Progress

##### #2 - 2010-06-23 21:24 - Chad Berkley

- Status changed from In Progress to Closed

this is not a bug in D1Client. This is a bug in metacat. However, I cannot find a real fix for this. I have implemented a work-around. The gist of the problem is that when the [MimeMultipart](#) message is read in [ResourceHandler](#), it iterates through each [BodyPart](#) then calls [BodyPart.getInputStream\(\)](#) to read the actual part. This works fine for parts that are smaller than about 6kB. After about 6kB, it fails to read the rest of the stream.

I confirmed this by reading the stream manually. I can read the entire stream if I just iterate over it, but as soon as I try to read it with [BodyPart.getInputStream\(\)](#), it fails.

The work-around that I implemented just reads the entire multipart message, then parses out the different parts into new streams. The problem with the work-around is that it requires that the stream be put into memory before being split into parts. This does not appear to be the case using MMP, however since it doesn't work, I don't see another choice.

If someone sees a better way to do this, or has any idea why [BodyPart](#) would not return the full stream, please let me know. I did a ton of searching for this and couldn't find anything, so I'm wondering if we're doing something wrong, or if no one is using MMP for larger parts.