Database Repair & Recovery
Dan Foreman, White Star Software

Abstract: Despite your best efforts to protect a database from harm, sometimes bad things happen to good databases. So what can you do if your database is damaged or corrupted? The “standard” recovery methods usually require a dump/reload or restoring the database from a backup. However, sometimes there are better alternatives. Learn about the different kinds of corruption and how they can occur. Learn how to verify the extent of the damage. And how, if possible, to repair the database or recover data from a damaged database.
Database Repair & Recovery

Dan Foreman, White Star Software
danf@prodb.com

PUG Challenge EMEA 2017
Introduction: Dan Foreman

- Progress guy since 1984
Introduction: Dan Foreman

• I used to be a Crazy Frequent Flier
  – 3.4 million miles on United Airlines
  – 3.5 million miles on American Airlines
Introduction: Dan Foreman

• I enjoy cycling & basketball (not @ the same time)
Introduction: Dan Foreman

• Author of several publications
  – *Progress Performance Tuning Guide*
  – *Progress Database Administration Guide*
  – *Progress System Tables*
  – *Progress Database Best Practices*
  – Available in traditional book or electronic versions
Someone found an alternative use for my books
Introduction - Disclaimer

- I gave a similar presentation in 2014 and it has some content from that session but it’s also much different
- This is the longest presentation I have ever done (over 70 slides) so I need to move rapidly... you might want to record the session
- Mobile phones on mute/vibrate please
• Is there is documentation for fixing or recovering a broken DB?
• I’m not confident it’s possible to produce a generic how-to-fix-it document or YouTube video that could safely be used in an emergency (sort of like trying to become a doctor when you’re sick by looking at WebMD)
• Is it possible to teach database brain surgery in a one hour presentation or a YouTube video?
• Even Progress wants you to call PTS when using *proutil dbpr* to repair a DB
• Maybe, but I can probably do a better job of teaching CAS (Collision Avoidance System)

• Wikipedia definition: A Collision Avoidance System is an automobile safety system designed to reduce the severity of a collision. It is also known as a **precrash system**, forward collision warning system, or collision mitigating system.

• Note that we have one big advantage that human brain surgeons don’t have…. Answer coming soon
True Crime #1

- Fortune 500 Company (sorry but they would not appreciate us sharing their name)
- Severe DB Corruption: February 23
- Last Good Backup: January 11
- Last Good AI Files: January 17
- We worked with Progress to facilitate a special version of `rfutil` that would ignore inconsistencies during the Roll Forward process
True Crime #2

- Fortune 1000 Company
- HP/UX Server
- System Admin for the server had been outsourced to IBM
- Backups had been outsourced by IBM to a 3rd party
- 3rd party stopped doing backups, unannounced, due to payment dispute
- DB Corrupted
- Restoration impossible
True Crime #3

- Database corrupted with multiple #1124 errors
- If you don’t know what a 1124 error is.....
- SYSTEM ERROR: Wrong dbkey in block. Found <dbkey>, should be <dbkey2> in area <num>. (1124)
- HP Server + EMC SAN administered by HP
  - Server & HP/UX diagnostics showed no problems
  - EMC diagnostics showed no problems
  - Cause: Bad SAN Fabric Switch
Basic Disaster Prevention & Planning

- Backups (yes, I know you think you have good backups but have you tested one recently?)
- After Imaging
- Replication of DB and AI files
- Test your Entire Recovery Plan Frequently
Our Advantage over Brain Surgeons

• **Warm Standby Database**
  – A database on another machine with a recent copy of the production DB
    Might also called a DR (Disaster Recovery) Database
  – This is relatively easy to do in Progress
  – Relatively hard to inexpensively license 😊

• **Hot Standby Database**
  – OpenEdge Replication
Disaster Prevention & Response Basics

• Strong security: physical & logical
• ALWAYS have an up-to-date Structure (.st) file
  – Customer suggestion: run prostruct list every time you backup the DB
After Imaging - Why Use It?

• True Horror Story #1
  – “We don’t need AI, we have disk mirroring”
  – A DBA (logged on as root) FTP’d a test database into the directory where the production database resided... unfortunately they had the same name
  – Disk Mirroring worked just fine.....
  – After Imaging would have probably saved the day
After Imaging - Why Use It?

• True Horror Story #2
  – A user ran an archiving program on live data that wasn’t ready to be archived
  – Once again the mirroring performed perfectly

• AI might have improved the situation as it is possible to Roll Forward to a specific point in time
After Imaging - Why Use It?

- True Horror Story #3 – Part 1
- BI file hit the V8 2GB limit @ 1600 on a busy day (300+ users)
- Large Production Database was corrupted
- Progress Support Recommendation: dump & load or restore from backup which meant substantial down time or data loss
After Imaging - Why Use It?

- True Horror Story #3 – Part 2
- Fortunately the customer called me and I was able to temporarily patch the database until a D&L could be performed
- Irony: I had recommended AI to this customer over one year prior to this event
After Imaging - Validation

- Important if you move AI files around
- Added in V10.1B
- `rfutil <db> -C aiverify [ full | partial ]`
  - `-a <AI file name>`
  - `or`
  - `-alist <list of AI files>`
Disaster Prevention - Essential DB Monitoring

• Purpose: to make sure you don’t have hidden or unreported corruption

• Two types:
  – Continuous
  – Periodic
Disaster Prevention - Continuous Monitoring

• Continuous, realtime Corruption Checks
  – MemCheck (minimal overhead)
  – DbCheck (approx 5%)
  – AreaCheck (approx 5%)
  – IndexCheck (approx 5%)
  – TableCheck (approx 5%)
• These are all Database Startup Options
• Can also be turned off/on in promon > R&D > #4 Admin
Disaster Prevention – Periodic Monitoring

• Periodic Corruption Checks

  `proutil dbanalys`    `probkup/procopy`
  `proutil dbrpr`       `proutil dbscan`
  `proutil idxfix`      `dbtool`

• Each option has it’s own limitations

• Record how long it takes to run the process because that info might be useful during a crisis

• See next slide for more info
<table>
<thead>
<tr>
<th>Option</th>
<th>OnLine</th>
<th>Granularity</th>
<th>What Does It Check</th>
</tr>
</thead>
<tbody>
<tr>
<td>dbanalys</td>
<td>Yes</td>
<td>Area</td>
<td>Block &amp; Record validation.</td>
</tr>
<tr>
<td>idxfix</td>
<td>Yes</td>
<td>Single Key</td>
<td>Key-to-Record and Record-to-Key validation.</td>
</tr>
<tr>
<td>probkup</td>
<td>Yes</td>
<td>Entire DB</td>
<td>Block validation.</td>
</tr>
<tr>
<td>dbtool</td>
<td>Yes</td>
<td>SQL Width issues. Database Blocks. Record Validation &amp; fixup.</td>
<td></td>
</tr>
<tr>
<td>dbrpr</td>
<td>No</td>
<td>Various</td>
<td>Chain repair. Delete record fragments. Reformat a block to a Free Block. Fix Bad Blocks. Delete Bad Records.</td>
</tr>
<tr>
<td>dbscan</td>
<td>No</td>
<td>Various</td>
<td>Similar to dbrpr but designed to run non-interactively. Not documented. Does not repair or fix anything.</td>
</tr>
</tbody>
</table>
Essential DB Monitoring – DB Log File

- Check the Database log (.lg) file for errors DAILY (or better yet continuously but that can be expensive).

- Look for words such as:
  
  kill* drastic warn* error system dead fatal abnormal exceed* fail* wrong unexpected* invalid died damage* overflow* violation insufficient missing disappear* corrupt* allow* attempt* cannot enough illegal beyond impossible increase unknown unable stop*

- Note that this is not a complete list but it’s a good start
Prevention – BI Disk Space – Part 1

- **Do Not** run out of BI disk space especially if there is no surplus space elsewhere
- To perform Crash Recovery (a part of the `proutil truncate bi` process), the BI file **will** grow; sometimes 2X or more
- If there is no space for the BI file to grow, Crash Recovery is not possible
Prevention – BI Disk Space – Part 2

- If Crash Recovery partially completes but then crashes, the next Crash Recovery will create an even larger BI file!!!
- Force Access (-F) is the only option (if you don’t have AI enabled)
- Even having AI enabled is problematic
  – Crash Recovery Notes are also written to the AI logs
  – You can’t do `rfutil aimage empty` during Crash Recovery!!
- This is why the –bithold parameter is so important
Prevention – BI Disk Space – Part 3

• Always use the *-bithold* parameter as an extra safeguard
• Set to a maximum of 25-50% of available BI Disk Space
• Frequently check to verify that the threshold is still valid based upon current & projected free disk space
• V9 and above supports Terabyte sized BI Files but extent sizes are still limited to 2gb unless you use the *EnableLargeFiles* option on *proutil* and the file system must be Large File Enabled too
Iceberg! Dead Ahead!

• Is it truly a disaster?
  – What is your definition of disaster?
  – What is the business’s definition of disaster?
  – Is an “Outage” the same as a “Disaster”?  
  – Everyone in the organization needs to be on the same page with regard to the definitions of:
    • Unscheduled Interruption
    • Outage
    • Disaster
  – How-To documentation to respond to each type of event
  – What are the business’s priorities?
Ok, Time to get to work
And we are approx. 50% done with the presentation
Damage Report (from a DB perspective)

• Is the DB Really Down?
  – Is everyone locked out/frozen/etc?
  – Are entries still being written to the DB log?
  – Could it be an AI Stall (-aistall) or BI Stall (-bistall)?
  – Can I promon into the DB?
    • If promon hangs, try promon <dbname> –NL
    • If that doesn’t work, try promon <dbname> -F
    • If either of those options are successful, gather as much data as possible (i.e. screen shots) because the ship is probably going to sink no matter what you do
Initial Steps – Preserve the Crime Scene

• Backup the DB using a non-Progress utility or split mirror backup
  – Not enough disk space for the backup?, shame on you!
  – Hint: allocate a large void DB to “reserve” emergency space

• A backup gives you some time to formulate a plan-of-action (i.e. find your existing plan) and marshal the necessary resources

• If the business deems that it’s more important to take some other action, that needs to be part of the response plan mentioned previously... stressful moments are not the time to make decisions that may have significant consequences
Initial Steps – Preserve the Crime Scene

• Make a copy of the DB log (might be overwritten if DB is restored from backup)

• Preserve other logs that might be overwritten by log cleanup utilities... have a list (or better yet a script) handy so you don’t forget any

• Structure (.st) file, if it’s not already up-to-date, shame on you
Initial Steps

• Make sure **only one** person is driving the process that touches an individual database
  – Air France Flight #447 crashed in the Atlantic because the Pilot & Co-Pilot were performing cancelling actions

• Stop Automated Tasks via *cron* or the Windows Scheduler

• Inform the business/user community (it’s better to have someone else do that...if you get slain, you won’t be able to accomplish much)
Initial Steps – BI Truncation

• Don’t truncate the BI as an autonomic, reflexive response
• Don’t try to truncate the BI file unless you are 100% sure there is enough disk space available to complete Crash Recovery
• Don’t truncate with –F until the crime scene is backed up; too often people come to us after using -F
WhoYa Gonna Call?
Who Ya Gonna Call?

• Unfortunately I don’t think *Ghost Busters* knows anything about Progress

• If you don’t have adequate security permissions (i.e. *root* or *Administrator*), get in touch with someone who does

• If the backup is stored off-site, dispatch someone to get it
  – If it’s stored off-site but not accessible 24/7 that’s just wrong
  – One customer had their DR server in the IT Manager’s basement; IT manager went on vacation and there were issues accessing the DR server
Who Ya Gonna Call?

• If there are some idle hands available, assign someone to take notes about:
  – The steps being taken
  – The sequence of the steps
  – The Success or Failure of the each step
  – Inputs & outputs of each step
  – Changes being made to the system without the benefit of Change Control
  – Recommendations received from 3rd parties and who those parties were

These notes are for the post-mortem process
Who Ya Gonna Call?

• Get someone or something (bodyguard, mercenary, X-Men, crime scene tape, etc.) to shield you from repetitive, time wasting questions like “when is the system going to be back up?”

• But conversely make sure you (or a designated spokesperson) updates the business periodically; don’t keep them in the dark even if your own knowledge is not complete
Who Ya Gonna Call?

- Open a Conference Call Bridge
- Get a Webex, Goto My PC, or similar session fired up
- Open a Support call with Progress Tech Support (PTS)
- Open a Support call with your Application Vendor
- Open a Support call with your Hardware Vendor
- Open a Support call with your SAN Vendor
- See if Dan Foreman or Tom Bascom, et al are available – you can get my contact info on a refrigerator magnet
- Locate & open up this presentation 😊
Time Constraints

• Time required to restore from Backup & Roll Forward versus fixing the problem

• Important bits of knowledge:
  – Do you know how long a restore + Roll Forward takes?
  – Do you know how long it takes to failover to Replicated DB?
  – Do you know how long it takes to fail back from the Replicated DB?
  – Do you know how long it takes to run some of the Progress scanning/repair utilities?
Time Constraints

• Can some activities be performed in parallel?
  – Multiple databases being restored
  – Overlapping steps
    • Restore & Roll Forward
    • Failover to Replicated DB
    • Fix the DB
  – Do you have enough hands (and brains) to safely multi-task? Very Dangerous!!
Database Status

• The App is not working for anyone
• The App is working for some users e.g. reporting users (no trx)
• Things to do:
  – Review DB Logs
  – Some log entries might come from a different DB than the log of the one you are looking at
  – Review AS, WS logs
  – Try connecting to the database directly (self-service client), i.e. not using the App
Database Status

• DB is Down and the Broker won’t start
  – Try `prostruct list` – very lightweight (only reads .db & Schema Area), read only operation
  – Try connecting to the DB with a 4GL Client session in Single User Mode
  – If the Single User Client won’t connect, try using Read Only (-RO)
  – If the Client still can’t connect, truncate the BI file and try again

• DB Broker starts but DB is corrupted

• DB is gone – entirely or parts
Corruption

• What kind of corruption?
  – Internal (Block, Record, Index, Chain, etc.)
  – Missing or damaged extents
  – Logical (e.g. broken parent/child links)

• Is “simple” recovery sufficient?
  – Restore from Backup + Roll Forward
  – Fail over to a Replicated Database
Database Status

- Startable (i.e. multi-user) but corrupted
- Single user (or utility) can access but not multi-user
- No utility or client can access – Game Over... maybe
Remove Java from the Equation
Remove Java from the Equation

• If a DB is normally started with *dbman* or OE Management try starting it from the command line...if you don’t know how, learn now...it’s an essential skill to have
• Try *proserve <dbname>*
• If *proserve* fails, try *_mprosrv <dbname>*
Restart Impossible - Time to Restore

- Can’t start or access the DB
- Restore from backup
- Apply AI Files
- Backup the restored DB
- Re-enable AI
- Sounds simple right?
And......

• Bzzztt!
• Simple Recovery not Possible!
• Debugging the recovery process is beyond the scope of this session
• So now are options are:
  – Repair the DB
  – Dump whatever data we can
Loss of AI Extents – Option 1

- Disable After Imaging with `rfutil aimage end`
- Fix the problem that caused the loss of the AI file(s)
- Backup the database
- Re-enable After Imaging
Loss of AI Extents – Option 2

- Disable After Imaging with `rfutil aimage end`. You may get an error regarding the missing AI Extent but typically AI is still disabled.
- Truncate the BI file with `proutil truncate bi`. You may get an error regarding the missing AI Extent but typically the BI file is still truncated.
- Remove all AI Extents with `prostrct remove`. If that doesn't work, use an operating system command.
- Recreate the original AI Extents with `prostrct add`.
- Backup the database.
- Re-enable After Imaging.
Loss of BI File

- Remove all the BI extents (if any) with an OS command
- Re-create the extents with `prostrct unlock <db> -extents`
- Force access with `proutil <db> -C truncate –F`
- The DB Tainted Flag is set but access should be possible
- The extent of the damage is unknown unless the BUSY AI file at the time of the problem is available to scan with:
  
  `rfutil <db> -C aimage scan verbose –a <extent-name>`
Loss of DB Extents

- Control Area (.db file)
  - prostrct builddb
- Schema Area (.d1)
  - Probably out of luck
- Empty Extent (i.e. above the High Water Mark)
  - prostrct unlock --extents
- If you follow “The Bascom Method” (single variable extent for each Area) recovery is more difficult than multiple, fixed extents
Loss of DB Extents that contain Data

- Extent “Transplant”
- Restore an earlier backup into an identically sized structure
- Copy the missing/damaged extent(s) from the restored DB over the broken DB
- Use `prostrct unlock` to synchronize the extent timestamps
- Note: if the lost extent contained the Area High Water Mark and the Area was actively growing then recovery may not be possible
Limited Corruption – A small number of blocks

• Index Area
  – `proutil idxfix`
  – Rebuild the indexes for that Area (`proutil idxbuild area <area-name>`)  
  – This is why keeping data & indexes in separate Areas is very important
Limited Corruption – A small number of blocks

- Data Area
  - Proutil dbrpr

DATABASE SCAN MENU
---------------------
1. Report Bad Blocks
3. Fix Bad Blocks
4. Report Bad Records
5. Delete Bad Records
6. Dump Records to RM File
7. Rebuild Free Chain
8. Rebuild RM Chain
9. Rebuild Index Delete Chain
10. Change Current Working Area
11. Fix Cluster Chains in Type II Area
A. Apply scan to all areas
Limited Corruption - Block Surgery

- Dump Block with `proutil dbmpr` > 4. Dump Block
- Make a backup copy of the dumped block (.dmp text file)
- Edit the Block
- Load the edited Block with `proutil dbmpr` > 5. Load Block
- You will need to know the DBKEY (i.e. address) of the Block
- Also the location of the data to be edited within the block
- A similar repair might be done by performing a “Block Transplant” from a healthy database
Limited Corruption - Block Surgery – Example

- BI Cluster size accidentally set to a larger than expected value
- The BI Truncation (i.e. crash recovery) ran out of space and there was not enough space to run again
- Edited DB Master Block
  - Changed the Cluster Size to a proper value
  - “Told” the DB that the BI file is currently truncated
- Load the edited Master Block
- Got caught by Big/Little Endian the 1st time but that was easy to fix
Large Amount of Corruption

• dbpr
• idxfix
• idxbuild
• For more detail please refer to
  – Rich Banville’s presentation titled: Database Corruption given during PUG Challenge USA 2013
    – Link http://pugchallenge.org/downloads2013/251_DBCorruption.pptx
Can’t Repair the DB so must try to Dump Data

• Conventional Dump & Load?
• It is unlikely that a dump & load will be completely successful but at least you will know what tables have problems
• How long will it take? Do you know?
• Hint: have a list of tables that don’t needed to be dumped or can be dumped at a later time
  – Scratch tables
  – Unused tables
Dump Options

- **Indexless** Binary Dump (-index 0) – works with Type 2 Areas only
- 4GL TABLE-SCAN option – also for Type 2 Areas only
  - TABLE-SCAN does work with T1 Areas but uses an index
- Problem: an indexless dump won’t guarantee the ordering of the data so when you hit the inevitable problem you won’t know where in the table the corrupt record lies, i.e. close to the beginning or end of the table
Binary Dump Limitations

- Binary dump – can’t use text utilities to review the output data
- NOTE: A binary dump & load can sometimes (albeit rarely) transport corrupted data without error during the dump/load/index rebuild process
- A multi-threaded Binary Dump is not possible if the Broker won’t start
Dump Options

• Custom 4GL dump process avoids the corrupt data
  – 4GL WHERE Phrase (do you have a 4GL license that supports this?)
  – Run till it crashes, edit the .d file, and start from the other “end”

• Dump using a non-primary index

• Recid Dump – Slowwwwwww......

• Read Only (-RO) 4GL/ABL Dump

• Ultra Radical: truncate the Area(s) that are causing the dump processes to fail (good reason to have nothing in the Schema Area)
Post Mortem

• Assuming that you’re still employed, document the incident
  – Actions taken & the sequence of those actions (that’s why a note taker is useful)
  – What actions worked
  – What actions didn’t (so they doesn’t get repeated in the future)
  – Incident numbers issued by any support organizations that were involved
  – Resources required
    • Disk space, etc.
    • People, permissions, access
How Can We Avoid This Happening Again?

• Enhanced monitoring
• Better logging
• Better documentation
• Newer version of Progress
• A bigger budget for:
  – More disk space & better storage management tools (i.e. split mirroring)
  – Disaster Recovery Server & Required Licenses
  – Better maintenance agreements (shorter SLA) or agreements with more competent resources
• Etc.
Resources

- Download the **Captain, Where should I go? What should I do?** presentation given on Wednesday
- Dan Foreman’s *Progress Database Administration Guide*
- Dan’s Mobile: +1 541-908-3437
- Email: danf@prodb.com
THANK YOU!

• Don’t forget your Evaluations!
• Please don’t forget to thank the conference organizers! Organizing & running a conference with 500+ people is a BIG, difficult job!
Questions?