



Transaction Processing with btrieve

May-11

Presenter – Wes Hein
Title – Owner & Director of Product Development

Company – BIAS Software

email: wes@biassoftware.com

Website: Biassoftware.com

- **Transaction Processing using btrieve**
 - Benefits
 - utilizes 'rollbacks' like SQL
 - Intermediate step to SQL conversion

- **Transaction Processing using btrieve**
 - Drawbacks
 - Utilizes 'rollbacks' like SQL
 - Can cause unending record lock issues
 - Transaction mode, transaction begin, cache strategy, error behavior strategy and all their interactions now matter
 - Need a thorough understanding of error handling

- **Transaction Processing using btrieve**

Do not turn on transaction processing until you understand how it impacts your system design.

Systems developed without transaction processing in mind WILL NOT WORK AS EXPECTED!

- **Basic strategy - Filter, find, edit**
- Agnostic counters
- Updates to linked files should not be done interactively, but at the end of the transaction. Required different overall strategy of record updates. (check counter, receipt counter, cash receipting #)
- Transaction mode, transaction begin, cache strategy, error behavior strategy and all their interactions now matter
- Extensive use of memory tables

- More error handling required – not sure I have a complete understanding – show adv payroll groups with restart error handling.
- Get the value but don't update until completion of transaction
- One task to do one job
- If you want an outside batch task to see changes made to a table in a transaction, you need to flush the record, exit the record, then call the batch task in the record prefix (also in the task suffix if the user can exit without going to a new record). Include the control to go back to if desired.

- Deferred transactions work well but bring ‘baggage’ with them
- ALL sub tasks and programs must be ‘within active transaction’ or data cannot be seen!
- You cannot call a deferred transaction from a physical transaction – although you can do the opposite (usually with bad results!)
- I have even used nested deferred!



THANK YOU