VegBank - Bug #2002

add VegBranch downloads to cache and fix infinite loop problem

03/07/2005 09:51 AM - Michael Lee

Status:	Resolved	**Start date:**	03/07/2005
Priority:	Normal	**Due date:**	
Assignee:	Chad Berkley	**% Done:**	0%
Category:	export	**Estimated time:**	0.00 hour
Target version:	1.1.0	**Spent time:**	0.00 hour
Bugzilla-Id:	2002		

Description

Related issues:

Is duplicate of VegBank - Bug #1653: Plot Downloads incomplete for a lot of p...	**Resolved**	**08/10/2004**
Is duplicate of VegBank - Bug #2132: download into VegBranch	**Resolved**	**06/20/2005**
Blocked by VegBank - Bug #2402: Strategy to update denormalized data, cache, ...	**New**	**04/06/2006**

History

#1 - 03/10/2005 08:00 PM - Michael Lee

text export via jsp is done, just needs to be hooked up to export web features now.

XML formed via these jsp's:
http://aldo.vegbank.org/get/summarycsv/observation/'VB.NP.378.KANSAS'?where=where_place_complex_ac
http://aldo.vegbank.org/get/summarycsv/taxonimportance/'vb.np.378.kansas'?where=where_place_complex_ac&textoutput=true&strata2Show=1
(strata2show is talked about in the jsp- 1,2,3 for strata only, only not-strata,
and all records in taxonimportance)

(pagination is a problem with these. I like ISI's approach of applying
pagination in batches of 500 plots, then downloading each via a separate link)

XSL here:
http://aldo.vegbank.org/vegdocs/xml/util/htmltable2csv.xsl

#2 - 04/01/2005 05:49 PM - Michael Lee

- Bug 1653 has been marked as a duplicate of this bug. ***

#3 - 04/01/2005 05:50 PM - Michael Lee

all downloads should be reworked to use jsp's. Currently only the text download
has jsp's to serve it. Others can follow once we work out the text download.

#4 - 04/18/2005 02:02 PM - Michael Lee

note that the URL's mentioned should NO LONGER HAVE SINGLE QUOTES in them as
URL's have changed.

#5 - 04/21/2005 09:20 AM - Michael Lee

using the jsp's is still way too slow. Using direct output from psql will work
in terms of speed. PMark needs to hook up to the java system to postgresql to
get the data in this way, package it, and then deliver (zipped) to user.

#6 - 05/24/2005 02:33 PM - Michael Lee

The downloads now work. Encoding comes back to bite us, though. We'd like
Latin1 Encodings for Excel, Access, and Text Editors to readily know what to do
with strange characters.

#7 - 07/21/2005 06:26 PM - Michael Lee

the .csv download is sweet! Need to use a cache to store the XML and maybe also
VegBranch .csv. This way, we can easily assemble plots by stacking XML snippets

on top of one another. Then zip and send to user.

#8 - 07/22/2005 08:25 AM - Michael Lee

- Bug 2132 has been marked as a duplicate of this bug. ***

#9 - 04/06/2006 07:29 PM - Michael Lee

There is a method on all beans "ToXML" that creates an XML representation of the bean. This needs to be TESTED to see if it works ok, and if not, it needs to be fixed. Then we need to decide where these should be stored (filesystem or database). Then when users request an XML download, these cached XML snippets copied and pasted into an XML doc with the appropriate root element and atts.

see:
http://vegbank.org/xml for all about our XML

#10 - 07/17/2006 12:27 PM - Chad Berkley

So far I have updated the LoadTreetoDatabase.java so that when a new xml document is uploaded and ingested into vegbank, the beans are created and serialized to a new database table, dba_xmlcache. This table contains only two columns (accessioncode (String), xml (bytea)) and stores the xml serialization of the object with the provided accessioncode.

Upon download, the XMLUtil.java class has been modified to first check to see if the xml is cached in the database table. If the xml is there, it pulls the xml from the table instead of calling VBBean.toXML(), which is very slow.

To bootstrap existing systems, I've written a utility to XMLUtil.java that searches the database for objects in observation, plantconcept, commconcept, project and party which can be made into beans. when one is found, the bean is serialized into the xml field of the dba_xmlcache table for use later in the download step.

#11 - 08/14/2006 04:39 PM - Michael Lee

So when does the bootstrapper function? Does it search the DB periodically? What if we update a plot- how/does the XML Cache get updated?

(In reply to comment #10)

> So far I have updated the LoadTreetoDatabase.java so that when a new xml document is uploaded and ingested into vegbank, the beans are created and serialized to a new database table, dba_xmlcache. This table contains only two columns (accessioncode (String), xml (bytea)) and stores the xml serialization of the object with the provided accessioncode.
>
> Upon download, the XMLUtil.java class has been modified to first check to see if the xml is cached in the database table. If the xml is there, it pulls the xml from the table instead of calling VBBean.toXML(), which is very slow.
>
> To bootstrap existing systems, I've written a utility to XMLUtil.java that searches the database for objects in observation, plantconcept, commconcept, project and party which can be made into beans. when one is found, the bean is serialized into the xml field of the dba_xmlcache table for use later in the download step.

#12 - 08/24/2006 11:43 AM - Chad Berkley

The xml cache is now working as far as I can tell. xml cache items get created upon upload to the system. If an item is downloaded and is not already in the cache, it is put there also. If a plot get changed (and thus re-uploaded), the cache entry will be recreated for the new accession number revision.

#13 - 08/26/2006 03:40 PM - Michael Lee

We have a slight problem in the caching system as plots with soilTaxon Records don't cache because of an infinite loop somewhere (we think) because there is an outOfMemory error.

#14 - 08/26/2006 04:04 PM - Michael Lee

Also, if we have time, it would be great to add a new field to dba_xmlCache called vegbranchCSV which is just a styled representation of the XML, using the vegbank/src/xsl/vegbank2vegbranchcsv.xsl stylesheet. Then the VegBranch download option could be turned on.

#15 - 08/30/2006 10:12 AM - Chad Berkley

I think I've finally tracked down what's going on with the soilTaxon infinite recursion problem. SoilTaxon has a recursive foreign key back to itself (SOILPARENT_ID). This allows soilTaxons to have a parent that could eventually lead back to itself causing an infinite loop. In fact, this is the case for most of the soilTaxon records. When the XML is serialized, the DBModelBeanReader attempts to follow this link only to eventually run out of memory. I'm not sure what the proper fix is for this at this point. I think the model needs to be changed. For a stop-gap for the release, I'm going to

attempt to limit the recursion to 20 levels and hopefully that will stop it in time and still preserve enough information. Because of the dynamic nature of this code base, i don't think I can totally alter the recursive routine (getObjectFromDB lines 602-612) without majorly hosing other bean classes.

(In reply to comment #13)

> We have a slight problem in the caching system as plots with soilTaxon Records don't cache because of an infinite loop somewhere (we think) because there is an outOfMemory error.

#16 - 08/30/2006 10:47 AM - Chad Berkley

I think I've fixed the infinite loop problem within soilTaxon. There is a marker tag in the db_model_vegbank.xml file to indicate to the xsl transform if the field is supposed to be recursed on or not. soiltaxonparent_id was not indicated to be recursive even though it was. I changed the tag, re-generated the java and it seems to work now. Michael should try it and make sure the xml looks right.

#17 - 09/06/2006 05:56 PM - Michael Lee

VegBranch downloads now work on aldo. Need to stress test a bit. If plots aren't in cache, it does time out on the browser.

#18 - 09/07/2006 04:32 PM - Michael Lee

VegBranch loads now work, but not tons of them

#19 - 03/27/2013 02:18 PM - Redmine Admin

Original Bugzilla ID was 2002