SAP Sybase ASE Backup & Restore Best Practices

or

Why we don't care about backups

UKSUG

SAP DATABASE & TECHNOLOGY USER GROUP

Joe Woodhouse

23 March 2022

PRIMA DONNA CONSULTING

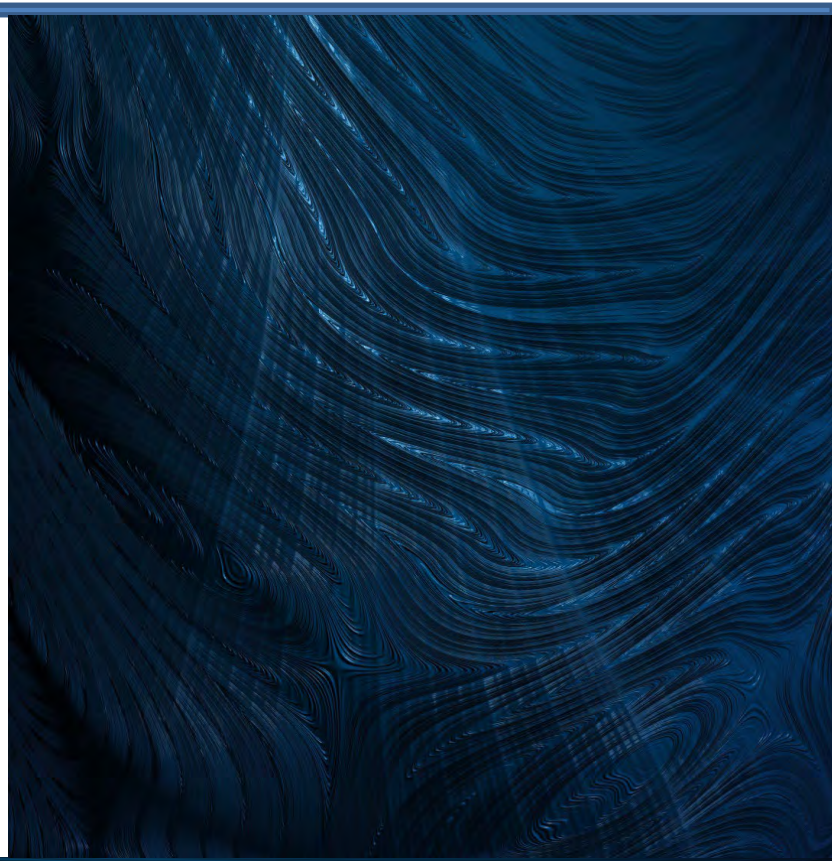
joe.woodhouse@primadonnaconsulting.com

How we'll do this

- Session will be recorded and recording shared with you later
- Slides will be available after (you don't need to take extensive notes)
- Audio off please, ask questions in chat (I can see them)
  - … but maybe hold until the end, we have a *lot* to cover
- Plenty of time for Q&A after (I'll stay late if that's helpful)
- Happy to take questions in email (spirit of the staircase)
- QR link to (brief! anonymous!) survey on the last page (thank you)

Contents

- About the author & this presentation
- First, some IT philosophy
- Some ASE internals
- The jobs to be done
- Best practices
- Some actual science



About the author

- Work exclusively with ASE, IQ, and Replication Server for 26+ years
- Sybase Australia 1996 – 2003
- Database engineer @ Prima Donna Consulting for 19+ years
- Based in London, UK and Melbourne, Australia
- International Sybase User Group Board of Directors since 2010
- UK Sybase User Group Board of Directors since 2019
- Not a lawyer – no charge for emails!

Acknowledgement

- Many thanks to Kevin Sherlock
  - Got me to revisit and rethink the fundamentals
  - Made me aware of the qatraceon trace flags

Why this presentation?

- Like sui mai dumplings and crème brûlée…
- … backups are easy to do OK, but hard to do excellently

What we won't do here

- This isn't Sybase DBA 101
- We won't cover the basic architecture, process, or syntax for:
  - Backup Server
  - dump database
  - dump tran
  - load database
  - load tran
- q.v. *System Admin Guide v2* and *Reference Guide - Commands*

First, some IT philosophy (1 of 4)

- Almost everything we think (and do) about backups is backwards
- What is the job to be done?
  - People who buy a ¼" drill bit don't want a drill bit
    - » They want a ¼" hole
- We don't actually care about backups!
  - We care about damage control when something breaks
  - We care about a time machine for our database

First, some IT philosophy (2 of 4)

- Often we do what is easy rather than what is helpful…
- … like looking for our keys under the light rather than near our car

First, some IT philosophy (3 of 4)

- We don't need to optimise for backups (mostly)
  - ASE backups: online, planned, controlled, automated, and calm
    - » Time and complexity really don't matter
- We should – but don't! – optimise for restores
  - Restores: outages, unplanned, urgent, often manual, and not calm
    - » Time and complexity both matter a *lot*
  - Any problem often means starting again from the beginning

First, some IT philosophy (4 of 4)

- And yet, most things we do are things to make backups easier
  - Or to tick boxes according to site policies
- Many of those choices do not help restores
- Many of those choices make restores worse!
  - Slower, more complex, adding to existing risks, creating new risks
- We also don't do many things that would make our restores better

Internals

Internals (1 of 2)

- Common myth #1:
  - Database dump is a one to one sequential copy of database pages
- Wrong, because:
  - Pages aren't necessarily written in order (striping)
  - The same page can appear more than once (online activity)
  - Full database dump also always contains a transaction dump
  - Has to deal with minimally logged transactions

Internals (2 of 2)

- Common myth #2:
  - Transaction dump is a one to one sequential copy of log pages
- Wrong, because:
  - Pages aren't necessarily written in order (striping)
  - The same page can appear more than once (online activity)
  - Transaction dump contains entire transaction log
    - » Even if those logs have been written in an earlier dump
    - » (Hold that thought)

Size of backup (1 of 5)

- Dump database size depends on pages *used* not *allocated*
- Size also depends on the settings of sp_dumpoptimize (ASE 15.0, 2005)
  - "reserved_threshold" and "allocated_threshold"
  - If a device fragment has > "reserved_threshold" % pages used
    - » then dump every page in that fragment
  - Else if allocation unit has > "allocated_threshold" % pages allocated
    - » then dump every page in that allocation unit
  - Defaults are 85% and 40% respectively!

Size of backup (2 of 5)

- Dump database size also depends on concurrent writes
- All writes change data (including index) pages
  - These must be tracked, and may be rewritten many times
    - » sp_configure "optimize for faster load"
    - » Resends changed pages, dump is bigger, load is faster
    - » May need ASE boot trace flag –T3022 for high activity
- Minimally logged writes must also be separately tracked
- Don't care about concurrent reads, no impact

Size of backup (3 of 5)

- Dump database cumulative size depends on pages *changed*
  - … since last full dump database
- Dump database cumulative size also depends on concurrent writes
  - All writes change data (including index) pages
    - » These must be tracked, and may be rewritten many times
  - Minimally logged writes must also be separately tracked
    - » Affected pages re-dumped just before end of dump
  - Don't care about concurrent reads, no impact

Size of backup (4 of 5)

- Dump transaction size depends on transaction log size
  - Must write entire transaction log
  - Including inactive (committed, rolled back) transactions
- Note: this does not mean number of data pages changed!
  - update t set x = x
    - » No actual data or index page changes, but a huge transaction

Size of backup (5 of 5)

- Size of any dump affected by dump compression
  - dump … to "compress::[…]"
  - dump … with compression = < 1-9 >
  - dump … with compression = { 100 | 101 }

Time of backup (1 of 3)

- Time taken by any backup depends on:
  - How much work it must do
  - How much resources it has to do it with

Time of backup (2 of 3)

- How much work it must do
  - How many pages to write
    - » In use, not allocated, but remember sp_dumpoptimize
  - How much concurrent writing
    - » But remember sp_configure "optimize for faster load"
  - If compressed dump
    - » Type of compression and compression level

Time of backup (3 of 3)

- How much resources it has to do it with
  - Number of stripes
  - # of CPU cores and their speed (especially for compressed dump)
  - Backup Server memory (-m*NNN*)
  - Internal parallelism = number of devices for this database
    - » dump database / dump database cumulative only
  - I/O systems and settings
    - » Hardware, # of LUNs, mount point tuning

Time of load (1 of 2)

- Like dumps, load timing depends on work to be done, and on resources
- Work to be done is based firstly on the dump size
- Also based on the target database pages *allocated* not *used*
  - Large empty database = fast dump, slow load
- Also based on concurrent write activity during the original dump
  - Those writes have to be dealt with
  - Transactions have to be rolled forward or rolled back

Time of load (2 of 2)

- Resources all as for dump timings
  - Stripes; hardware; BS memory; # of ASE devices; OS tuning
- New factor (only for loads): remember dsync and directio?
  - Loads use the type of I/O used by the underlying ASE devices
  - This can be overridden
    - … by ASE settings
    - … by Backup Server settings

Digression: ASE trace flags (1 of 4)

- Common myth about scope of trace flags:
  - Boot-time trace flags are server-wide…
  - … but run-time are for just me, right?

Digression: ASE trace flags (2 of 4)

- WRONG, there is actually only one that is for the current session:
  - dbcc traceon(3604), send output of other dbcc's to this session
- All other trace flags are always set globally

"Traceflags sometimes behave as if they were session specific, but this is not actually the case. ASE maintains one global array to track traceflags. Some traceflags affect all spids, some affect only spids that have used dbcc traceon to turn on at least one traceflag - that session is then affected by all traceflags that are on. When a session turns off a traceflag, the flag is turned off for all spids as well."

Digression: ASE trace flags (3 of 4)

- If you want a flag set just for your session, use

```
set switch [serverwide] { on | off }
   < traceflag [, traceflag] >
   [ , with option [, option] ]

option: { override, no_info }
```

- Granular permissions: permission to run can be separately granted

Digression: ASE trace flags (4 of 4)

- dbcc traceflags -- list all currently active
- dbcc istraceon(nnn) -- @@error = 0 if flag is set, -1 if not set

I read so you don't have to (1 of 7)

Setting (ASE)	Scope	Effect
sp_configure "enable buffered io for load"	load	Ignore ASE device settings for dsync/directio and perform ordinary UNIX buffered write on devices during the load.
sp_configure "enable concurrent dump"	dump	Stops dump database from blocking (one) dump tran on that database.
sp_configure "enable delta dump tran"	dump, load	If a log page has been dumped already, do not include it in future dump trans. Smaller & faster tran dump = faster load.
sp_configure "enable dump history"	dump, load	ASE will track dumps and loads with their parameters. Prerequisite for some other features such as listonly=load_sql and sybrestore.
sp_configure "enforce dump configuration"	dump	If set, all dumps must use a predefined dump configuration. **Not recommended.**
sp_configure "optimize for faster load"	dump, load	Resend full pages rather than just transactions for pages changed during the dump. Makes dump file larger and slower; makes load faster. Only relevant to concurrent writes.
sp_configure "restore database options"	load	Restore (almost all) database options as part of load database.

Italics indicates not documented in the manuals, but documented in CRs and bugfix lists.

I read so you don't have to (2 of 7)

Setting (ASE)	Scope	Effect
sp_dboption "allow db suspect on rollback error"	*other*	*Rare condition fills logs, prevents recovery, database unusable. If set, marks db suspect without requiring ASE restart, and allows special dump tran no_truncate.*
sp_dboption "allow incremental dumps"	dump	Allows cumulative (incremental-since-full) dumps on this database.
sp_dboption "enforce dump tran sequence"	other	Prevents any minimally logged behaviours that would prevent the next dump tran.
sp_dboption "full logging for [...]"	other	Uses full logging instead of minimal logging; **many** consequences, be careful.
sp_dump_info	dump	Estimates size of next cumulative dump. Requires "allow incremental dumps".
sp_dumpoptimize	dump, load	Increases dump time but *might* decrease dump size. Does not persist past Backup Server restart!
– T3000 (boot, dynamic)	*dump*	*Print dump start and end times; needs 3604 or 3605 to display. Dumps only.*
– T3022 (boot, dynamic)	*dump, load*	*Occasionally needed to avoid issues with high concurrent writes when "optimize for faster load" is enabled.*
– T3170 (boot, dynamic)	*load*	*Occasionally needed to avoid issues during load tran after online database for standby access.*

Italics indicates not documented in the manuals, but documented in CRs and bugfix lists.

I read so you don't have to (4 of 7)

Setting (Backup Server)	Scope	Effect
–C	other	Number of concurrent Backup Server connections, might need 3 per dump or load.
–m	dump, load	Memory in Mb for Backup Server to use. Is divided between all stripes of all concurrent sessions.
–N	other	Number of network connections to Backup Server. Used by local and remote dump/loads.
–p	other	Packet size, used only for remote Backup Servers. Local sessions always use 2048 (ASE 15.7+) or 512 bytes (earlier).
–P	other	Number of threads. One thread needed per stripe per concurrent dump/load.
–T	other	Open Server trace flags. Not usually useful. Only the last one listed has any effect, must logically OR multiple flags together in a single parameter. https://wiki.scn.sap.com/wiki/display/SYBASE/Open+Server+Traceflags

Italics indicates not documented in the manuals, but documented in CRs and bugfix lists.

I read so you don't have to (5 of 7)

Setting (Backup Server)	Scope	Effect
–D	dump, load	Backup Server trace flags. Only the last one listed has any effect, must logically OR multiple flags together in a single parameter, i.e. to specify both –D64 and –D2048 use single –D2112 instead. https://wiki.scn.sap.com/wiki/display/SYBASE/Backupserver+Traceflags
–D1	other	Traces Backup Server API calls
–D2	other	Forces tape autoconfig
–D4	other	Traces tape autoconfig
–D8	other	Traces async I/O
–D16	other	Traces RPC calls and events
–D32	other	Forces sync I/O

Italics indicates not documented in the manuals, but documented in CRs and bugfix lists.

I read so you don't have to (6 of 7)

Setting (Backup Server)	Scope	Effect
–D64	*other*	*Forces "with verify" for all dump database (only)*
–D128	*other*	*Required for versions 15.0.3 ESD#3 and below for XPDL to work from HP Tru64 to later versions*
–D256	*other*	*Enables directio on the **file(s)**, issues with compressed dumps until 15.7 ESD#2*
–D512	*other*	*Forces directio on the **devices**, only for uncompressed dumps, problems with ZFS until very recent versions*
–D1024	*other*	*Enables posix async I/O (instead of KAIO)*
–D2048	*other*	*Resend the full changed page list at end of dump, similar to "optimize for faster load" but 100%*
–D4096	*other*	*Suppress messages "Attempting dsync/directio I/O for device [...]" from error log*
–D8192	*other*	*Support for multithreaded TSM API*

Italics indicates not documented in the manuals, but documented in CRs and bugfix lists.

I read so you don't have to (7 of 7)

- exec SYB_BACKUP...qatraceon | qatraceoff *nnn* -- three dots

qatraceon	Scope	Effect
0	*diagnostic*	*Print currently active qatrace flags*
1	*diagnostic*	*Print blocksize*
2	*diagnostic*	*Print I/O optimization parameters*
3	*testing*	*Allows dumping to /dev/null for testing (not loading!)*
4	*diagnostic*	*Print allocation percentage of each allocation unit; relevant to sp_dumpoptimize*
5	*diagnostic*	*Print database page size*
8	**verify**	**Forces all dump database (not cumulative or tran) to use WITH VERIFY behaviour**
105	**diagnostic**	**Adds messages to BS error log re. resends of page lists; relevant to "optimize for faster load" and –D2048**

Italics = Not documented in the manuals, or supported

Bold = Not in the manuals, but are in CRs and buglists (?? supported ??)

Jobs to be done:
Start as you mean to finish
Tune for what matters

What is the job to be done? (1 of 2)

- We need damage control for when things break
  - Our hardware and infrastructure
  - Our processes
  - Our scripts
  - Our application code
  - Our people
  - External threats
  - The ASE and Backup Server software themselves

What is the job to be done? (2 of 2)

- We need a database time machine
  - System is alive but someone did something wrong
    - » Roll back to before an otherwise unrecoverable error
  - Answer questions about the past (auditors, regulators)

Common practices for backups

- Dump database nightly, to disk
- Dump tran hourly (or every 15 mins… 10 mins… 5 mins…), to disk
- bcp out some system tables, to disk
- All files on disk included in file system backups to tape
- Keep two days' files locally on disk
- DR is through disk replication (e.g. SRDF)
- Weekly restore into Prod Support / UAT
- *… what's wrong with this picture???*

Common problems with backups!

- Transaction logs could fill during the dump database
- A restore scenario could involve 25 (or 97… or 145… or 289) loads
- How long will it take to do restore after any disaster?
- If any dumps are broken, we won't know for up to one week
- Minimally logged transactions – do they break the tran dumps?
- We might fill local storage
- Database problems will be copied to DR
- Will "up to the minute" special backup work?

Common practices for restores

- Manually restore loads
  - … from local disk
  - … lots and lots of loads…
- Restore handled in isolation of wider context
- Handed back after the restore
- *… what's wrong with this picture???*

Common problems with restores!

- What is broken, how much have we lost?
- Manual process: wrong sequence, bringing online too soon
- Where are the backup files? The documentation?
- Do we have what we need to reinstall? Rebuild? Load?
- Handled in isolation: database layout, options, ID mismatches
- Forgetting post-load activities
  - Forgetting to bring online
  - Forgetting to disable replication truncation point

Database time machine practices

- "Oops someone accidentally ran a TRUNCATE TABLE"
- "Auditor/regulator needs to see the database as of a certain date"
- Usual answer:
  - "I'll run a load sequence including WITH UNTIL_TIME"
- *What's wrong with this picture???*

Database time machine problems!

- If restoring PROD
  - Outage! Hurry up
  - Do we have everything we need?
- If restoring to a copy
  - Do I have an ASE with enough free space for this copy?
  - Displace an existing copy?
  - Do we have everything we need?

Setting up for a win: Best practices

Best practices (0 of 7)

1. Optimise for what's difficult
2. Start with a well-tuned ASE and Backup Server
3. The right backups: dump database + cumulative + dump tran
4. Taken in the right way: dump & load options + dump history
5. Defence in detail: ensure we have everything we need
6. Defence in depth: more than one type of backup/DR
7. Zero trust: assume everything is broken until proven otherwise

Best practices (1 of 7)

1. Optimise for what's difficult
   - Backups are easy and the stakes are low
   - Restores are hard and the stakes are high!
   - We mostly don't care about how long the backup takes
     - » We maybe care about minimising concurrent writes
   - We mostly don't care about how complex the backup process is
     - » We definitely care about reducing load complexity and risk

Best practices (2a of 7)

2. Start with a well-tuned ASE and Backup Server

Category	Type	Setting	Default	Value	Comment
Better dumps	sp_configure	enable concurrent dump	0	1	Allow concurrent DD and DT
Better dumps	sp_configure	enforce dump configuration	0	0	Don't recommend; forces same number of stripes for all databases
Better dumps	sp_dboption	allow incremental dumps	false	true	Needed for cumulative backups
Faster dumps	sp_configure	enable delta dump tran	0	1	Subsequent DT will be smaller and faster
Faster dumps	Backup Server	-m	--	*	Start with 512, test up to 2048
Faster dumps	ASE devices				More ASE devices = internal dump parallelism
Faster dumps	OS tuning	more LUNs			Spread dump over multiple filesystems
Faster dumps	OS tuning	filesystem tuning			ext4 disable journaling xfs tune barriers "disk" tuning, nr_requests, elevators
Smaller dumps	sp_dumpoptimize	"archive_space=[…]"	--	"minimum"	Small increase to dump time Possible reduction in dump size Smaller file = faster file transfers, faster load

Best practices (2b of 7)

2. Start with a well-tuned ASE and Backup Server

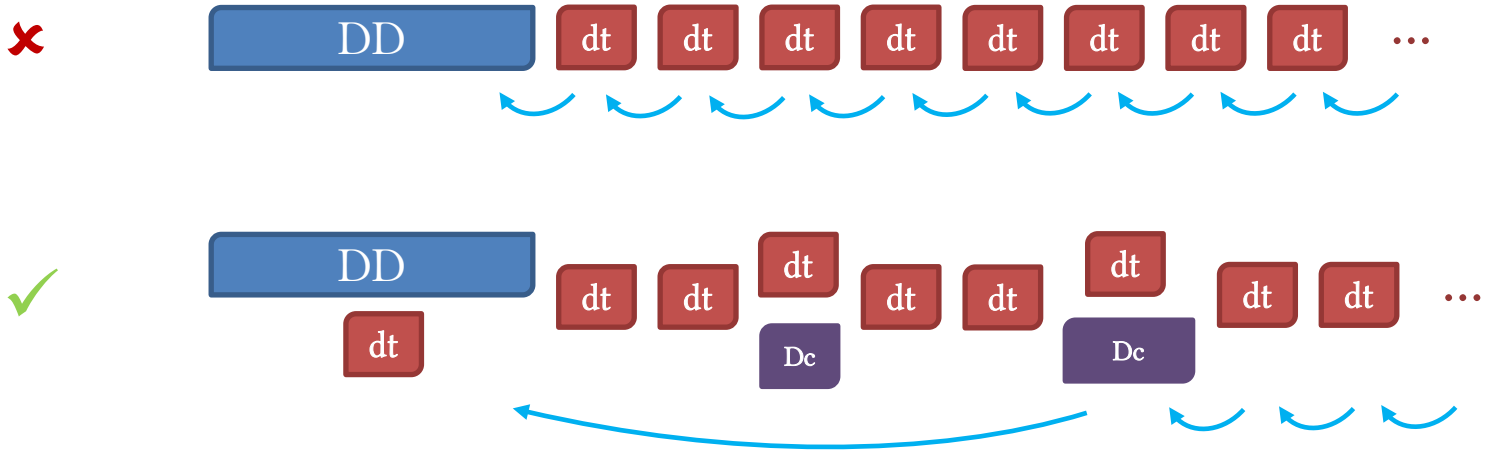
Category	Type	Setting	Default	Value	Comment
Better loads	sp_configure	enable dump history	0	1	Helps automate loads
Better loads	sp_configure	restore database options	0	1	
Faster loads	sp_configure	enable buffered io for load	0	1	Sometimes much faster load; testing required
Faster loads	sp_configure	optimize for faster load	0	1	Dumps will be larger and slower May require ASE –T3022 if many concurrent writes
Faster loads	Backup Server	-m			Start with 512, test up to 2048
Faster loads	Backup Server	-D256			ASE 15.7 ESD#2+ Directio for the dump *files* ; test carefully
Faster loads	Backup Server	-D2048			Better version of "optimize for faster load"
Faster loads	ASE devices				As per faster dumps
Faster loads	OS tuning				As per faster dumps

Best practices (3a of 7)

3. The right backups: dump database + cumulative + dump tran
   - sp_dump_info
     - » if size of cumulative dump is < 40% of full, it will be faster
   - Belt and braces
     - » DDC can re-enable DT after minimally logged
     - » DDC can roll through a missing/bad DT
     - » DT can roll through a missing/bad DDC (and DD!)

Best practices (3b of 7)

3. The right backups: dump database + cumulative + dump tran



Best practices (4a of 7)

4. Taken in the right way: dump & load options + dump history
   - Stripes
     - » Start with cores / 2, test & time up to cores x 2
   - Compression
     - » "with compression" almost always faster than "compress::"
     - » Old advice: use lowest compression, but see end of this talk
     - » Compression 100 | 101 faster than 1..9, but less effective
     - » Joe's advice: who cares how long the backup takes??

Best practices (4b of 7)

4. Taken in the right way: dump & load options + dump history
   - Dump history is essential!!!
   - sybrestore
     - » Generates full sequence of load statements for you
   - load database … with listonly=load_sql
   - load database … with listonly=create_sql
     - » This one doesn't require dump history

Best practices (4c of 7)

4. Taken in the right way: dump & load options + dump history
   - Before every dump…
     - » sp_flushstats
     - » checkpoint
     - » dump tran
       - smaller log = smaller dump database file
       - smaller log = faster load

Best practices (5 of 7)

5. Defence in detail: ensure we have everything we need
   - Where is the documentation? (On the server that just died???)
   - System tables, scripts to recreate *everything*
     - » Treat these like source code; use version control; maintain!!!
   - Dump history will generate your load scripts for you
   - Can you reinstall ASE? .CFG file? RUN_SERVER? etc.
   - Where are the dump files? Tape? Another host?
     - » These all need to be tuned too… and tune for *restore*

Best practices (6a of 7)

6. Defence in depth: more than one type of backup/DR
   - Backups (& mirroring & disk replication) are *physical* copies
     - » Any corruptions or mistakes will also be copied
   - If this data is important, also need Replication Server
     - » Copies & reapplies transactions; not data movement
   - Cumulative backups aren't just faster than constant full DD
     - » They also remove dependencies on transaction dumps

Best practices (6b of 7)

6. Defence in depth: more than one type of backup/DR
   - Special "after a crash" dump for up to the minute recovery
     - » dump tran … with no_truncate
     - » Even if one or more data devices are dead, get a dump tran
   - … no use to us if ASE itself is dead
   - New sybdumptran utility, point it at the dead ASE
     - » -h option works with dump history
     - » Dump history will include this dump in the load SQL

Best practices (7 of 7)

7. Zero trust: assume everything is broken until proven otherwise
   - Dump … with verify
   - Load … with verify (or just load it at all, somewhere, anywhere)
   - Assume you won't have access to scripts & documentation
     - … how do you know that what you have is up to date?
   - Archive database
     - Mount the dump file(s) as a special read-only database
     - Does not need full size, and is a good test

Time for some science (1 of 2)

- DEV host #1: 8 cores, 94Gb RAM, RHEL 7.9
- Database = 818,510Mb, sp_spaceused says 482,759,632Kb reserved

#	Test	Dump time	Dump size
0	No tuning, 1 stripe	30m51s	573Gb
1	compress::1::	2h31m20s	104Gb
2	with compression = 1	2h32m58s	112Gb
3	with compression = 1, 8 stripes	20m52s	112Gb
4	as above + sp_dumpoptimize "archive=minimum"	22m16s	100Gb
5	as above but 10 stripes	20m51s	100Gb
6	as above + Backup Server –m2048	19m39s	100Gb
7	as above but compression = 101	14m52s	136Gb

Time for some science (2 of 2)

- DEV host #2: 8 cores, 32Gb RAM, RHEL 7.9
- Database = 5,500Mb, sp_spaceused says 3,974,878Kb reserved

#	Test	Dump time	Dump size	Load time	EBIFL load time
0	No tuning, 1 stripe	34s	3.9Gb	49s	21s
1	sp_dumpoptimize "archive_space=minimum"	35s	3.9Gb	48s	25s
2	As above + compression = 1	1m24s	907Mb	51s	36s
3	As above + compress::1::	1m15s	782Mb	45s	38s
4	As above + compression = 9	4m22s	771Mb	52s	34s
5	As above + compress::9::	4m33s	614Mb	46s	31s
6	As above + compression = 100	1m08s	1.4Gb	52s	44s
7	As above + compression = 101	58s	1.1Gb	47s	31s

Other jobs to be done

- Upgrade
- Replication rebuild
- Cross-platform dump and load ("XPDL")
  - So many things can go wrong, keep it as simple as possible
  - Don't upgrade
  - Don't change database geometry
  - Dump best taken with zero active users

Q&A, and thank you

Joe Woodhouse
joe.woodhouse@primadonnaconsulting.com

Too busy putting out fires to reduce your toil?
Answering the on-call phone too often?
Drowning in technical debt?

Joe is a freelance consultant available through Prima Donna Consulting and can be engaged nimbly and flexibly.

Would you like this or another of Joe's papers presented to your workplace? Recent topics:

- ASE Tips & Tricks
- ASE Tempdb performance & tuning
- ASE Like It's (Still) 2012
- ASE Anti-Patterns

PRIMA DONNA CONSULTING