



*statistic shouldn't be a four letter
word*

how statistics can help
manage data in GIS



Overview

- Description of problem
- Proposed solution
- Implementation details
- Examples



Description of Problem

- Automatic loading of large amounts of data
- Manual cleanup of errors
- Even tiny error rate can mean massive post-load cleanup



Data

- Large datasets: e.g. all streets in US, all equipment for 100m customers
- Low error threshold
- Manual cleanup to be minimized



Wanted:

- Means to quantify error rates
- Means to diagnose errors
- Means to direct cleanup effort



Proposed Solution

- Second pass of data
- Check to ensure data was loaded
- Count successes and failures
- Provide metrics
- Generate “reload” files
- Keep it simple



Second Pass

- Use different code, not loader code
- Re-examine data files



Check to ensure data was loaded

- Check to see did a unique value “make it”
- If simple, check if location is correct



Count successes and failures

- Need to calculate % success.
- Need totals broken down different ways



Provide Metrics

- Users need direction
- Developers need diagnostics
- Management need totals



Metrics for users

- Will want to break cleanup work up:
 - by area (Joe does Pasco county)
 - or
 - by type of object (Joe does all rivers)
 - or
 - by class of object (Joe does all water)



Metrics for Developers

- Need means to diagnose errors
- Want to know what was common to the 3% of highways that failed to import
- Same breakdowns as users
- Different means of navigation – need to quickly determine that the 3% were all in a county or all on-ramps



Metrics for managers

- Need totals - % success for entire process
- Need subtotals to monitor cleanup



Generate "Reload" files

- For each error generate a file that can be used to reload the data
- Allows developer to retest
- Allows users to see missing data for manual reconstruction



Keep it simple

- Second pass takes longer
- Doesn't load data
- Overhead



Implementation details

- Directory structure for metrics/reload
- Filter out **all** lines required for reload
- Use excel to do the work
- Ignore OLE if possible
- Distill the information many ways
- Provide navigation links



Directory structure for metrics/reload

- Generate directory structure to make data navigable:

E:\Verizon.production.STATS\CMDNNJCE\CMDNNJCE;06649.25;Cisco_5500

- Allows users and developers alike to find error information



Filter out all lines required for reload

- One error might fail a batch of lines or whole portion of a file
- Need enough to reload missing and make translator work
- Easier to remove duplicates than recreate manually



Use excel to do the work

- Excel has every function necessary
- Place initial row with summation functions
- Add data rows as they are generated



Ignore OLE if possible

- OLE is Slow
- Can make a tab delimited text file called whatever.xls and excel will open without dialog.
- Can place functions in excel using text



Distill the information many ways

- Users and Developers may need data summed in different ways:
 - All water features by county
 - All rivers by county
 - All features in a particular county
 - All rivers/water features by State
 - Etc.
- Especially useful for developers



Provide navigation links

- Allow users and developers to get from one file to the next using hyperlinks
- Can be implemented simply in excel:
=hyperlink(“path\file”, “What to display”)



Examples





Conclusion

- Users can proceed with cleaning up data in an order that makes sense to them
- Developers can aid the data loading process by quickly identifying places where development effort would be best served
- Managers have a tool that allows the process to direct itself, freeing them to concentrate on guidance.



Questions?

John Miller

Managing Principal Consultant

Red Planet Consulting

Jmiller@RedPlanetConsulting.com

303.478.5770