Database Integrity
Looking for Change

Don Katnik
Maine Dept. of Inland Fisheries & Wildlife

OFWIM 2012 – Austin, TX
Manual Error Detection

- Run queries to find errors
- Requires multiple queries
- Can be tedious
- Saved queries can be automated

OFWIM 2010: "Integrity Tool" = a set of saved queries

OFWIM 2012: queries to test integrity of a database that changes frequently

Database Integrity: Looking for Change • Don Katnik, Maine Dept. of Inland Fisheries & Wildlife • Austin, TX 2012
Database Change

Intentional:

Add new records
Delete existing records
Edit parts of existing records

Unintentional

Create duplicates when trying to add a new record
Accidentally delete a record
Accidentally edit a record
Sometimes change multiple records when you meant to change just one
Share copies (or exports) of a database

"Static" derived products: reports, tables, maps, etc.

Represent the data as it existed previously

To reconcile these "dated" products, sometimes need to see an earlier version of the data (as it existed when they were created)

Helpful to keep an archive (daily, monthly...)

A database archive is one critical part of testing for change

Feature-level metadata tells if the change was intentional
Testing Change – Decision Tree

Compare “Current” vs. “Previous”

- Exists Now, Not Before
  - Is there “ADD” Metadata?
    - Yes – OK
    - No – ERROR
  - Exists Before, Not Now
    - Is there “DELETE” Metadata?
      - Yes – OK
      - No – ERROR
  - Exists Now & Before
    - Are both versions the same?
      - Yes – OK
      - No – Is there “Modify” Metadata?
        - Yes – OK
        - No – ERROR

Database Integrity: Looking for Change • Don Katnik, Maine Dept. of Inland Fisheries & Wildlife • Austin, TX 2012
Change Over Time Example

1. AFS
2. ROCKS
3. MY
4. WORLD

1. TWS
2. ROCKS
3. MY
4. WORLD

1. TWS
2. OFWIM
3. OFWUM
4. WORLD

1. TWS
2. OFWIM
3. OFWUM
4. WORLD

1. TWS
2. OFWIM
3. OFWUM
4. WORLD

1. TWS
2. OFWIM
3. OFWUM
4. WORLD

Database Integrity: Looking for Change • Don Katnik, Maine Dept. of Inland Fisheries & Wildlife • Austin, TX 2012
Value 1 did not exist then
Cannot compare to current
Pairwise Comparisons – Which?

Value 1 is identical
Looks OK!
**Pairwise Comparisons – Which?**

**Then**
1. AFS
2. TWS
3. OFWIM
4. OFWUM

**Now**
1. OFWUM
2. ROCKS
3. MY
4. WORLD

---

Database Integrity: Looking for Change • Don Katnik, Maine Dept. of Inland Fisheries & Wildlife • Austin, TX 2012
Compare Against ALL

Compare Current version against ALL archive versions

!!! Comprehensive 😊

Continually adding archives, test will take longer & longer to run

Value comparisons – maybe not a problem?

Spatial (geometry) comparisons – BIG problem 😞
Serial Pairwise Comparisons

First Archive

If I'd thought of this sooner...

If I'd thought of this sooner...

Database Integrity: Looking for Change  ●  Don Katnik, Maine Dept. of Inland Fisheries & Wildlife  ●  Austin, TX 2012
Each object is stored as it existed when last "correct"

"Correct" = change documented with metadata

Documented changes updated into LDC automatically

Metadata (and change date) is part of LDC record

"Current" database only needs to be compared against LDC

LDC schema same as "current" database except for a metadata ID field that references the metadata record associated with the last documented change
Testing Change – Decision Tree

Compare “Current” vs. “Previous”

Exists Now, Not Before

Exists Before, Not Now

Exists Now & Before

Is there “ADD” Metadata?

Yes – OK

No – ERROR

Is there “DELETE” Metadata?

Yes – OK

No – ERROR

Are both versions the same?

Yes – OK

No – Is there “Modify” Metadata?

Yes – OK

No – ERROR

Update Object in LastDocChange (LDC)
Creating an LDC Database

If the database is new, create the LDC at the same time (easiest)

If the database to be tested already exists:

Option 1. Make today "Time Zero" and test changes going forward

The LDC = primary database at Time Zero (today)
Easiest option for an existing database
Starting point is arbitrary
Prior errors persist - acceptable?

Database Integrity: Looking for Change • Don Katnik, Maine Dept. of Inland Fisheries & Wildlife • Austin, TX 2012
Creating an LDC Database

If the database to be tested already exists:

Option 2. Select a "meaningful" date for Time Zero, then test change going forward from that time

Simplest: Jan 1, start of fiscal year, etc.
Date of significant modification of database
"Error overhaul" date
Creating an LDC Database

If the database to be tested already exists:

Option 3. Go back to earliest archive and build LDC from there

- Most difficult option
- Only has to be done once
- Most thorough
- No "before" vs. "after" versions of data
- Assumes archives (and metadata) exist
Testing "In Use" & Future Work

Archive and test geodatabase nightly
Evaluate/fix errors every 1-2 times/month
Integrity test/LDC update scripts converted from VBA to Python
Error handling tools/forms still in VBA
Have not applied concept to Access relational database

Database Integrity: Looking for Change • Don Katnik, Maine Dept. of Inland Fisheries & Wildlife • Austin, TX 2012
Conclusions

Testing for unintentional changes is critical to maintaining database integrity – especially for data that changes often and is shared with external users

2 “Gotta Haves”:

- Regular archiving or Last Documented Change database
- Feature-level metadata documenting changes

Best to incorporate concept into initial design of a new database

Still an option for an existing database

Actual script that runs test is not complicated