Mulgara - Bug #89

Track down problems with BackupOperation

03/20/2008 01:36 AM - Andrae Muys -

Status:	Closed	**Start date:**	
Priority:	Immediate	**Due date:**	
Assignee:	Paula Gearon	**% Done:**	0%
Category:	Mulgara	**Estimated time:**	0.00 hour
Target version:			
Resolution:	fixed		

Description

Viewpoint is reporting duplicate StringPool entries, as well as the introduction of blank-nodes in backups.

Topaz is reporting NPE's when trying to run backups on the committed-phase.

We need to identify the bug, and fix it.

Associated revisions

Revision 699 - 03/20/2008 07:44 AM - Andrae Muys -

refs #89

Find and fix the backup bug.

Revision 700 - 03/20/2008 11:55 AM - Andrae Muys -

refs #89

This fixes the Backup issues, but in the process makes the backups incompatible with Restore.

The issue is that the StringPoolSession does some absolute/relative mapping on URI's that isn't done when working directly against the StringPool. This operation has to be reversed in the Restore, and as Restore also by passes the transaction (this isn't an issue as it **has** to hold the write-lock anyway) this reversal isn't happening.

Note that fixing Restore to do the reversal will be incompatible with any prior backups, so is unacceptable without upgrading the backup version number.

This suggests that we need to provide alternative functions on StringPoolSession that are dedicated to providing mapping free operations.

Revision 709 - 03/25/2008 08:47 AM - Andrae Muys -

refs #89

This patch does pass 100% of the tests, and remains faithful to the original backup formats. There are additional cleanup tasks I would like to see done before this makes its way into trunk - but anyone who needs an urgent patch to the backup operation can use this.

Revision 710 - 03/25/2008 09:40 AM - Andrae Muys -

refs #89

merge -r 699:709 ../../branches/mgr-89-backup

This merge fixes the problem with backup obtaining the wrong phase.

The core of the fix is in BackupOperation, BackupRestoreSession, and StringPoolSession.

As discussed on the wiki - this bug was related to a failure by BackupOperation to use the phase provided it by the enclosing transaction, instead bypassing the transaction and obtaining its own reference to the current-phase. This is

clearly in error as this is a read-only operation and so should not require access to this phase - and moreover this bypass also bypassed the write-lock, meaning the phase was not stable, leading to errors.

Revision 713 - 03/26/2008 05:15 AM - Andrae Muys -

refs #89

Removed the ResolverSystemFactory as an argument to Operation::execute(). This interface could not be used safely, so removing it was important.

Revision 714 - 03/26/2008 05:49 AM - Andrae Muys -

refs #89

svn merge -r 712:713 ../branches/mgr-89-backup

History

#1 - 03/20/2008 02:41 AM - Paula Gearon

Viewpoint is showing strings out of order when iterating the string pool. The reappearance of the same string out of order may be due to the erroneous reuse of a gNode. The StringPool cache would end up retrieving the old string, as it would expect this mapping to stay the same.

#2 - 03/20/2008 07:44 AM - Andrae Muys -

(In r699) refs #89

Find and fix the backup bug.

#3 - 03/20/2008 08:07 AM - ronald -

Note that the Topaz (PlosONE) NPE is occurring on mulgara 1.0 (with patches r440, r389, r388, r258, r254, r233, r217, r193, r212, r205, r183, r175, r170, r169, r178, r179, r184 - but none of these should be affecting this).

#4 - 03/20/2008 08:11 AM - Andrae Muys -

Looks like BackupOperation is bypassing the phase handed it by the OperationContext, and instead is trying to obtain its own copy from the StringPoolFactory. This means it may be ending up on the currentPhase, which - if it is currently undergoing a write - may not be stable.

This might explain both the NPE, the erroneous blank-nodes. Not sure if this can explain the double-entries Viewpoint is seeing - Paul might be able to comment on this better than I.

#5 - 03/20/2008 10:26 AM - Andrae Muys -

The key section of the Topaz NPE stacktrace is:

```
Caused by: java.lang.NullPointerException
    at org.mulgara.store.xa.AVLNode.release(AVLNode.java:1040)
    at
org.mulgara.store.stringpool.xa.XAStringPoolImpl$Phase$GNodeTuplesImpl.c
lose(XAStringPoolImpl.java:2728)
    at
org.mulgara.resolver.BackupOperation.backupDatabase(BackupOperation.java
:196)
    at
org.mulgara.resolver.BackupOperation.execute(BackupOperation.java:145)
```

The key line in BackupOperation.java which appears to be the culprit is line 143:

```
        [[StringPool]] stringPool =
            resolverSessionFactory.getPersistentStringPool();
```

Note that this bypasses the phase associated with the transaction and obtains a direct reference to the currentPhase - which is a bug.

If this is done while holding the write-lock this won't be a problem as we will have obtained independently the same phase that was associated with the transaction. However if we do this under a read-only transaction - we have a problem. Given the symptoms described by Topaz I expect there was a concurrent writing transaction in progress when this line was called that called prepare() or rollback() prior to the backup-operation reaching line 199 where it tries to release its hold on the Phase.

The problem with this is that a prepare() or rollback() will force the release() of the currentPhase, invalidating the token, and resulting in a NPE when our bypassed-phase attempts its own release at line BackupOperation:199.

The fix is to not bypass the transaction initially, and then none of this will be an issue.

#6 - 03/20/2008 11:55 AM - Andrae Muys -

(In r700) refs #89

This fixes the Backup issues, but in the process makes the backups incompatible with Restore.

The issue is that the StringPoolSession does some absolute/relative mapping on URI's that isn't done when working directly against the StringPool. This operation has to be reversed in the Restore, and as Restore also by passes the transaction (this isn't an issue as it **has** to hold the write-lock anyway) this reversal isn't happening.

Note that fixing Restore to do the reversal will be incompatible with any prior backups, so is unacceptable without upgrading the backup version number.

This suggests that we need to provide alternative functions on StringPoolSession that are dedicated to providing mapping free operations.

#7 - 03/25/2008 08:47 AM - Andrae Muys -

(In r709) refs #89

This patch does pass 100% of the tests, and remains faithful to the original backup formats. There are additional cleanup tasks I would like to see done before this makes its way into trunk - but anyone who needs an urgent patch to the backup operation can use this.

#8 - 03/25/2008 09:40 AM - Andrae Muys -

(In r710) refs #89

merge -r 699:709 ../../branches/mgr-89-backup

This merge fixes the problem with backup obtaining the wrong phase.

The core of the fix is in BackupOperation, BackupRestoreSession, and StringPoolSession.

As discussed on the wiki - this bug was related to a failure by BackupOperation to use the phase provided it by the enclosing transaction, instead bypassing the transaction and obtaining its own reference to the current-phase. This is clearly in error as this is a read-only operation and so should not require access to this phase - and moreover this bypass also bypassed the write-lock, meaning the phase was not stable, leading to errors.

#9 - 03/26/2008 01:14 AM - Andrae Muys -

- Status changed from New to Closed

- Resolution set to fixed

#10 - 03/26/2008 05:15 AM - Andrae Muys -

(In r713) refs #89

Removed the ResolverSystemFactory as an argument to Operation::execute(). This interface could not be used safely, so removing it was important.

#11 - 03/26/2008 05:49 AM - Andrae Muys -

(In r714) refs #89

svn merge -r 712:713 ../branches/mgr-89-backup