PUG CHALLENGE EXCHANGE AMERICAS

What's New with VSTs

Rich Banville

June 5, 2017

VSTs: What and Why

- Virtual System Tables
  - Meta-schema definition but no stored record data
- Useful for DBA
  - Can monitor activity
    - Detect, analyze and fix issues in running system
    - Monitor for maintenance needs
    - **Local** or **remote** monitoring
  - Can change parameters
- Useful for applications
  - Embed within the application
    - Control private buffers for example
    - Monitor activity for query performance

VST Relationships – Connection Perspective

_AreaExtent
_Area-number
_Extent-path

_Area
_Area-number
_Area-Recid

_StorageObject
_Object-number
_Object-type
_Area-Number

_ActIOFile
_IOFile-FileName

_AreaStatus
_Area-Name

_TableStat
_TableStat-Id

_File
_File-num

_Index
_File-recid
_Idx-num

_IndexStat
_IndexStat-Id

_UserLock
_UserLock-Id
_UserLock-Usr
_UserLock-Table

_UserTableStat
_UserTableStat-id
_UserTableStat-Conn
_UserTableStat-num

_UserIndexStat
_UserIndexStat-id
_UserIndexStat-Conn
_UserIndexStat-num

_UserIO
_UserIO-Id
_UserIO-Usr

_MyConn
_MyConn-Id
_MyConn-UserId

_Connect
_Connect-Id
_Connect-Usr
_Connect-Server
_Connect-TransId

_Server
_Server-Id
_Server-Num

_LockReq
_LockReq-Id
_LockReq-uNum

_Lock
_Lock-Usr
_Lock-TransId

_Trans
_Trans-Num
_Trans-Usrnum

VSI Common Questions

Fun with VSIs and VST Rowids

- A VSI is a Virtual “System-table” Index
  - It uses cursors and brackets for positioning, choosing and retrieving data
  - Provides a rowid for record retrieval (like a real index)
  - Does not have a physically stored b-tree
- All VSTs have exactly one VSI on its one ID field
  - The VST rowids always start at 1
  - The VST ID field almost always start as 1
    - Except for _TableStat, _IndexStat
- VST’s ID vs numerical association
    - ID: _Connect-id = 1
  - Numerical association can start at 1 or 0
    - _Connect-Usr = 0

Quickest Access to VST Data

Indexes retrieve rowid's for a record that satisfies your request

- *FAST*: A "find" using a non-index field is slower than a find by an indexed field

```
find _connect where _connect-Usr = 10  // Find user id 10
```

  - 11 index requests return 11 rowids to the AVM, one at a time
  - The AVM requests 11 records one at a time and filters out 10 of them returning 1 record to the user.
- *FASTER*: Lookup using VSI

```
find _connect where _connect-id = 11  // Find user id 10
```

  - VSI simulates index "search" returning rowid 11, the 11th entry in the user control structure
  - Index returns 1 rowid and 1 record is requested by the AVM
- *FASTEST:* For VSIs, the indexed field value IS the rowid

```
find _connect where recid(_connect) = 11  // Find user id 10
```

  - AVM requests 1 record directly – no VSI involvement

Special Indexing

- _TableStat-id, _IndexStat-id
  - Id is object number starting at –base value
  - Object number is therefore indexed
  - Rowid != Stat-Id
  - Rowid = (Table# - base) + 1

```
Find _DbParams where _DbParams-Name = "-basetable".
    theEntry = (myTable-Num - _DbParams-Value) + 1.
Find _TableStat where _TableStat-Id = theEntry.
//  OR, using Recid rather than VSI
Find _TableStat where recid(_TableStat) = theEntry.
```

-basetable 1

Table ID	Rowid
1	1
2	2
3	3
4	4

-basetable -2

Table ID	Rowid
-2	1
-1	2
0	3
1	4

- OE 11.7.0 fixed VSI for base != 1

Special Indexing

- Find MY stats: _UserTableStat-Id, _UserIndexStat-Id
  - Object stat data by user by table
  - Stat-Id = Rowid
  - Rowid = (User# * range) + (table# - base) + 1

```
Find _MyConnect.
Find _DbParams where _DbParams-Name = "-basetable".
    theBase = _DbParams-Value .
Find _DbParams where _DbParams-Name= "-tablerangesize". *
    theRange = _DbParams-Value .
theEntry = (_MyConn-Userid * theRange) + (myTable-Num – theBase) + 1.
Find _UserTableStat where recid(_UserTableStat) = theEntry.
```

- OE 11.7.0 fixed VSI for base != 1

*NOTE: -baseusertable in OE 11.7.1

base 1, range 3

User	Table	Row
0	1	1
0	2	2
0	3	3
1	1	4
1	2	5
1	3	6

base 2, range 3

User	Table	Row
0	2	1
0	3	2
0	4	3
1	2	4
1	3	5
1	4	6

VSTs: Getting updates / reverting updates

- Applying new vst updates **proutil <*db*> -C updatevst**
- OE 11.4 backward compatibility - after OE 11.5 updatevst
  - New VST tables must be removed for backward compatibility

```
Database Features
  ID   Feature                  Active  Details
 ----  -----------------------  ------  -------
   9   64 Bit DBKEYS              Yes
  23   New VST Tables             Yes
```

An invalid feature 23 has been encountered in the database's Enabled feature list. (11727)

The list of enabled features in database x contains features that are not recognized by this codebase (11810)

proutil <*db*> -C disablenewvsttables

So What's New?

_Lock Improvements

- _Lock table (OE 11.4) and _UserLock table (OE 11.6) query performance
  - Unreliable sequential hash table scan -> now reliable sequential scan
  - Only returns active lock entries (OE 11.5)
- -L 100 000: *For each _Lock:*
  - **179 (sec) now 1 (sec)** improvement ??? %
  - Eliminated latch contention **7.4 mill/sec ~1 billion latches now ~190 latches**
- –L 250 000: *For each _Lock:*
  - Gave up on the query after 58 minutes!
- Bottom line
  - Vast query performance improvement for _Lock and _UserLock
  - Can now reliably be used in production w/no impact

_Lock Improvements (OE 11.4)

- User Identification
  - _Lock-DomainId
  - _Lock-Tty /dev/pts/6 myHostName 6

- Transaction information
  - _Lock-TransId – Join to _Trans table for deep txn information
  - _Lock-Trans-State
    - NONE (dead!)*
    - BEGIN (allocated)*
    - ACTIVE (doing stuff)
    - PREPARING (Prep)[2PC]
    - PREPARED (Phase 1) [2DB]
    - COMMITTING (Phase 2) [2DB]
  - _Lock-Trans-Flags
    - FWD
    - UNDO

_DbParams (OE 11.5 – new table)

- Consistent reporting of database startup parameters
  - _DbParams – New VST table with startup parameters
    - _Startup VST is obsolete
  - _DbParams, prolog and broker to .lg file, promon with translated messages
  - Contains {Name, value, msg #, description, default info, modifiable flag}
- Update _DbParams-value to change startup parameter online

```
Find _DbParams where _DbParams-Name = "-spin".
Assign _DbParams-Value = "10000".
```

- Values are all character datatype
- Failure to assign is silent so double check!

_Dbparams Example

Name	Value	Description	Msg-Num	Default	Modifiable
-aiarcdir	0	After-image Management Archival Directory List (-aiarcdir): Not Enabled	13873	yes	no
-aiarcdircreate	0	Create After-image Management Archival Directory(s) (-aiarcdircr)	13874	yes	no
-aiarcinterval	-1	After-image Management Archival Interval (-aiarcinterval): -1	13872	yes	no
-aibufs	20	Number of After-Image Buffers (-aibufs): 250	4256	no	no
-aistall	0	After-Image Stall (-aistall): Not Enabled	4254	yes	no
-baseindex	1	Starting index number for statistics range (-baseindex): 1	17555	yes	yes
-basetable	1	Starting table number for statistics range (-basetable): 1	17554	yes	yes
-baseuserindex	1	Starting index number per user for statistics range (-baseuserindex): 1	18406	yes	yes
-baseusertable	1	Starting table number per user for statistics range (-baseusertable):1	18404	yes	yes
-bibufs	20	Number of Before-Image Buffers (-bibufs): 250	4252	no	no
-bistall	0	BI File Threshold Stall (-bistall): Disabled.	6551	yes	no
-bithold	0	BI File Threshold size (-bithold): 0.0 Bytes	9238	yes	no
-blocksize	8192	Database Blocksize (-blocksize): 8192	6573	yes	no
-bwdelay	0	BIW writer delay (-bwdelay): 0	12812	yes	yes

_Servers - Server parameters (OE 11.5)

- Primary broker, manual broker / server, auto DB and SQL Server information
- _Server-Broker-Pid
- _SrvParam array fields {Name, Value, Type, Desc, Msg-Num, Default, Modifiable}
  - One array entry per parameter

- *-maxport*
- *-minport*
- *-Ma*
- *-Mi*
- *-Mm*
- *-Mp*
- *-Mpb*
- *-N*
- *-PendConnTimeout*
- *-S*
- *-ServerType*
- *-SQLCursors*
- *-SQLStack*
- *-SQLStmtCache*
- *-SQLTempStoreBuff*
- *-SQLTempStoreDisk*
- *-SQLTempStorePageSize*
- *-SQLTruncateTooLarge*

_UserLock

- Contains multiple field arrays containing info of first 512 current locks
- Added **current total locks** and **current totals by lock type** (OE 11.6)
  - _UserLock-Total-
    - *Record*
    - *RecGet*
    - *EXCL*
    - *IX*
    - *Table*
    - *Partition*
    - *IS*
    - *Purge*
    - *SHR*
    - *SIX*

Lock table overflow, increase -L on server (915)

- Who's the guilty party?
  - _UserLock-HWM (OE 11.7.0)

For each _UserLock by _UserLock-HWM descending:

User Activity Statistics (OE 11.7.0)

- Separated database and user level parameters for table and index activity statistics
- Example:
  - 2000 tables 8000 indexes & 1000 users
    ((2000 * tstatsize(80)) + (8000 * istatsize(48))) * 1000 = **~380Mb**
  - ~Equivalent to –B 50 000
- New Parameters
  - *-baseuserindex*
  - *-baseusertable*
  - *-userindexrangesize*
  - *-usertablerangesize*
  - Defaults to database –basetable/index params
- _DbParams and _StatBase can update base stats value on the fly.

```
Find first _Statbase.
Assign _BaseUserTable = 100 _BaseUserIndex = 10.
```

User Activity Statistics (OE 11.7.0)

- Separated database and user level parameters for table and index activity statistics
- Example:
  - 2000 tables 8000 indexes & 1000 users
    ((2000 * tstatsize(80)) + (8000 * istatsize(48))) * 1000 = **~380Mb**
  - ~Equvilent to –B 50 000
- New Parameters
  - *-baseuserindex*
  - *-baseusertable*
  - *-userindexrangesize*
  - *-usertablerangesize*
  - Defaults to database –basetable/index params
- _DbParams and _StatBase can update base stats value on the fly.

```
Find first _DbParams where _DbParams-Value = "-baseusertable".
Assign _DbParams-Value = 100.
```

Database Buffer Pool activity (OE 11.7.0)

- _ActBuffer: {-B}, {-B2} and {-B, -B2}
  - Provides read/write, and checkpoint activity in the buffer pools
  - Stats on # of each block type (RM, index, master, …) in the cache
  - # active buffers – tells you if your buffer pool if approaching full or not
    - *_Buffer-Active*
    - *_Buffer-Master*
    - *_Buffer-Index*
    - *_Buffer-RM*
    - *_Buffer-Free*
    - *_Buffer-Seq*
    - *_Buffer-Area*
    - *_Buffer-Object*
    - *_Buffer-ObjList*
    - *_Buffer-Control*
    - *_Buffer-ClusterMap*
    - *_Buffer-ObjClist*
    - *_Buffer-BlockMap*

_Checkpoint - Synchronizing database changes from memory to disk

1. All database changes are halted
2. BI buffer pool is flushed (-bibufs)
3. AI buffer pool is flushed (-aibufs)
4. Db buffer pool is scanned (-B)
    1. Buffers previously marked for checkpoint are written out (Buffer Flushes / DB Writes)
    2. “Dirty” buffers are marked for the next checkpoint & added to checkpoint queue
5. File system cache is synchronized to disk
6. Db changes are allowed once again

- It is important to know the activities impacting the checkpoint
    - Avoid “Buffers flushed at checkpoint”
    - Increase “Time between checkpoints”

More Checkpoints

- History of 8 checkpoints is not a history
- Additional checkpoints (OE 11.7.0)
  - -numCheckpointStats
    - New startup parameter
    - Default 32
    - Max 1024
- Also included
  - _Checkpoint-Buffers: written so far (OE 11.6.0)
  - _Checkpoint-Cluster: bi cluster ring information (OE 11.7.0)
  - _Checkpoint-Number (OE 11.7.0)

_Checkpoint – Why is my checkpoint taking so long?

```
-------------  Database Writes  -------------
 CPT Q      Scan     APW Q    DB Writes   BI Writes
   0          0        0         99           9
   0          0        0        188          19
```

- “Flushes” changed to “Writes”
- Added BI writes (OE 11.6)
- Still not enough information
  - What else happens in a checkpoint?
  - How long does each activity take?

_Checkpoint – Why is my checkpoint taking so long?

-- Database Writes --		---------- Performance Timings --------			
DB Writes	BI Writes	Duration	Sync Time	DB Write	BI Write
99	9	0.06	0.01	0.01	0.05
188	19	0.09	0.01	0.01	0.07

- Total Duration & Sync Time (OE 10 something)
- DB Write Time, BI Write Time (OE 11.6)
  - Helps identify if APW is keeping up and auto tuning properly
    - Auto tuning enhanced to consider previous checkpoint statistics

- Increase –bibufs to avoid “empty bi buffer waits” can have side effects
  - Monitor APW / BIW activity
  - Monitor Checkpoint cost and find a balance
  - Validate Bi file’s disk speed

_AreaStatus (OE 11.7.0)

- _AreaStatus-Type
  - Control, Schema, Data, BI, AI, TL
- AI improvements including
  - _AreaStatus-State
    - *Full*
    - *Busy*
    - *Archived*
    - *Empty*
    - *Locked*
    - *Arch/Locked*
  - _AreaStatus-LastOp
    - Timestamp of last state change
  - _AreaStatus-AIActivated
    - Timestamp when “New”ly activated
  - _AreaStatus-AI-Seq

_Connect (OE 11.7.0)

- Database wide connection information
- _Connect-NumTrans
  - # transaction begin requests
  - Includes commit, rollback, and txns with no changes
- Private buffers
  - Avoid fouling DB buffer pool by table scan type queries
  - _Connect-NumSeqBuffers, _Connect-UsedSeqBuffers
    - Already in _MyConnection
  - Application can only update _MyConn-NumSeqBuffers at runtime

```
Find _MyConnection.
Assign _MyConn-NumSeqBuffers = 20.
      For each History  NO-LOCK: end.    // Table scan to foul buffer pool replacement
Assign _MyConn-NumSeqBuffers = 0.
```

Db user notify (OE 11.7.0)

- Provides active checking for schema changes – no forced logout/login required
  - *Index activate*
  - *Binary load w/build indexes*
  - *MT idxbuild*
  - *TP idxbuild*
- Separate client thread “polls” for schema changes

  proserve <*db*> -usernotifytime <*poll-freq*>

  - Polling is off by default but can be changed at runtime*
    - promon or _DbParams VST

- Suggestion: Set to balance performance vs maintenance requests
  - Set relatively high at db startup: 600 (10 minutes)
  - Anticipate a change? Set it lower prior to maintenance operation

_Connect: DbUserNotify Support (OE 11.7.0)

- _Connect-TimeStamp
  - Schema cache timestamp (OE 11.6)
- _Connect-NotifyTime
  - Currently set poll time
- _Connect-LastNotifyCheck
  - Time of last poll for notifications
- _Connect-Notifications
  - Notifications outstanding or in process
  - Currently only supports “R” for schema re-cache

User Misc Field (OE 11.7.0)

- What is it?
  - 16 bytes of “user” defined data
  - _Connect and _MyConnection share currently set value
  - Could set as part of login process
  - Could set for application monitoring
    - Does not provide history like Application Auditing
- Security concerns…
  - Restrict update for non-DBA’s to _MyConnection only

Find _MyConnection.

Assign _MyConn-UserMisc = “Now I’m here.”.

For each _Connect:

Assign _Conn-UserMisc = “I’m everywhere!”.

Transaction Activity (OE 11.7.0)

- _UserIO and _Trans – per user/trans BI note read/write activity
  - *_UserIO-BIRecReads*
  - *_UserIO-BIRecWrite*
  - *_ Trans-BIRecReads*
  - *_ Trans-BIRecWrites*

- User I/O statistics
  - Includes forward and undo processing activity
  - “Is that user doing anything?”

- _Trans statistics
  - Is a transaction doing anything or just sitting there
  - Is the Bi growth really needed or is it from incorrect transaction scoping?

Resetting Database Statistics

proutil <*db*> -C zerostats (Use cautiously – all database statistics are zeroed)

Find _DbStatus. Display _ DbStatus-ZeroStatsDate.

- Remember: VST dates are character datatype “Thu Apr 27 16:18:21 2017”

- Affect on promon – actions per second vs database up time
  - Value of per second activity degrades over time

R&D ⟹ 1. Status Displays … ⟹ 1. Database

Database was started at: 04/24/17 14:15
It has been up for: 24:09:48
Time of last zerostats operation: 04/25/17 10:15

PUGCHALLENGE EXCHANGE
AMERICAS