# Managing Migrations With Informix ER

## With Tom Beebe

tom@advancedatatools.com

***Webcast on July 18th 2019***

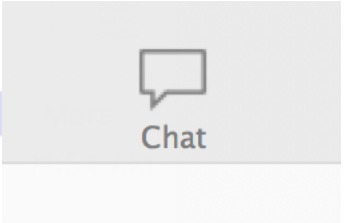
# Tom Beebe

Tom is a Senior Database Consultant and has been with Advanced DataTools for over 10 years. He has been working with Informix since college with a long time fondness for open source languages. Tom is the lead consultant for Networking, Unix System Administration and Web Development needs. Currently, he is the Project Manager and lead developer on a variety of Web Development projects. Tom is IBM Information Champion Award.

**tom@advancedatatools.com**
**www.advancedatatools.com**
**703-256-0267 x 106**

# Webcast Guidelines

- The Webcast is being recorded.  The Webcast replay and slides may be available in a few days.
- Please Mute your Line. Background sounds will distract everyone.
- Use the Chat Button in the upper right to ask questions.



# About This Talk

- Almost all DBAs need to do a migration at some point
- As more people move from UNIX systems to Linux it is very time intensive
- Downtime is precious and rare
- Enterprise Replication allows for real time data migrations between environments

# What is Enterprise Replication

- Asynchronous log based replication
- While HDR is instance level replication, ER is table level replication
- Operating System and Informix Version agnostic
- Can have multiple replicate types
- Allows for update anywhere or primary target replication

# Consider ER For Migrations

- Allows for direct replication between different environments
- Allows for dynamic testing/dev environments without impacting prod
- Migrate without downtime
- Integrates with connection manager
- Can keep the old system running and cut over when needed

# Other Positives

- Supports all data types (version dependent)
- Supports time based replication to limit network use during peak periods
- Can be repaired rather than recopied to catch up
- A number of new 14.1 features

# Considerations

- All databases need to be logged
- All tables need to be logged (no raw)
- All tables need a primary key, shadow column or a unique index
- Needs a replicate created for each table
- Servers need to be trusted and able to communicate

# Standard Replication

- Replicate table A to table B – one direction

```
cdr define repl -C always –S tran –A –I foo_repl \
“P prod@old_db:informix.foo” “select * from foo” \
“R prod@new_server:informix.foo” “select * from foo”
```

- Will take all data insert into foo on old_db and also insert it on new_server
- Can use to just copy, or drip feed

# Update Anywhere

- Replicate between old_db and new_server keep updates consistent across both

```
cdr define repl -c timestamp –S tran –A –I foo_repl \
“P prod@old_db:informix.foo” “select * from foo” \
“R prod@new_server:informix.foo” “select * from foo”
```

- Useful if doing a slow move or wanting to run both in parallel for a while

# Loopback

- Introduced in 12.1xc11
- Can join a server to itself
- Can replicate a table to another table on the same server
- Useful for creating cloned copies of data for migration or testing

# ER Versus SQL Copy

- Both are platform agnostic
- SQL copy takes far less setup, just one sql line per table
- ER is more flexible
- ER can be restarted, scheduled and continually sending data
- ER will support Smartblob data without needing to unload to disk
- ER Makes copying Timeseries data easier

# Initial Steps To Set Up ER Migration

- Set up trusted hosts between the two
- Use cdr define server to set up a domain on the old server and join the new server to it
- Make sure all tables have primary keys or erkeys set up
- Use dbschema on the old server, run it against a new clean database on the new server

# Using ER For Initial Data Copy

- Set up replication on all tables
- Start replicates
- Use sync command to transfer data
- Data will remain up to date
- Warning: make sure to order the syncs based on foreign key dependencies
- If primary is still active you may need a few update passes

# Quck Replicate Everything

- Many ways to do this, here is one
- Unload to replicate.sh

  select tabname, owner from systables where tabid > 99 and tabtype = ‘T’;

On the command line

```
perl -p -i -e 's/(.*)\|(.*)\|/cdr define repl -C always -M
<OLD_SERVER> -S row repl_\1 "P
<DATABASE>@<OLD_SERVER>\2.\1" "select * from \1" "R
<DATABASE>@<NEW_SERVER>\2.\1" \"select * from \1"\n cdr
start repl repl_\1/' replicate.sh
```

# Once Replication Is Up

- Can set up a replicateset container of all replicates to allow all commands to be run once
- Run cdr sync on replicates to do an initial data copy
- cdr sync replicate –m <old_server> -r <replicate name> <new_server>

# Different Ways To Use ER For Migrations

- Use it for an ongoing data migration
- Copy an initial set of data in advance and keep the new server up to date
- Trickle feed the new server

# Copy An Initial Data Set

- Use methods learned in the prior migration webcast to set up the data and populate it with an initial set of data
- Set up replicates for all of the tables
- Do not start the replicates yet
- For each table start the replicate and then run a repair to sync up any missing or altered records

# Trickle Feed

- Useful if migrating slowly and bandwidth is at a premium
- Set up the new empty database
- Set up replicates
- Set up time based replication to only replicate at set off hours
- Or do copies as able, and let new and updated rows come over as they happen
- Then when ready to look at going live use repairs to sync up the remaining rows

# Repairing Records

- Cdr check repair command
- Can be slow if not using shadow columns to speed it up
- Will check checksum of every row and update any that don’t match the primary
- By default will delete and rows that are missing from the master
- Can use a where clause to limit the scope

# Cdr check repair

cdr check repl --master=<old_server> \
-R --repl=repl_table <new server>

If using LOBs look at –skipLOB to speed up processing

# Connection Manager

- Can be set up against ER groups
- Can be used to have all clients to point to a standard single location that when go-live is ready can just point over to the new server
- Can set up load balancing against the old and new servers with update-anywhere ER setups
- Can make a migration occur without needing to modify any client settings!

# CDR Migrate Server

- New feature in version 14.1
- Needs at least one server to be on 14, the other server can be 12 or 11.x
- Used to automate the database migration process

# Features

- Define the domain
- Add erkeys to any tables without a PK or unique index
- Create storage spaces for ER if they don’t exist (from a storage pool)
- Create the target database
- Create the replicates
- Synchronize the data between servers

# Prerequisites

- Servers must be set up trusted and both in each others sqlhosts file with groups set up
- A storage pool be set up if ER smart blob spaces are not created
- Recommended that erkeys/PK/Unique indexs already exist since it can be slow to create

# Limitations/Notes

- Does not support Timeseries between versions
- Does not update the starting serial value on target tables, need to manually set where you want it to start

# Phases

- define_er : Define ER
- add_erkey : Add ERKEY to source schema
- create_spaces : Create storage spaces using storagepool
- create_schema : Create database schema
- create_replicates : Create replicate, replicate set and grid definitions
- sync_data : Synchronize data
- all : Execute all phases

# Shadow Columns

- ERKEY – Adds a unique key in situations of no PK or Unique Index
- CRCOLS – Adds extra columns to track changes to the row, needed for timestamp conflict resolution
- REPLCHECK – Adds extra hidden column that has a row checksum to speed up cdr check commands

# Adding Shadow Columns

- ALTER TABLE customer ADD CRCOLS;
- ALTER TABLE customer ADD ERKEY;
- ALTER TABLE customer ADD REPLCHECK;

# Downsides To ER

- Not as fast as traditional backup/restore or other binary methods
- Doesn’t work with unlogged systems
- If using vastly different versions check the docs for any notes if using newer data types
- Not guaranteed to be real time data due to being asynchronous

# Questions?

Send follow-up questions to
info@advancedatatools.com

# Informix Webcasts

**from the IBM Informix Champions**

- **Database Indexes - Best Practices for Informix DBAs by Lester Knutsen - Thursday, August 29, 2019, at 2:00 pm EDT**
  - - What is an index?
  - - How do you improve performance with an index?
  - - What is the performance cost of an index?
  - - How do you speed up creating indexes?
  - - How do locks affect indexes?
  - - How to monitor an index to see if it is needed or used?
  - - When is an index unnecessary?
  - - When is an index needed or required?
  - - What is a clustered index?
  - - How do you maintain indexes for peak efficiency?
  - - What is a functional index?

Registration and more information:
https://advancedatatools.com/Informix/NextWebcast.html

# Informix Training
# Update for Informix 14.1

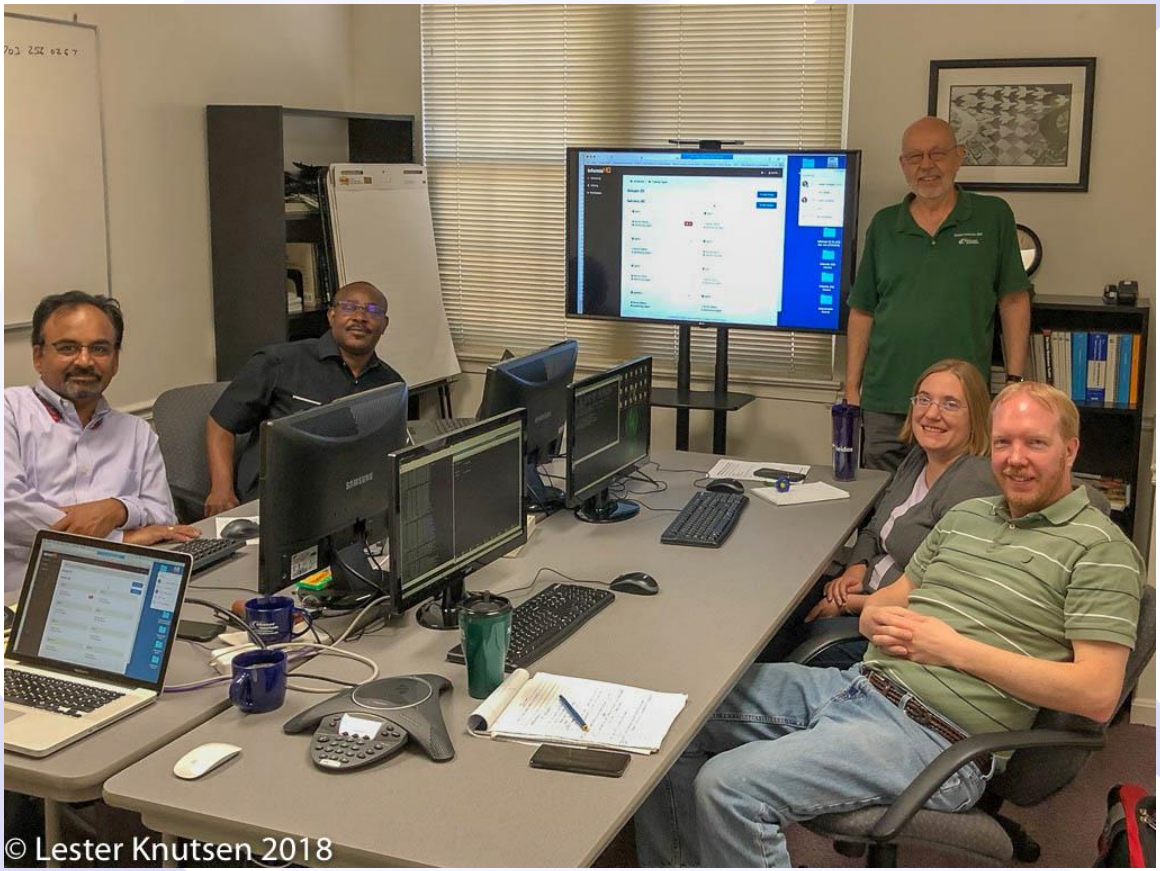
Attend classes online on the web, or in person at our training center in Virginia. All you need is a web browser to connect to our WebEx training system and an SSH client (like Putty) to connect to our training lab for hands-on exercises. Each student uses an 8-core Linux server, with 16GB RAM, SSD drives with Informix 12, and several large databases for benchmark exercises.

- **October 7-10, 2019 - Informix for Database Administrators**
  - This course is for new database administrators, programmers, and technical support personnel who will be setting up, managing, and tuning IBM Informix databases.

- **More Information and Registration at:**
  http://www.advancedatatools.com/Training/InformixTraining.html

Advanced DataTools

# Informix Training Servers




Each Student in class will have a server running Informix 12.10 with:

- 8 CPU Cores
- 16 GB RAM
- 1 SSD Disk
- 1- 4 Disks


## *Informix Support and Training from the Informix Champions!*

Advanced DataTools is an Advanced Level IBM Informix Data Management Partner, and has been an authorized Informix partner since 1993. We have a long-term relationship with IBM, we have priority access to high-level support staff, technical information, and Beta programs. Our team has been working with Informix since its inception, and includes 8 Senior Informix Database Consultants, 4 IBM Champions, 2 IIUG Director's Award winners, and an IBM Gold Consultant. We have Informix specialists Lester Knutsen and Art Kagel available to support your Informix performance tuning and monitoring requirements!

- ***Informix Remote DBA Support Monitoring***
- ***Informix Performance Tuning***
- ***Informix Training***
- ***Informix Consulting***
- ***Informix Development***

**Free Informix Performance Tuning Webcast replays at:**
http://advancedatatools.com/Informix/Webcasts.html
Email: info@advancedatatools.com
Web: http://www.advancedatatools.com

# Thank You

***Advanced DataTools Corporation***

For more information:

**info@advancedatatools.com**

**http://www.advancedatatools.com**