Infrastructure - Task #6014

Story # 5523 (Closed): Run Tidy after Split Brain

Verify inconsistent records identified by tidy have not been updated

2014-07-30 21:22 - Robert Waltz

Status:	Closed	**Start date:**	2014-07-30
Priority:	Normal	**Due date:**	
Assignee:	Robert Waltz	**% Done:**	100%
Category:	Environment.Production	**Estimated time:**	0.00 hour
Target version:	2014.30-Block.4.3		
Milestone:	CCI-1.3	**Story Points:**	
Product Version:	*		

Description

verify that the inconsistent records on a machine identified by tidy have not been updated
* compare dateSysMetadataModified and serialVersion of records identified on production machine to tidy db entries

History

#1 - 2014-07-30 21:24 - Robert Waltz

- Category changed from d1_replication_auditor to Environment.Production

- Subject changed from verify inconsistent records identified by tidy have not been updated to Verify inconsistent records identified by tidy have not been updated

#2 - 2014-08-06 17:37 - Robert Waltz

- Status changed from New to In Progress

On unm run against the tidy database

To verify ucsb controlled records to change on ORC

```
su postgres -c "psql -d d1-tidy  -c \"select guid, \
serial_version, TO_CHAR(date_modified, 'YYYY-MM-DD\\"T\\"HH24:MI:SS.MSTZ') \
from ucsb_systemmetadata usm \
where exists (
select cr1.pid \
from change_record cr1, change_record cr2 \
where usm.guid = cr1.pid and \
cr1.id = cr2.id and \
cr1.runid = cr2.runid and \
coalesce(cr1.cnucsb,'nvl') like cr2.resulting and \
coalesce(cr1.cnorc, 'nvl') not like cr2.resulting and \
cr1.runid = 2 )\"" > ucsb_to_orc_verification.in
```

To verify ucsb controlled records to change on UNM

```
su postgres -c "psql -d d1-tidy  -c \"select guid, \
serial_version, TO_CHAR(date_modified, 'YYYY-MM-DD\\"T\\"HH24:MI:SS.MSTZ') \
from ucsb_systemmetadata usm \
where exists (
select cr1.pid \
from change_record cr1, change_record cr2 \
where usm.guid = cr1.pid and \
cr1.id = cr2.id and \
cr1.runid = cr2.runid and \
coalesce(cr1.cnucsb,'nvl') like cr2.resulting and \
coalesce(cr1.cnunm, 'nvl') not like cr2.resulting and \
cr1.runid = 2 )\"" > ucsb_to_unm_verification.in
```

To verify orc and unm controlled records to change on UCSB

```
su postgres -c "psql -d d1-tidy  -c \"select guid, \
serial_version, TO_CHAR(date_modified, 'YYYY-MM-DD\\"T\\"HH24:MI:SS.MSTZ') \
from orc_systemmetadata usm \
where exists (
select cr1.pid \
from change_record cr1, change_record cr2 \
where usm.guid = cr1.pid and \
cr1.id = cr2.id and \
cr1.runid = cr2.runid and \
coalesce(cr1.cnorc, 'nvl') <> 'null' and \
coalesce(cr1.cnunm, 'nvl') <> 'null' and \
coalesce(cr1.cnorc, 'nvl') like cr2.resulting and \
coalesce(cr1.cnunm, 'nvl') like cr2.resulting and \
coalesce(cr1.cnucsb, 'nvl') not like cr2.resulting  and \
cr1.runid = 2 )\"" > to_ucsb_verification.in
```

---

delete first and last 2 lines of [orc|ucsb]_verification.in

results in file with the following lines similar to

knb-lter-fce.1108.2 | 2 | 2014-05-15 22:26:16.088

translate file into csv

perl -i -pe "s/\s([\s]) \s+/\s([^ad]) \s+|\s(.+)$/\1 \2 \3/" [orc|ucsb]_verification.in

read csv file and query database for correct entries, note any query that is 0.

Test records

```
su postgres -c "psql -d metacat  -c \"select guid, serial_version, TO_CHAR(date_modified, 'YYYY-MM-DD\\"T\\"HH24:MI:SS') \
from systemmetadata \
```

LIMIT 10\"" > sysmeta_verification

install on target CNs

apt-get install libdbi-perl libdbd-pg-perl libio-prompt-perl

---

save the following perl script and
run with appropriate csv files generated above

on the appropriate cns

#!/usr/bin/perl

#####

file input must be a csv file that conforms to the pattern

pidserial_versiondate

#####

The program will execute a query against a metacat database's

table systemMetadata attempting to determine if any of the

rows in the database are different from the rows in the

CSV file

#####
#####

```
use strict;
use Carp;
use DBI;
use Getopt::Long();
use IO::Prompt;
```

#####

GLOBAL VARIABLES

```
#####

my $dbh;
my $dbname = "metacat";
my $host = "localhost";
my $username = "metacat";

my $metacat_select = "select count(*) AS COUNT from systemMetadata
where guid = ? and
serial_version = ? and
date_modified = to_timestamp(?, 'YYYY-MM-DD\"T\"HH24:MI:SS.MS')";

my $metacat_select_sth;

my $pid_column_key = 'PID';
my $serial_version_column_key = "SERIAL_VERSION";
my $date_modified_column_key = "DATE_MODIFIED";

sub usage {
my $message = $_[0];
if (defined $message && length $message) {
$message .= "\n"
unless $message =~ /\n$/;
}

my $command = $0;
$command =~ s#.*/##;

print STDERR (
$message,
"usage: $command [--help] --file /full/path/of/csv/file.cvs\n",
"\n\tfile input must be a csv file that conforms to the pattern\n",
"\tpidserial_versiondate\n"
);

die("\n");
}

#####
```

read a cvs file with the format

pidserial_versiondate

return a reference to an array of hashes with the format

@output = {

'PID' => $pid,

'SERIAL_VERSION' => $serial_version,

'DATE_MODIFIED' => $date_modified);

```
#####
sub readCVS
{
my $file = shift;
my @output_array = ();
my $csv_fh;
open ($csv_fh,  "< $file") or die "Cannot open $file: $!\n";
while (<$csv_fh>)
{
chop;
my ($pid,$serial_version,$date_modified) = split(" ", $_);
my %record_hash = ($pid_column_key => $pid,
$serial_version_column_key => $serial_version,
$date_modified_column_key => $date_modified);
push (@output_array, \%record_hash);

}



close ($csv_fh);
return \@output_array
}
```

#####

verify that each row in the CSV file has a corresponding

row in the database. Return 1 (true) if the row is verified,

otherwise return 0 (false)

```
#####
sub verifyMetacatRow
{
my ($pid, $serial_version, $date_modified) = @_;
$metacat_select_sth->bind_param(1,$pid);
$metacat_select_sth->bind_param(2,$serial_version);
$metacat_select_sth->bind_param(3,$date_modified);
$metacat_select_sth->execute();
my $metacat_select_hash_ref = $metacat_select_sth->fetchrow_hashref('NAME_lc');
if ( defined($metacat_select_hash_ref->{count}) &&
($metacat_select_hash_ref->{count} == 1) )
{
return 1;
}
return 0;
}

main: {

my $file;
my $help;
Getopt::Long::GetOptions(
'file=s' => \$file,
'help!' => \$help
) or usage("Invalid commmand line options.");

usage() if (defined $help);

usage("The full path of a CSV file (space separated) must be specified.")
unless ((defined $file) && (-e $file)) ;

my $password = prompt("Enter Metacat Postgresql DB password: ", -e => '*');
chomp($password);

my $rows_to_validate = readCVS($file);

my $dbh = DBI->connect("DBI:Pg:dbname=${dbname};host=${host}", "$username", "$password", {'RaiseError' => 1, 'PrintError' => 0});
$metacat_select_sth = $dbh->prepare($metacat_select);
foreach my $row (@{$rows_to_validate})
{
print "FAILURE: " . $row->{$pid_column_key} . "\n" unless (verifyMetacatRow($row->{$pid_column_key}, $row->{$serial_version_column_key},
$row->{$date_modified_column_key}));
}
```

}

#3 - 2014-08-08 03:22 - Robert Waltz

Verification results:

The query for cn-unm-1 inconsistencies resulted in 2 pids from the TIDY database represented in the ucsb_to_unm_verification.in file.

After running the perl script on cn-unm-1 with all three *_verification.in files, 3 pids showed inconsistence.

It appears that the interaction between cn-orc-1.dataone.org and cn-unm-1.dataone.org as a separate 'read-only' cluster from cn-ucsb-1 resulted a further corruption of cn-unm-1's database.

The query results for unm inconsistencies resulted in 82 pids from the TIDY database represented in ucsb_to_orc_verification.in

After running the perl script on cn-unm-1 with all three *_verification.in files, 82 pids showed inconsistencies.

The query results for ucsb inconsistencies resulted in 971 pids from the TIDY database represented in to_ucsb_verification.in

After running the perl script on cn-ucsb-1 with all three *_verification.in files, 0 pids showed inconsistencies.

I created a second query:

```
su postgres -c "psql -d d1-tidy  -c \"select count(*) \
from ucsb_systemmetadata ucsb, orc_systemmetadata orc \
where ucsb.guid = orc.guid and \
ucsb.serial_version = orc.serial_version and \
ucsb.date_modified = orc.date_modified and \
exists ( \
select cr1.pid \
from change_record cr1, change_record cr2 \
where ucsb.guid = cr1.pid and \
orc.guid = cr1.pid and \
cr1.id = cr2.id and \
cr1.runid = 2 and \
cr1.runid = cr2.runid and \
cr1.cnorc <> 'null' and \
cr1.cnunm <> 'null' and \
cr1.cnorc like cr2.resulting and \
cr1.cnunm like cr2.resulting and \
cr1.cnucsb is null)\""
```

This query shows that though the tables have 971 differing systemMedata the guid, serial_version and date_modified of those records are exactly the same.

All 971 records showed a difference in the xml_access table in which allowed all records public read while ucsb had no records at all. All records came from CDL.

Creating a shell script to query CDL for the systemMetadata pids, 516 systemMetadata records were found to have accessPolicies of public read, while 554 were 'NotFound' and 1 was a proxy error returned as a 500 level error.

From these results, I can only assume that the 971 pids on cn-ucsb-1 are incorrect and should be updated according to the records on cn-orc-1.

#4 - 2014-08-08 19:33 - Robert Waltz

- translation missing: en.field_remaining_hours set to 0.0

- Status changed from In Progress to Closed

#5 - 2014-08-21 01:20 - Robert Waltz

- Estimated time set to 0.00

*- Product Version changed from 1.3.0 to **