CIFE CENTER FOR INTEGRATED FACILITY ENGINEERING

Case Study One: Programming in Field Engineering Quantity Collection

By

Forest Peterson & Martin Fischer

CIFE Working Paper #WP113
April 2009

STANFORD UNIVERSITY

If you would like to contact the authors, please write to:

c/o CIFE, Civil and Environmental Engineering Dept.,
Stanford University
The Jerry Yang & Akiko Yamazaki Environment & Energy Building
473 Via Ortega, Room 292, Mail Code: 4020
Stanford, CA 94305-4020

CC2009 - Case Study One: Programming in Field Engineering Quantity Collection

Forest Peterson, Graduate Research Assistant, Dept. of Civil and Env. Eng., Stanford Univ., e-mail: granite@stanford.edu

Martin Fischer Ph.D., Professor, Dept. of Civil and Env. Eng., Stanford Univ., and Director, Center for Integrated Facility Engineering, e-mail: fischer@stanford.edu

CIFE, Stanford University, The Jerry Yang & Akiko Yamazaki Environment & Energy Building, 473 Via Ortega, Room 292, MC:4020, Stanford, CA 94305-4020

Keywords: Field engineer, Programming, Project Control, Quantities

Abstract: The question guiding this study is can field engineers on large infrastructure and industrial projects automate repetitive tasks through the use of programmed functions. Field engineers have repetitive tasks, some of these are endogenous tasks and others are exogenous tasks. Endogenous tasks are those consisting of calculating values through recipe-formulas and exogenous tasks being collecting outside measurements. There has been research in the past few years on open source software and programming language aimed at nonprofessional or semi-professional programmers. This paper is intended to target below that scope and address the non-programmer. Macros are a programming interface which repeats a predetermined sequence of operations and are well suited for often repeated tasks. This topic is important because many construction engineers are not aware what macros are and do not write them for their repetitive tasks. This study adds to science knowledge in that recipe-formulas are shown to be a tool available to field engineers that possibly could be standardized and automated.

The study is to learn how to use macros. Write a macro to complete several field engineer tasks and find if it is a net benefit to the engineer. A field engineer task is to verify haul truck invoices are correctly calculated using the fuel surcharge for a given week. Two macro tools are used to accomplish this task, one provided as part of Microsoft Office and the other a free tool available online. A function is coded to pull the weekly average fuel rate from a US government website, apply a recipe formula to calculate the percent surcharge the check the invoice total cost.

The macros were able to pull the needed information from the government website and calculate the percent surcharge. Individually neither macro tool could accomplish the field engineers task. Combined and applied to those functions they could perform they were able to accomplish the task.

The purpose of the study is achieved. Macros are able to perform an exogenous field engineer task and several endogenous tasks. The difficulty in performing these tasks with macros was unexpected. Using two vendors software tools provided results that indicate the difficulty is not the macros themselves but functionality issues with each specific tool. This study is limited in the number of macro tools tested and the field engineer tasks automated.

In practice the implication is that field engineers are likely performing tasks that could be automated with a macro. The results indicate that adding macros to construction engineering courses as part of lab assignment solutions would provide a benefit in their employment. More research is needed to explore how macros could be introduced into existing course lab assignments. Additional research is needed to define criteria to identify tasks most likely to benefit from automation. Exploring the establishment of a lunch box or shared library of common recipe formulas encoded as macros.

Introduction

Field engineers on large projects have repetitive tasks that must be completed routinely. Some of these tasks could be automated using macro programming applications, resulting in fewer endogenous tasks and more emphasis on exogenous tasks. Endogenous tasks are those consisting of calculating values and exogenous tasks being collecting outside measurements.

Macros are applications that will repeat a predetermined sequence of operations, well suited for often repeated field data manipulation tasks. Engineers often utilize spreadsheets for these tasks. Many construction engineers both recent graduates and those with more experience are not aware what macros are and do not write them for small repetitive tasks. As an engineer myself, I had heard of these tools and was aware they could simplify data manipulation steps that were done the same every time. With no time to learn these tools and having not been taught in undergraduate studies, the opportunity to exploit these tools was lost. The broad use of macros on projects to derive quantities would result in better use of engineers working on those things that are not able to be completed by macros.

There has been research in the past few years on "open source" software and programming language aimed at nonprofessional or semi-professional programmers. This paper is intended to target below that scope and address the non-programmer who can still benefit from some *programming light* in their daily life. Here at the Center for Integrated Facility Engineering (CIFE) Timo Hartman has been a visionary of open source code and wrote on integration of the process model through XML[i]. David Chua of the National University of Singapore published an article also outlining the use of XML to create an application that would automate some tasks of the scheduler in creating a realistic process model[ii]. Again here at Stanford Jinxing Cheng investigated the application of Process Specification Language (PSL) as a method to integrate various project management software tools[iii]. All of these authors' works highlight the need for smaller scale programming within the project office to integrate multiple software tools that otherwise function well but provide a more powerful tool when combined. Though these authors are addressing more advanced problems with more advanced tools. There is always object oriented programming such as C++ and JAVA available but these require time and training to gain proficiency. Here is where macros value becomes apparent, an engineer can learn their use in several hours, compatible software is already installed on their machines and macros can perform fairly powerful tasks. What macros are, how macros could be applied and an example of a specific macro application is described in the following three sections.

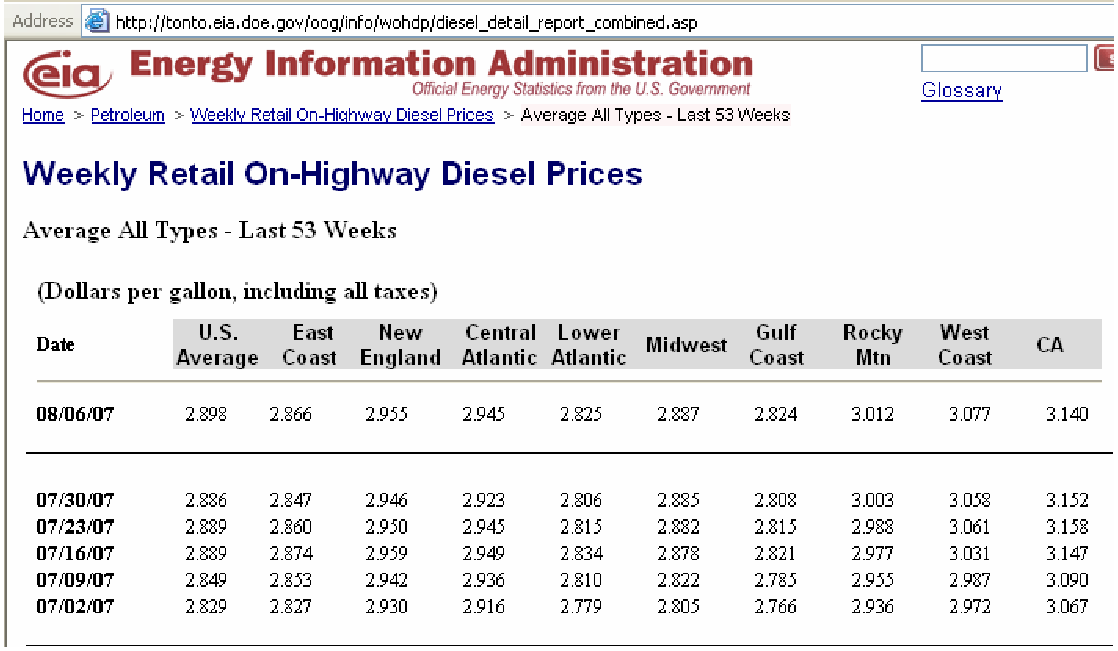
What is a Macro?

At its most basic definition, a macro repeats a predetermined sequence of instructions. These instructions can be keystrokes, mouse movements and lines of program instruction. Macros are described as "[macros] allow short sequences of keystrokes and mouse actions to be transformed into other sequences of keystrokes and mouse actions. In this way, frequently-used or repetitive sequences of keystrokes and mouse movements can be automated."[iv]

Activities that are repetitive are good candidates for a macro. Many of the activities in field engineering are repetitive and endogenous, meaning there is no outside data involved. There are relatively few actions that are exogenous, outside collected data, or requiring non-repetitive actions. Examples of repetitive work include updating the schedule, verification of invoices, calculation of weekly quantities and entering weekly cost and quantity updates. Some examples of non-repetitive work includes determining field production measurements, determining schedule activity duration (this is now becoming more automated and less ad-hoc through the use of an integrated scope-cost-schedule model) or logic changes, unit-cost determination and determining work activity codes. The non-repetitive tasks are often determined by someone with knowledge and experience and only entered by engineers in a data-entry fashion. Once the data is formatted into an electronic state the remaining tasks are repetitive.

There are many programs available to write and record macros. Microsoft includes a macro recorder and editor with the Microsoft Office application[v]. Many other macro applications are available online for free or for a fee. One example is VsiSystems Macro wizard 4.1 which I used for this example[1].

[1] VsiSystems http://www.vsisystems.com/



Address http://tonto.eia.doe.gov/oog/info/wohdp/diesel_detail_report_combined.asp

Energy Information Administration

Official Energy Statistics from the U.S. Government

Glossary

Home > Petroleum > Weekly Retail On-Highway Diesel Prices > Average All Types - Last 53 Weeks

Weekly Retail On-Highway Diesel Prices

Average All Types - Last 53 Weeks

(Dollars per gallon, including all taxes)

Date	U.S. Average	East Coast	New England	Central Atlantic	Lower Atlantic	Midwest	Gulf Coast	Rocky Mtn	West Coast	CA
08/06/07	2.898	2.866	2.955	2.945	2.825	2.887	2.824	3.012	3.077	3.140
07/30/07	2.886	2.847	2.946	2.923	2.806	2.885	2.808	3.003	3.058	3.152
07/23/07	2.889	2.860	2.950	2.945	2.815	2.882	2.815	2.988	3.061	3.158
07/16/07	2.889	2.874	2.959	2.949	2.834	2.878	2.821	2.977	3.031	3.147
07/09/07	2.849	2.853	2.942	2.936	2.810	2.822	2.785	2.955	2.987	3.090
07/02/07	2.829	2.827	2.930	2.916	2.779	2.805	2.766	2.936	2.972	3.067

Figure 13: This government website contains average weekly fuel cost for regions within the U.S. Trucking companies use this website to determine surcharge rates to account for volatility in fuel prices. To verify invoices for hauling, it is necessary to calculate the surcharge first.

There are known limitations to macros particular to each version. Two of these limitations are Visual Basic for Applications (VBE) will not operate outside of Microsoft applications and VsiSystems does not interact directly with VBE. Additional limitations may exist but are likely dependent on the macro version being used, it could be speculated the more expensive macros have fewer limitations and are more popular while free macros have more limitations making them less popular.

Using Macros

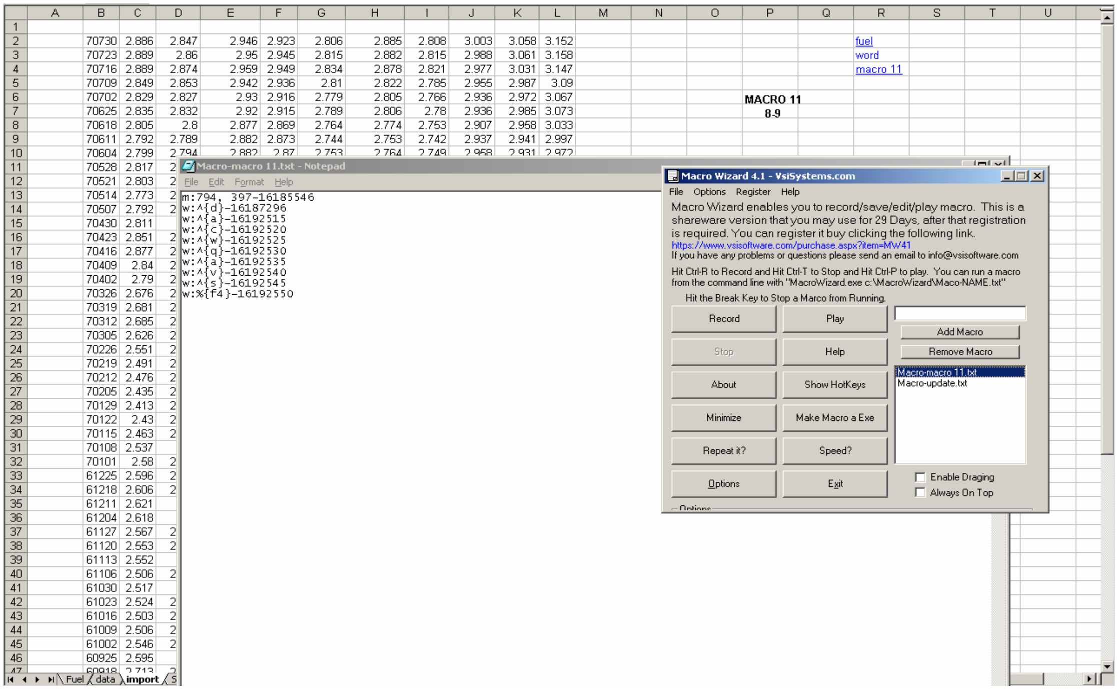
An actual field macro use identified is the need to look up fuel cost and then calculate a surcharge percentage in order to verify subcontractor haul invoices. This task must be completed every week prior to verifying invoices. On a large job with several dozens of haul trucks or more operating, the task of verifying invoices can become overwhelming. Due to fuel cost fluctuations a surcharge is assessed to the hourly rate based on average weekly fuel cost within a given region.

The weekly fuel cost is posted on a government website with cost broken down by region and fuel type. Each week it is necessary to retrieve the new fuel cost, calculate the weeks surcharge and update the invoice verification software. Verification software is simply a replication of the subcontractors invoicing software using a spreadsheet. Hours are entered with deductions for lunch, breakdowns and then additions for startup and driving time. The trailer type is entered since each trailer combination has a differing rate. The load weight is also entered since this can effect the rate, an under loaded truck is billed at a lower rate than a fully loaded truck. The importance of this is that under loading results in faster cycle time, potentially used to hide a stop for lunch and results in a faulty volume derived from load count[2].

The exogenous variables are hours, adds, deducts, trailer type, load tonnage, load count, sales tax and surcharge. From these exogenous variables all of the calculations can be made to

[2] Haul quantities can be derived from know trailer capacity, known material density in Loose Cubic Yards (LCY) and cycles.

determine endogenous variables of cost, quantity hauled, unit cost, and production rate. Cost is compared with the invoice submitted by the subcontractor; these should be equal to the penny, any difference requires a suitable explanation. Each month a list of invoices with discrepancies is sent to the subcontractors for verification. For a small haul of approximately 150 thousand cubic yards, I had a return of about $10K per month. Without careful screening this cost would be paid and is too small to notice. Though the task of checking invoices is tedious and any way to simplify the process encourages engineers to check each invoice rather than conduct semi-random spot checks. Of the exogenous variables (hours, adds, deducts, trailer type, load tonnage, load count, sales tax and surcharge) all are non-repeating except for surcharge and tax rate. The tax rate usually is very stable and only needs to be entered once. An online search algorithm written to automatically search the local tax rate based on project location would be useful for this, but is outside the scope of macros. The surcharge rate is the main variable that is exogenous and can be determined in a repetitive manner as well as being required at a regular subproject interval. The automation of the surcharge lookup results in reduced potential for error and eliminates one task from an engineers list of things to do each week.



	A	B	C	D	E	F	G	H	I	J	K	L
1												
2		70730	2.886	2.847	2.946	2.923	2.806	2.885	2.808	3.003	3.058	3.152
3		70723	2.889	2.86	2.95	2.945	2.815	2.882	2.815	2.988	3.061	3.158
4		70716	2.889	2.874	2.959	2.949	2.834	2.878	2.821	2.977	3.031	3.147
5		70709	2.849	2.853	2.942	2.936	2.81	2.822	2.785	2.955	2.987	3.09
6		70702	2.829	2.827	2.93	2.916	2.779	2.805	2.766	2.936	2.972	3.067
7		70625	2.835	2.832	2.92	2.915	2.789	2.806	2.78	2.936	2.985	3.073
8		70618	2.805	2.8	2.877	2.869	2.764	2.774	2.753	2.907	2.958	3.033
9		70611	2.792	2.789	2.882	2.873	2.744	2.753	2.742	2.937	2.941	2.997
10		70604	2.799	2.794	2.882	2.87	2.753	2.764	2.749	2.958	2.931	2.972
11		70528	2.817	2								
12		70521	2.803	2								
13		70514	2.773	2								
14		70507	2.792	2								
15		70430	2.811									
16		70423	2.851	2								
17		70416	2.877	2								
18		70409	2.84	2								
19		70402	2.79	2								
20		70326	2.676	2								
21		70319	2.681	2								
22		70312	2.685	2								
23		70305	2.626	2								
24		70226	2.551	2								
25		70219	2.491	2								
26		70212	2.476	2								
27		70205	2.435	2								
28		70129	2.413	2								
29		70122	2.43	2								
30		70115	2.463	2								
31		70108	2.537									
32		70101	2.58	2								
33		61225	2.596	2								
34		61218	2.606	2								
35		61211	2.621									
36		61204	2.618									
37		61127	2.567	2								
38		61120	2.553	2								
39		61113	2.552									
40		61106	2.506	2								
41		61030	2.517									
42		61023	2.524	2								
43		61016	2.503	2								
44		61009	2.506	2								
45		61002	2.546	2								
46		60925	2.595									
47		60918	2.713	2								

Macro-macro 11.txt - Notepad

```
m:794, 397-16185546
w:^{d}-16187296
w:^{a}-16192515
w:^{c}-16192520
w:^{w}-16192525
w:^{q}-16192530
w:^{a}-16192535
w:^{v}-16192540
w:^{s}-16192545
w:%{f4}-16192550
```

Macro Wizard 4.1 - VsiSystems.com

Macro Wizard enables you to record/save/edit/play macro. This is a shareware version that you may use for 29 Days, after that registration is required. You can register it buy clicking the following link.
https://www.vsisoftware.com/purchase.aspx?item=MW41
If you have any problems or questions please send an email to info@vsisoftware.com

Hit Ctrl-R to Record and Hit Ctrl-T to Stop and Hit Ctrl-P to play. You can run a macro from the command line with "MacroWizard.exe c:\MacroWizard\Maco-NAME.txt"

Hit the Break Key to Stop a Marco from Running.

Macro-macro 11.txt
Macro-update.txt

□ Enable Draging
□ Always On Top

Figure 14: The process of writing a macro is straightforward and should be understood by field engineers with no prior programming experience. There are two methods of creating the macro, either recording a series of keystrokes or writing code. The government website provides the option of downloading the fuel cost as a text file. With this text file macros are able to then manipulate the format to suite various purposes

Example macro

In order to create a macro to update the fuel cost data from a government server, the sequence of tasks needed to retrieve this data must be macro compatible. Normally to update the data the following sequence of steps is required, in pseudo-code:

```
Open internet explorer or similar browser application (mouse click)
```

→ enter the website address into the address bar (keyboard and mouse click)[3]

→ copy last weeks fuel cost row from page (mouse click)

→ open haul truck spreadsheet (mouse click)

→ insert new row then paste row of fuel cost data onto tab (mouse click)

→ manually calculate surcharge or write recipe-formula to calculate surcharge rate from lookup table (mouse click and keyboard, maybe hand calculator, this is not uncommon)

→ copy new surcharge rate to invoice verification tab or write lookup function in invoice tab to lookup surcharge based on date. (mouse click and keyboard)

→ save (mouse click).

Some of these steps work in macros but most do not.

To accomplish the same task manually or in macros requires two differing methods. Manually most of the steps are completed using the keyboard and mouse functions. In macros mouse operations do not perform well, reliably or at all. Another problem is the webpage format, copying one particular row is difficult to accomplish using macros without a *.csv or text file format.

To lookup fuel cost from the government website using macros a new sequence of steps must be developed, in pseudo-code these are:

First open the spreadsheet for haul truck (mouse click)

→ click update macro linked to following sequence (mouse click)

→ open the government website in the provided alternate text format (macro hyperlink)[4]

→ copy entire page contents (ctrl + A)

→ close webpage (alt+f4)

→ open text file (macro hyperlink)

→ overwrite old data and paste new data (ctrl + A, ctrl + V)

→ save file and close (ctrl + S, alt+f4)

→ import text file using excel import external data tool (macro in VBE)

→ delete unneeded header from text website (macro in VBE)

→ sort data according to date (macro in VBE)

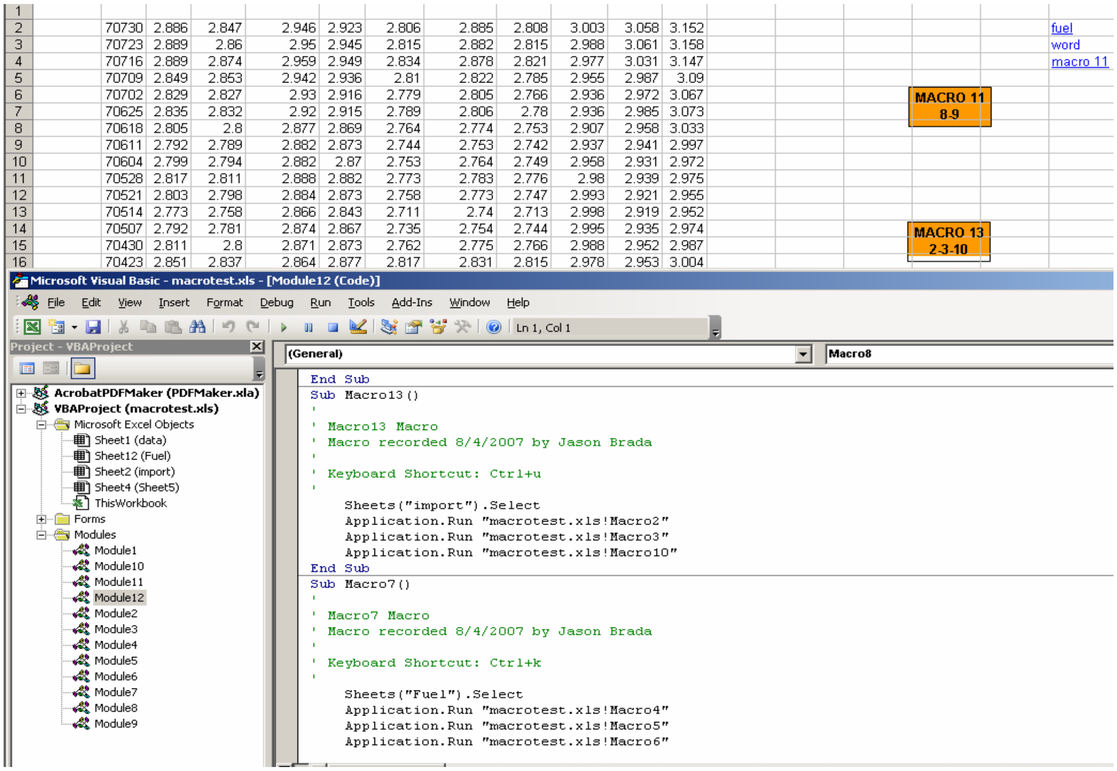
→ in order to maintain database of past data a macro to insert one row and add date one week from previous last date (macro in VBE)

→ utilize lookup functions and excel equations to calculate surcharge rate (endogenous calculations)

→ save (ctrl + S)

This macro works well and the only major hurdle was timing between various operations. The text file provided on the government server is a big help, otherwise the formatting and buttons on the webpage would need to be dealt with. Major issues are the date format provided is not compatible with excel and the header gets in the way. On the government server the date is in 070730 format or year/month/day, excel saw this as 08/24/2093, the solution is likely simple but if the text file format was consistent with excel it would not be an issue. The header provided

[3] http://tonto.eia.doe.gov/oog/info/wohdp/diesel_detail_report_combined.asp
[4] http://tonto.eia.doe.gov/oog/info/wohdp/List_Serve_report_All.txt



1													
2	70730	2.886	2.847	2.946	2.923	2.806	2.885	2.808	3.003	3.058	3.152		fuel
3	70723	2.889	2.86	2.95	2.945	2.815	2.882	2.815	2.988	3.061	3.158		word
4	70716	2.889	2.874	2.959	2.949	2.834	2.878	2.821	2.977	3.031	3.147		macro 11
5	70709	2.849	2.853	2.942	2.936	2.81	2.822	2.785	2.955	2.987	3.09		
6	70702	2.829	2.827	2.93	2.916	2.779	2.805	2.766	2.936	2.972	3.067		
7	70625	2.835	2.832	2.92	2.915	2.789	2.806	2.78	2.936	2.985	3.073		
8	70618	2.805	2.8	2.877	2.869	2.764	2.774	2.753	2.907	2.958	3.033		
9	70611	2.792	2.789	2.882	2.873	2.744	2.753	2.742	2.937	2.941	2.997		
10	70604	2.799	2.794	2.882	2.87	2.753	2.764	2.749	2.958	2.931	2.972		
11	70528	2.817	2.811	2.888	2.882	2.773	2.783	2.776	2.98	2.939	2.975		
12	70521	2.803	2.798	2.884	2.873	2.758	2.773	2.747	2.993	2.921	2.955		
13	70514	2.773	2.758	2.866	2.843	2.711	2.74	2.713	2.998	2.919	2.952		
14	70507	2.792	2.781	2.874	2.867	2.735	2.754	2.744	2.995	2.935	2.974		
15	70430	2.811	2.8	2.871	2.873	2.762	2.775	2.766	2.988	2.952	2.987		
16	70423	2.851	2.837	2.864	2.877	2.817	2.831	2.815	2.978	2.953	3.004		

```
End Sub
Sub Macro13()
'
' Macro13 Macro
' Macro recorded 8/4/2007 by Jason Brada
'
' Keyboard Shortcut: Ctrl+u
'
    Sheets("import").Select
    Application.Run "macrotest.xls!Macro2"
    Application.Run "macrotest.xls!Macro3"
    Application.Run "macrotest.xls!Macro10"
End Sub
Sub Macro7()
'
' Macro7 Macro
' Macro recorded 8/4/2007 by Jason Brada
'
' Keyboard Shortcut: Ctrl+k
'
    Sheets("Fuel").Select
    Application.Run "macrotest.xls!Macro4"
    Application.Run "macrotest.xls!Macro5"
    Application.Run "macrotest.xls!Macro6"
```

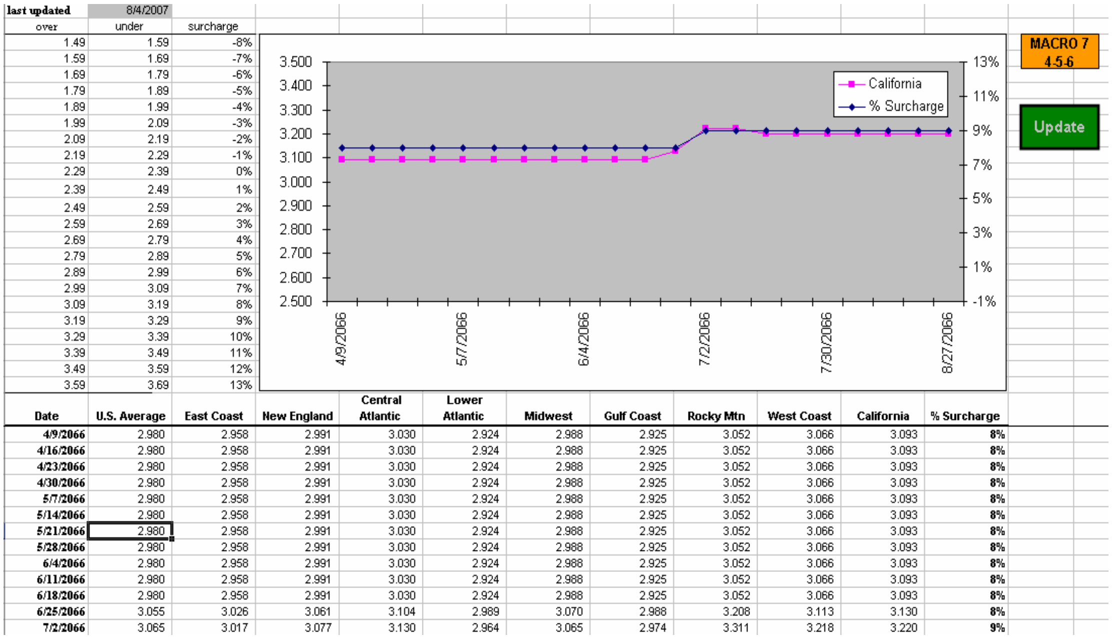
Figure 15: This view of Visual Basic (VBA) shows a series of short function macros that were then combined under a separate macro. With these simple macros, repetitive procedures or calculations can be completed automatically.

has no use in this application and is completely unneeded, only adding an extra step in removal for visual purposes.

Both macros used had limitations. The Microsoft VBE macros application worked well with Microsoft applications but was unusable in any other capacity. Online help pages and web blogs gave advice for adding time delays but these did not work properly. Manual editing of VBE code gave mixed results with the best being a fresh macro recording. VBE recording had good results with mouse operations. The VsiSystems macro application worked very well with all applications but with limited Microsoft macro compatibility other than through assigned shortcut keys. VsiSystems had a good timing recorder and allowed easy adjustment to macro speed and delays. Using the mouse appeared to not be an option with VsiSystems, any change in button location in subsequent runs resulted in erroneous macro function.

A further macro that would be needed is to prevent overwriting of previous data and archive data in stable form. A macro written to copy → paste special → values, of archived data to hard punched rather than equations. In this way all data over a predetermined age is converted from equations to text and less susceptible to corruption.

Macros are not an extremely difficult concept and are easy to create. The difficulty stems from the need to work around the limitations and exploit the strengths in these tools. Many actions taken by engineers are repetitive and introduce the opportunity for error. The use of macros in data collection from a hosted server has been introduced and demonstrated with successful results.



last updated	8/4/2007	
over	under	surcharge
1.49	1.59	-8%
1.59	1.69	-7%
1.69	1.79	-6%
1.79	1.89	-5%
1.89	1.99	-4%
1.99	2.09	-3%
2.09	2.19	-2%
2.19	2.29	-1%
2.29	2.39	0%
2.39	2.49	1%
2.49	2.59	2%
2.59	2.69	3%
2.69	2.79	4%
2.79	2.89	5%
2.89	2.99	6%
2.99	3.09	7%
3.09	3.19	8%
3.19	3.29	9%
3.29	3.39	10%
3.39	3.49	11%
3.49	3.59	12%
3.59	3.69	13%

Date	U.S. Average	East Coast	New England	Central Atlantic	Lower Atlantic	Midwest	Gulf Coast	Rocky Mtn	West Coast	California	% Surcharge
4/9/2066	2.980	2.958	2.991	3.030	2.924	2.988	2.925	3.052	3.066	3.093	8%
4/16/2066	2.980	2.958	2.991	3.030	2.924	2.988	2.925	3.052	3.066	3.093	8%
4/23/2066	2.980	2.958	2.991	3.030	2.924	2.988	2.925	3.052	3.066	3.093	8%
4/30/2066	2.980	2.958	2.991	3.030	2.924	2.988	2.925	3.052	3.066	3.093	8%
5/7/2066	2.980	2.958	2.991	3.030	2.924	2.988	2.925	3.052	3.066	3.093	8%
5/14/2066	2.980	2.958	2.991	3.030	2.924	2.988	2.925	3.052	3.066	3.093	8%
5/21/2066	2.980	2.958	2.991	3.030	2.924	2.988	2.925	3.052	3.066	3.093	8%
5/28/2066	2.980	2.958	2.991	3.030	2.924	2.988	2.925	3.052	3.066	3.093	8%
6/4/2066	2.980	2.958	2.991	3.030	2.924	2.988	2.925	3.052	3.066	3.093	8%
6/11/2066	2.980	2.958	2.991	3.030	2.924	2.988	2.925	3.052	3.066	3.093	8%
6/18/2066	2.980	2.958	2.991	3.030	2.924	2.988	2.925	3.052	3.066	3.093	8%
6/25/2066	3.055	3.026	3.061	3.104	2.989	3.070	2.988	3.208	3.113	3.130	8%
7/2/2066	3.065	3.017	3.077	3.130	2.964	3.065	2.974	3.311	3.218	3.220	9%

Figure 16: In this calculation average fuel cost is imported by region from a government website. Averages are calculated weekly. From this average a surcharge is calculated from an negotiated baseline set each year. In this case the baseline is $2.29 per gallon of diesel fuel resulting in an 8% surcharge for the given time periods shown.

Conclusion

In conclusion the use of a macro to complete a repetitive task is possible. The time needed to write the macro and debug is longer than the individual task. In the presented example it took some time[5] to find applicable macro applications, learn how to use the applications, write a sequence and debug. With this experience it could be possible to complete a similar task in a few hours. Once the macro has been written the repetitive steps no longer need to be taken, resulting in savings in time and reduced risk.

A couple of concerns could be if the macro becomes corrupt and must be rewritten or debugged, also the complexity of the final macros, or non-standardization in programming could result in confusion. The complexity of subsequent macros as the entire list of tasks completed by an engineer are reduced to only keying non-automatable exogenous tasks, could result in becoming out-of-touch with the inner working of automation resulting in a black box. This is the risk with any automation.

The macro worked well and could cover the entire range of field engineer work tasks as applicable. Training engineers to use macros where repeatable information is entered or repetitive tasks sequences exist would be helpful for future field engineers. Model-based quantities collection and control presents the opportunity for small macros to automate various tasks that for whatever reason are not included in the model-based integration. Even the task of updating the integrated tools after a change could be given to a macro. This flexibility provides the field engineer with a tool to modify applications in specific scenarios not envisioned by the software developers. Thank you to Tobias and Saurabh for preliminary explanations on how to use macros and text file importing.

[5] 20 hours

[i] Hartmann, T., Fischer, M., Rank, E., Schreyer,M., and Neuberg, F. "Integration of a Three Dimensional CAD Environment into an Interactive Workspace" Stanford University Center for Integrated Facility Engineering, CIFE Technical Report 146, , 2003

[ii] Chua, D.K.H., Jun S.L., Hwee, B.S. (1999). “Integrated Production Scheduler for Construction Look-ahead Planning,” IGLC-7, Berkeley, CA, 287-298, July, 26-28.

[iii] Cheng J. "A Simulation Access Language And Framework With Applications To Project Management" Stanford University 2006

[iv] "Macro (computer science)." Wikipedia, The Free Encyclopedia. 16 Jul 2008, 20:40 UTC. Wikimedia Foundation, Inc. 17 Jul 2008 <http://en.wikipedia.org/w/index.php?title=Macro_%28computer_science%29&oldid=226099611>

[v] Microsoft Excel Macros: http://office.microsoft.com/en-us/excel/HA010548371033.aspx