



**ITUG
SUMMIT 2003**

*The flagship event for NonStop
experience, solutions & influence.*

Performance characteristics of Audited and Nonaudited files

Bob Sardiga

ITUG

25th

ANNIVERSARY


I T U G
*The International HP NonStop Users Group
independent, not-for-profit, user run*

Background

- **Chicago Mercantile Exchange was investigating RDF replication**
- **They have application level replication now**
- **Mix of audited and unaudited application files.**
- **RDF can only see audited files.**
- **We needed a way to pick up the nonaudited files**
- **We installed & used NonStop AutoTMF on test system with defaults**
- **Completed installation and evaluation.**

Background (contd.)

- **Noted that one program ran much faster (almost 10x)**
- **Decided to look into it further**
- **The application was difficult to set up and rerun**
- **We set up a test environment on an hp internal system**

Test Environment

- **Dedicated S-72000**
- **4 processor**
- **Master Audit Trail only**
- **In the CME application the only change was that the files were made audited for RDF replication**
- **We isolated that part of the CME transaction**

The test files

- **Key-sequenced**
- **Original file had large key and alternate key & large record**
- **We used 25 byte key.**
- **1,000 byte record**
- **Preloaded with 10,000 records**

Testing

- **Tested 3 cases**
 1. Unprepared program compiled only
 2. Prepared (AutoTMF) but file not audited
 3. Prepared and file audited
- **Negligible CPU cost to prepare object and not audit the file. This verified vendor info**
- **Skipped first case for subsequent runs**
- **Tested using case 2 & 3**

Test 1

- **Insert 10,000 new records**
- **Inserts evenly placed over entire file**
- **Not at end or hot spot**
- **This caused 2,500 block splits**
- **Run with file nonaudited**
- **Run with file audited**

Test 1 Insert 10,000 Records

	Not audited insert 10K	audited insert 10K	ratio A/NA
start	20:16:34	20:31:57	
stop	20:20:16	20:32:56	
lapsed time	0:03:42	0:00:59	
lapsed time sec	222	59	0.27
cpu time sec	2.27	2.59	1.14
prog I-O	12,510	12,510	1.00
\$D0201	23,819	4,553	0.19
\$audit I-O	16	2,508	156.75
total I-O	23,835	7,061	0.30
Block splits	4,999	0	0.00
Begin trans	0	313	

Test 2

- **2 copies of the file on different volumes**
- **Randomly update both files**
- **Read with lock, rewrite, unlock both records**
- **Simulate multiple file updates and record locking**
- **Run with nonaudited and audited files**

Test 2 update 20K records

	not audited update	audited update	ratio A/NA
start	22:14:38	22:10:23	
stop	22:18:08	22:12:17	
lapsed time	0:03:30	0:01:54	
lapsed time sec	210	114	0.54
cpu time sec	6.76	8.78	1.30
prog I-O	50,032	50,032	1.00
\$D0201	12,513	4,891	0.39
\$D0103	12,509	5,712	0.46
\$audit I-O	9	4,085	453.89
total I-O	25,031	14,688	0.59
Block splits			
Begin trans	0	625.00	

Test 3

Auditcompress

- **Auditcompress can reduce the amount of audit generated**
- **Only the changed columns, 'fields' get logged**
- **In test 3 we updated a timestamp in 20K records**

Test 3

compressed audit

	update without compress ed audit	update with compress ed audit	ratio compressed/ uncompress ed	
start	13:27:19	13:20:03		
stop	13:28:54	13:21:19		
lapsed time	0:01:35	0:01:16		
lapsed time sec	98	68	0.69	less
cpu time sec	8.62	8.58	0.9954	even
prog I-O	20000	20000	1.00	
\$D0201	1126	930	0.83	
\$D0103	6223	5776	0.93	
\$audit I-O	4083	658	0.16	much less
total I-O	11432	7364	0.64	less
Block splits				
Begin trans	625	625		

Compressed audit

- **TMF logs only changed info (plus header)**
- **In the test a timestamp was changed (2% of 1000 byte record)**
- **It did significantly reduce amount of audit**
- **As a larger portion of the record is updated improvement will be less**
- **May not be compatible with all replicators**
- **Low cost**
- **Did not test recovery**

Findings

- **Disc process I-O's greatly reduced**
- **Audit disk writes increase**
- **Total I-O's reduced**
- **Elapsed time reduced (updates)**
- **Elapsed time reduced significantly for (inserts w/block splits)**
- **Some increase in CPU time**
- **Compressed audit can be additional improvement**

Block Splits

- **Likely with Large record size and Large primary key and random inserts (CME)**
- **Inserts to end are not a problem**
- **You may also cause splits in the alternate key files (CME)**
- **Block can hold only a few records so if we do multiple inserts we will force block splitting**
- **Most effectively done by TMF because audit trail can be used to recover**
- **If file is not audited must be done by disk process**

Block Splits (contd.)

- **Disk process saves the full block**
- **Creates 1 or 2 new blocks**
- **Inserts new record in a block and writes updated blocks to disk**
- **Releases or rewrites the original (full) block**

How to get to TMF aware programs

- **Will require development effort (-)**
- **May be legacy application that will be replaced (not worth the cost) (-)**
- **You can decide what is a business transaction (+)**

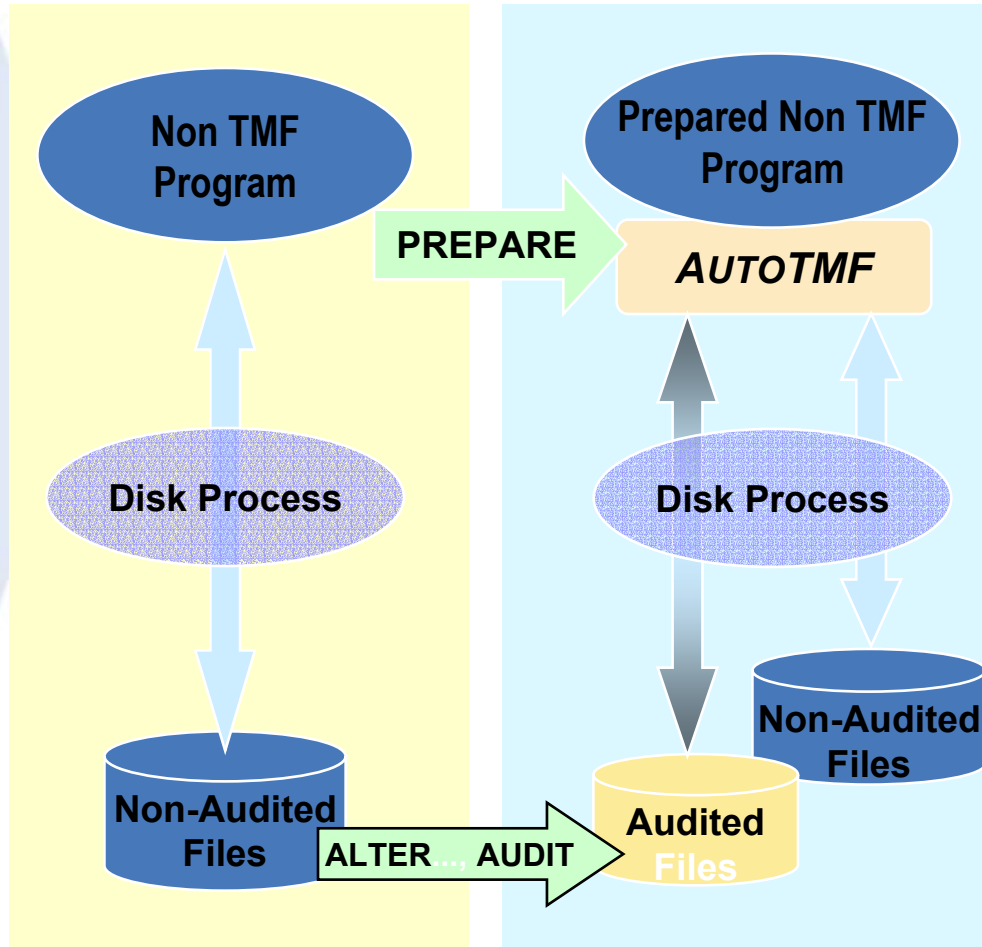
TMF aware programs

- **Development working on other projects (-)**
- **Not enough cycles (-)**
- **Looked at “automated” solution**

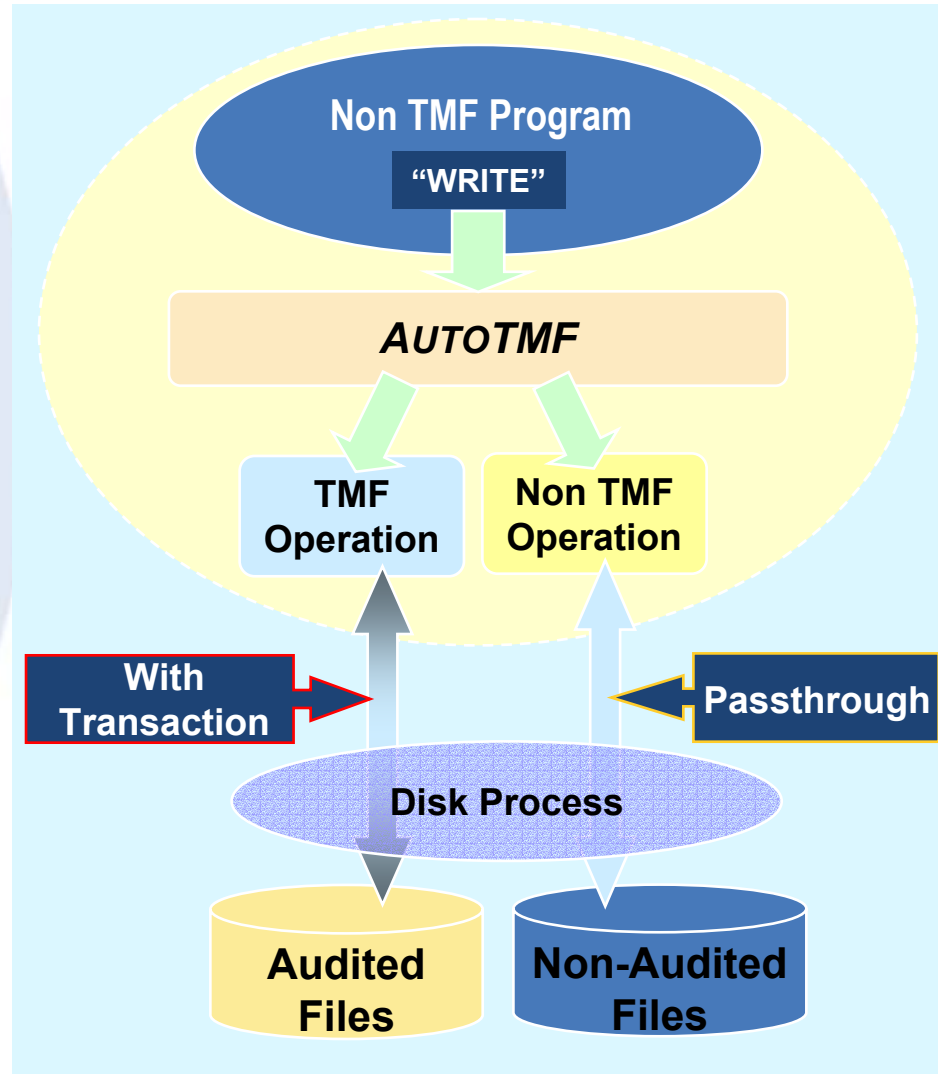
Implementing NonStop AutoTMF

- **Source Program is unchanged**
- **Object program is “prepared”**
 - **Changed to use Auto TMF intercept library**
- **At run time**
 - **If target of I-O is not audited no action taken (pass-through)**
 - **If target is audited and no TMF transaction in effect then start one.**
- **You are not required to audit all files**
- **Many options available (took defaults)**

Migration



AutoTMF runtime



Architecture

ITUG
SUMMIT 2003

ESCORT CI

Monitor Management
(START, STOP, STATUS)

\$ZESC
(Monitor)

Config
Cache

Configuration

Configuration
Database

Configuration

Auditing

Preparation

Audited
Files

Audited
Files

Audited
Files

Application

AUTOTMF
Runtime

Application

AUTOTMF
Runtime

Application

AUTOTMF
Runtime

"BEGINTRANSACTION"
"ENDTRANSACTION"

T
M
F

ITUG

The International HP NonStop Users Group
independent, not-for-profit, user run

Summary

- **Get benefits of TMF**
 - **On-line dumps**
 - **File recovery**
- **Not business transaction control (AutoTMF is not aware of a business transaction)**
- **Can now use Replicators that use TMF audit trails (RDF, Shadowbase, E/R)**
- **Negligible performance penalty on unaudited files**

Summary

- Performance bonus on audited files
- Consider especially if doing inserts
- Moving I-O's from a data disk to the Audit disk may help you if the data disk is overloaded. May hurt you also!
- Very easy to implement
- Default settings worked for CME testing vendor claims they will work for 90% of files
- Can select custom options by file

- Thanks for attending
- Questions?
 - Bob Sardiga
 - BobSardiga@comcast.net
 - 630.243.1077