Metacat - Bug #2564

escaped "less than" in inlinedata causes invalid eml output

10/12/2006 05:35 PM - Matthew Perry

Status:	Resolved	**Start date:**	10/12/2006
Priority:	Immediate	**Due date:**	
Assignee:	Michael Daigle	**% Done:**	0%
Category:	metacat	**Estimated time:**	0.00 hour
Target version:	1.9	**Spent time:**	0.00 hour
Bugzilla-Id:	2564		

Description

From: inigo san gil <isangil@lternet.edu>
> > My valid EML file

(http://www.cedarcreek.umn.edu/data/emlFiles/pl00e001.xml) has the following line (content):

|Field|Plot|Ntrt|NitrAdd|Date|Taxon|Species |Biomass |Prop <== Labels

However, once harvested, the metacat link to the document http://metacat.lternet.edu/knb/metacat?action=read&qformat=xml&docid=knb-lter-cdr.7901001.2

has that same line slightly changed to: |Field|Plot|Ntrt|NitrAdd|Date|Taxon|Species |Biomass |Prop <== Labels

I noticed that the evil < sign appeared in the inlinedata element. Inline content is handled differently than the rest of the document - it is stored on the file system (the metacat_inline_data folder) rather than in the relational db.

It is interesting to note that a < sign anywhere else in the eml document will be handled correctly (well... it will be displayed as < at least ... see bug 2517). Only in the inlinedata section will this cause the eml output from metacat to be invalid.

This is most likely related to the DocumentImpl.toXml() function, specifically around line 1158 of DocumentImpl.java

Reader reader = Eml200SAXHandler.readInlineDataFromFileSystem(fileName);

History

#1 - 05/22/2008 01:45 PM - Jing Tao

I inserted an eml document into metacat and here is the content before xerces parsing it:

<?xml version="1.0"?><eml:eml
xmlns:eml="eml://ecoinformatics.org/eml-2.0.1"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" packageId="eml.1.1"
system="knb" xsi:schemaLocation="eml://ecoinformatics.org/eml-2.0.1
eml.xsd" scope="system"><dataset scope="document">
<title>Checking > < " ' &</title><creator scope="document">
<individualName>
<surName>Smith</surName>
</individualName>
</creator>
<contact scope="document">
<individualName>
<surName>Jackson</surName>
</individualName>
</contact>
<access authSystem="ldap://ldap.ecoinformatics.org:389/dc=ecoinformatics,dc=org"
order="allowFirst"
scope="document"><allow><principal>public</principal><permission>read</permission></allow></access></dataset></eml:eml>

You can tell in the "title" element they are encoded as numeric character references:
Checking > < " ' &amp

However, after xerces parse the eml the charcters became (xerces automatically did this):
Checking > < " ' &

So we have to add another method - normalize in Metacat. After normalizing, they changed back:
Checking > < " ' &

In "inline" element, same thing will happen - numeric character references were changed to characters. However, we didn't add normalize method to handle characters in "inline" element purposely (Inline element is handled differently to other elements in eml, the content of inline element is stored in a external file rather than database). The reason is that in "inline" element, the data could be a xml segment. If we normailize it, it can be mess. But if we don't normalize it, it can cause the trouble like this.

Do you have any good suggestion?

#2 - 05/22/2008 02:19 PM - Jing Tao

Here is some thought from Duane:

Hi Jing,

Thanks for looking into this further. This seems like a complex problem.

I'm not sure this is a good suggestion, but here is the only solution I can think of:

1. Prior to running Xerces parser, replace all instances of '<' with '<' in inline data. (Same for other character entities.)
2. During insertion, Xerces parser converts '<' back to '<'.
3. Inline data file after insertion contains the original '<'.

Seems kind of crazy to do this, but maybe it would work.

We'll continue to think about this more.

Thanks,
Duane

#3 - 09/17/2008 03:48 PM - Michael Daigle

The document is read from disk in 1.9. This means whatever was submitted will be returned verbatim.

#4 - 03/27/2013 02:20 PM - Redmine Admin

Original Bugzilla ID was 2564