

BIOGRAPHICAL INFORMATION

John Miller
Managing Principal Consultant
Red Planet Consulting inc.

Specific responsibilities:

Responsible for managing four client accounts.
Co-founder of company in 2000. Director.

Past Experience:

Worked for many Gas, Electric, Telecommunications and Water clients in both Europe and North America.

Educational Information:

B.A. Mathematics Trinity College Dublin
BAI Engineering Trinity College Dublin

Professional Memberships:

none

STATISTIC SHOULDN'T BE A FOUR LETTER WORD
HOW STATISTICS CAN HELP
MANAGE DATA IN GIS

John Miller
Red Planet Consulting

4999 Pearl East Circle
Suite 300
Boulder, CO 80301

ABSTRACT

This paper will outline techniques for managing both data loading and data cleanup. The basic effort is to provide both management and worker with accurate metrics of the process at hand. While the needs are often subtly different, analysis and care taken upfront can produce a system that effectively allows the workers to manage the process themselves.

Management effort can be freed from oversight to concentrate on ensuring that the process is functioning properly and applying changes to it where necessary.

Typically very large datasets or very strict requirements produce projects that do not lend themselves well to conventional management techniques. Real examples from data migration and import processes which combine both traits will be used.

DESCRIPTION OF PROBLEM

When automatically loading of large amounts of data, manual cleanup of errors is required. Even tiny error rate can mean that a massive post-load cleanup effort is required.

Data

Examples of the large datasets that can cause this problem include all streets in US as landbase or all equipment for 100m customers as plant data. A low error threshold requirement is also needed.

Requirements:

In order to manage the process a means to quantify error rates is needed. A means to diagnose errors will provide a rapid turnaround in loading errors. Additionally a means to direct cleanup effort will allow management to concentrate on guiding the process rather than needing to actively manage it.

PROPOSED SOLUTION

The proposed solution is to use a second pass of the data. This will be used to check to ensure that the data was loaded correctly. In addition the amount of successes and failures will be counted. Metrics of the success and failure (broken down in useful ways) will be provided to allow all interested parties to monitor the process. "Reload" files will be generated for the portions of the load that failed. Since this is an overhead task, not directly contributing to the loading of data the effort will be focussed on keeping it simple.

Second Pass

It is important to use new code (preferably by a different developer), rather than loader code. This avoids systematic errors. This new code opens all the files that were opened as part of the original data load.

Check to ensure data was loaded

The data being loaded has unique fingerprints that can be tracked from load file to destination database. Checking to see if a unique value in the data file is now available in the database constitutes the success/failure test. If it does not require too much additional processing a check to see if the location is correct can also help.

Count successes and failures

The ultimate goal in generating metrics is a % success figure. By "bean-counting" each success and failure it is possible to generate these statistics both in an overall sense and broken down in many useful ways.

Provide Metrics

Different parties are interested in different aspects of this information.

Users need direction – what should I start cleaning next? Where is it? What should it look like?

Developers need diagnostics – why didn't it load? What is unique to the failures?

Management need totals – Who should work on which errors?

Metrics for users

Users generally want to break cleanup work up either between a team or for work management:

by area i.e 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

Developers need a means to diagnose errors. Developers need to know what was common to the 3% of highways that failed to import. Developers need the same breakdowns as users but need a different means of navigation. Developers need to quickly determine that the 3% were all in a single county or all on-ramps or something like that. In addition, developers need a method to allow the problem data to be loaded again for testing.

Metrics for managers

Managers need totals (% success) for entire process. They also need subtotals to monitor cleanup and development efforts as well as progress from load to load. Detail is unnecessary.

Generate "Reload" files

For each error a file needs to be generated that can be used to reload the data. This will allow the developer to retest. It also allows users to see missing data for manual reconstruction.

Keep it simple

A second pass takes longer. It doesn't contribute directly to loading of data and as such constitutes overhead.

IMPLEMENTATION DETAILS

A directory structure for metrics and reload files is used to simplify navigation and aid in the cleanup and development effort. Reload files can require more lines from the data file than just the erroneous ones. Excel can be used to do the calculation work. The information can be presented in many different ways. Navigation links are a must to allow easy traversal of the information.

Directory structure for metrics/reload

Generate directory structure to make data navigable e.g.
E:\Verizon.production.STATS\CMDNNJCE\CMDNNJCE;06649.25;Cisco_5500
This allows users and developers alike to find error information quickly.

Filter out all lines required for reload

One error might fail a batch of lines or whole portion of a file. The loader might require meta information from the top of a data file. It needs enough to reload missing and make translator work. It is easier to remove duplicates than recreate manually so having excess data in the reload file isn't necessarily a bad thing.

Use excel to do the work

Excel has every function necessary for statistics generation. When writing the files an initial row with summation functions can be added and then add data rows as they are generated.

Ignore OLE if possible

OLE is Slow. It is possible to make a tab delimited text file called whatever.xls and excel will open this file without a dialog. It is possible to place functions in excel using a text file.

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.

This is especially useful for developers as it provides a means to rapidly identify a commonality between the data causing the problems. This can greatly aid in debugging the load process.

Provide navigation links

Allow users and developers to get from one file to the next using hyperlinks. Excel has this functionality built in:

=hyperlink("path\file", "What to display")

Examples

This shows the numbers (in red) that management are interested in reporting as an overall load metric:

Template	HECI	# CLLI	# lines in mstrept	# CFA lines in mstrept	# unique CFA in mstrept	# CFA in SW	# CFA missing	CFA % Loaded	# EQCU lines in mstrept	# unique EQCU in mstrept	# EQCU in SW	# EQCU missing	EQCU % Loaded
Total:	80	19	23664	13114	13010	12395	615	95.27287	401	401	355	46	88.52868
Siemens EWSM Switch SMDS T3QASMSRA	56	2032	1929	1929	1752	177	90.82426	103	103	91	12	88.34951	
Cascade STDX 9000 CNQKCG01RA	51	4512	3361	3361	3211	150	95.53704	68	68	66	2	97.05882	
CBX 500 BAM2EE0BRA	104	6418	6184	6184	6084	100	98.38292	64	64	59	5	92.1875	
Cisco 7206 Router (Shelf) CNM2BC0ARA	29	315	189	143	63	80	44.05594	12	12	11	1	91.66667	

Looking at these numbers allows the managers to direct development effort to some problems and leave other problems for cleanup. Typically data with high %success can be more resistant to incremental improvement than ones with low %success. To describe this another way, each successive bug found in a particular data type usually fixes less data than the previous bug fix. In this case the first three lines all will require 100 or more operations to manually clean up. Nevertheless, all three lines are above 90%.

In directing development effort it might be worth fixing "Cisco 7206 Router (Shelf)" before "Siemens EWSM Switch SMDS" first.

Template	HECI	# CLLI	# lines in mstrept	# CFA lines in mstrept	# unique CFA in mstrept	# CFA in SW	# CFA missing	CFA % Loaded
Total:	80	19	23664	13114	13010	12395	615	95.27287
Siemens EWSM Switch SMDS T3QASMSRA	56	2032	1929	1929	1752	177	90.82426	
Cascade STDX 9000 CNQKCG01RA	51	4512	3361	3361	3211	150	95.53704	
CBX 500 BAM2EE0BRA	104	6418	6184	6184	6084	100	98.38292	
Cisco 7206 Router (Shelf) CNM2BC0ARA	29	315	189	143	63	80	44.05594	

Links allow a developer clicking on:

CNM2BC0ARA

To be taken directly to a file describing the distribution of that data type across each area being loaded. This helps diagnose individual errors.

Conclusions

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.