HACK, D.R. Utilization of scripting languages for customization of mine planning packages by specialized users. *Application of Computers and Operations Research in the Minerals Industries,* South African Institute of Mining and Metallurgy, 2003.

Utilization of scripting languages for customization of mine planning packages by specialized users

D.R. HACK
Halstead GeoNumerics, Inc., Portland, Oregon, USA

The complexity of commercial mining software has gained rapidly over the past decade as hardware developments have surpassed software capabilities at a staggering rate. Software users, recognizing the limitations of off-the-shelf applications, have adapted embedded scripting languages. Custom programming has always had a vital role in mine computing innovations, yielding a dynamic research environment in industry and academia. Fields such as industrial minerals can benefit from customization of existing software, since the majority of mine planning software was designed for large metal mines. Different customization approaches utilizing embedded scripting languages are presented, with examples of sand and gravel applications in development. Following implementation by Surpac and development of the Surpac Command Language, Tcl has been widely applied. Mintec introduced Python, also extensible and portable amongst various operating systems, for use in their MineSight products. In early 2000, Gemcom introduced the use of Microsoft Visual Basic for Applications, widely used customization tools for a decade. Datamine implemented a flexible multi-language approach, with all program commands and processes available through the COM Server Interface. This extends customization from the existing macro language to scripting in HTML, JavaScript, or any language that supports COM.

Introduction

Computer methods have been an integral component in the daily life of operating mines for more than a quarter century. Applications range from pre-packaged off-the-shelf programs to massive IT management systems customized by the vendor. The next logical step is end-user customization. Users from sectors for which the software was not originally designed, such as industrial minerals, will benefit particularly from customization, by adapting existing programs to their narrow requirements while utilizing built-in power and sophistication. Consequentially, developers have recognized scripting applications as an inroad to unexplored market sectors.

There is not yet a wide variety of industry specific solutions to describe, as computer usage for mine planning is fairly new for many sectors, such as sand and gravel. Solutions generally begin as small tools and proceed through several generations to become more sophisticated modules useful to others. The time to maturity of custom solutions from user to commercial marketplace and/or availability to other users can be considerable. Therefore, this paper will serve more as an overview of aspects relevant to impending development, and a few imminent applications will be discussed.

Brief history of scripting languages relevant to mining applications

Astounding increases in computer speed over the past few years have fostered a comparable increase in the power and sophistication of scripting languages. In addition, the emergence of web-based applications has popularized embedded scripting languages since scripts can be embedded in HTML documents. Scripting, however, has been around for at least forty years. A 'scripting language' is a programming language supported by a *host* program. A scripting program (or 'script') is useful for the automation of complex routines or advanced features within the host program. In addition, a scripting language can be used to integrate components and applications.

Job Control Language (JCL) is reported to be the earliest scripting language[1], used in the 1960s to sequence job steps to arrange the flow of data in card decks for OS/360. Unix shells, such as sh, csh, and ksh, became popular in the early 1970s, and remain so, for typing interactive commands and automating common tasks. The next milestone was the development of REXX in 1979 was the next milestone relevant to the advance of scripting applications in mining. REXX was used to simplify programming tasks on IBM's CMS timesharing systems[2], and remains influential as a macro language for arbitrary application programs. This concept, critical in the emergence of embedded scripting languages in mining, means that macro languages and interpreters do not need to be designed for each application, as long as the application supports the REXX programming interface. AutoLISP and Visual Basic for Applications (VBA) are early customization innovations that remain in wide use. AutoLISP, a scripting language embedded within AutoCAD and a direct descendant of XLISP, introduced many engineering software users to the concept of scripting. AutoCAD Release 1 was shipped in December 1982, and AutoLISP has been included with AutoCAD since version 2.18 (Release 6) in January 1986.

detrimental impurities. SGMatrix is currently a stand-alone program, however it is evolving into a Surpac-resident data management environment for aggregate operation exploration and production data. Using Tcom, the program accepts input as sieve analysis cumulative per cent passing results in Microsoft Excel format. It automates the calculation of the numerous granulometric indices and parameters in use, whether sedimentologic, engineering, or concrete industry in origin. The next step is the generation of a 'master correlation table'. Displayed in this 'matrix' are linear correlation coefficients (ρ= -1.0 to 1.0) of every selected index correlated with every other index (see Figure 1). Calculating numerous types of correlations is a popular practice amongst engineering geologists and soils engineers. If sample locations are available, all of the selected variables will be correlated with Northing and Easting. This functions as a first-order appraisal of spatial data characteristics, assisting in the preliminary identification of spatial structures.

'SGRecPak' is a Surpac-resident extension for the automation of a concise reclamation bond estimate based on site-specific factors developed during the mine and reclamation planning process, such as pre- and post-mining topographic maps, required reclamation topsoil thickness, locations and extent of temporary or permanent topsoil stockpiles, and haul road locations. The calculation of reclamation performance bonds involves detailed planning, calculations, and cost estimates, based on concepts presented in the reclamation plan. A variety of output options are available, such as estimation of total reclamation costs, estimation of costs for individual activities (such as re-vegetation, drainage/erosion control, or ripping), whether graphically or in report format.

Tcl/iGANTT

As described above, iGANTT's basic data storage structure lends itself to customization by third-party developers because the data structure is externally available. This could be particularly valuable to aggregate users, for example, because much emphasis is placed on integrated scheduling of topsoil and overburden movement during mining and reclamation.

Python/Minesight

Custom reporting options are valuable to industrial minerals users, especially when programs are originally designed for metal mines, as was cited prior with respect to gravel pits versus underground gold mines. As Mintec's quarry, and other specialized, products develop a greater market share, Python integration will likely follow the pattern of Tcl integration, given the genetic similarities.

COM Server Interface/Datamine

The Cemento Andino operation in Peru has recently implemented the use of several Datamine products for use in data management, short and long-term planning, and exploration[26,27]. All of these aspects require customization via scripting. Four primary scripts, based on existing templates, were written in Java Script and are as follows:

- *Data Update* (*Actualizacion*)—Enables the user to import and update blast hole information, as well as update the topography and the block model with new survey data.
- *Long-Term Planning* (*Largo Plazo*)—Links with the RM Scheduler (RMS) software program, prepares the bock model so it can be used by RMS, launches RMS, and then combines the existing information with the updates as obtained via RMS.
- *Short-Term Planning* (*Corto Plazo*)—Uses the block model generated with the Long-Term Planning script and generates a short-term calendar in consideration of quality and production targets. It also generates reports that can be checked using Microsoft Excel. See Figure 2 for a portion of the GUI.
- *Exploration Office*—Provides a link with the Exploration Office software.

The interface was developed to address the practical requirements of users, is written in Spanish, and is very intuitive. The scripts can be easily modified and improved at any time if needed.

26	Sample #: CN046–COMP															
27	Input data file: CN–024															
28	Sample #:CN024–COMP															
29																
30	CORRELATION MATRIX:															
31		Northing	Easting	D10	D25	D30	D50	D60	D75	#10	#30	#50	#100	#200	UC	GC
32	Northing		-0.745	0.2206	0.4659	0.3542	**0.627**	**0.67**	**0.752**	**-0.62**	**-0.67**	-0.029	0.2192	0.1356	0.3808	0.3238
33	Easting			-0.173	-0.312	-0.435	**-0.57**	**-0.6**	**-0.68**	**0.559**	**0.597**	0.0856	-0.19	-0.044	-0.272	-0.404
34	D10				0.1958	0.1375	-0.042	0.0325	0.1169	-0.176	**-0.58**	**-0.72**	**-0.6**	-0.622	**-0.54**	-0.242
35	D25					**0.735**	0.4676	0.4644	0.459	**-0.57**	-0.487	0.0841	0.1021	-0.244	-0.029	**0.635**
36	D30						**0.634**	**0.625**	**0.594**	**-0.67**	**-0.51**	0.0171	0.0553	-0.304	-0.021	**0.775**
37	D50							**0.911**	**0.873**	**-0.82**	**-0.6**	0.1136	0.2616	0.0963	0.4323	**0.559**
38	D60								**0.894**	**-0.81**	**-0.64**	0.0713	0.2345	0.0816	0.4222	**0.537**
39	D75									**-0.81**	**-0.68**	0.0089	0.1741	0.0979	0.4317	**0.514**
40	#10										**0.769**	0.1412	0.0104	0.1008	-0.226	**-0.51**
41	#30											**0.506**	0.2809	0.3816	0.0924	-0.203
42	#50												**0.827**	**0.705**	**0.651**	0.3949
43	#100													**0.676**	**0.684**	0.3819
44	#200														**0.847**	0.1176
45	UC															0.329
46	GC															
47	Sorting															
48	Skewness															
49	Kurtosis															
50	Dust Ratio															
51																

Figure 1. Excel output from SGMatrix

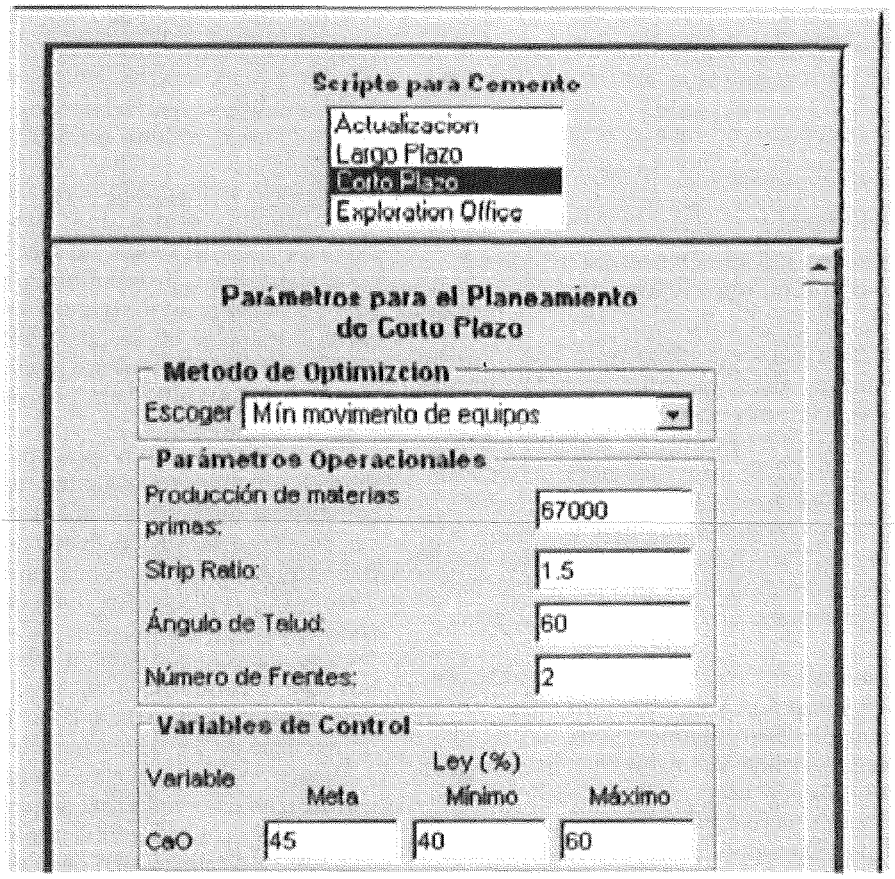

Figure 2. GUI from Datamine customization at Cemento Andino

Comparative discussion

In consideration of the spectrum of new or unexplored options available for specialized users (such as in the aggregate or cement industry) regarding scripting applications in mine planning packages, it is important to present and compare aspects relevant to their industries. This could assist new users in the final purchase selection, based on whichever mode of customization best suits their purposes.

Comparative benefits

Regarding the significance of a seamless interface with Windows applications, VBA would intrinsically be a first-tier selection. However, since open-source languages like Tcl are now capable of Windows integration, and developments in Python are imminent, VBA is no longer the sole option. VBA's potential dominance has been compromised by corporate control of language developments, as well as a tradition of tedious terminology and concepts. This dominance has been subsequently circumvented by Tcl and Python extensions. VBA does have unique benefits, such as supposed faster performance than standalone development tools. Another VBA benefit is that less end-user training may be necessary because applications look and feel like the Windows applications that the majority of computer users are already accustomed to. A notable strength of Gemcom's implementation of VBA is that system integration has been painstakingly constructed and extensively documented for developers[28]. All of the OLE Automation objects in Gemcom follow a consistent naming convention. This will assist Gemcom developers in the navigation of the dense layers of terminology. In spite of these benefits, broad active VBA development for Gemcom has not yet risen to the level of other applications, but implementation is relatively new.

Datamine's COM Server Interface approach '*...gives the programmer more flexibility in the choice of scripting language and development environment, in order to make best use of their skills and preferences...*'[22]. All Studio commands are available from an HTML page using any supported language, and other applications can be started from the same HTML page[24]. As with the Cemento Andino project, Datamine found that the advantages with the scripts as prepared for the client was that users were able complete tasks very quickly because of this automation, and the learning process was much faster than the software prior to customization. This impacts the subsequent efficiency and economic benefit of these data management and planning operations.

As was implied above, Tcl has a bright future because of the huge global development community. Furthermore, Surpac was a pioneer not only in the introduction of an embedded scripting language in a complex mine modelling and planning package, but also in the adaptation of this package's architecture to accommodate internal manipulation of functions in their products. The adaptation of Tcl by other companies like MineMax, as well as genetically related languages like Python by Mintec, has suggested the long-term popularity of Tcl. A potential disadvantage of Tcl implementation over a more centrally controlled language like VBA is that even though the development community is global in scale and very active, it also has the tendency to be quite fragmented. For

instance, if one has an interest in a particular function such as Excel portability, one can encounter literally dozens of extensions posted on various websites that are of widely varying quality. Unless, of course, one has a recommendation from another user, such as is the situation with the popularity of Tcom in the Surpac community.

Considerations in selection

Considerations in selection can be summarized as follows. The outcome of each decision clearly impacts that following in the list.

- Is a mine planning package already in use? For example, if a company already uses Gemcom Desktop Edition, then VBA is the only choice available.
- How soon must the system be available, and how much customization is needed? In many instances, solutions may already be available for a particular requirement. Conversely, extensive programming may be required for unique problems. The client should research what solutions, whether developed by the vendor or by independent programmers, are available and at what cost.
- Is it more economical for the vendor, or a qualified consultant, to develop and write the programs? Or should the client pay for company staff to attend training? Languages such as Tcl and Python have been traditionally easier for new users to adopt, where others such as VBA require more experience in concepts of object oriented programming. Stated otherwise, is the cost benefit derived only from operating efficiencies enhanced by customization, or further by the client's ability to take control of their industry specific computing requirements?
- How intensive are the scripting requirements? For example, are the requirements as limited as custom reporting (such as with Python use in Minesight), or as broad as access to internal functions and processes (such as with Surpac Vision)?

Future directions

Coincident with the emergence of web-based applications, mining software developers have taken seriously the importance of end-user customization via adoption of embedded scripting languages. The next necessary step is for developers to support an innovative user-base in a way that fosters robust experimentation, while keeping in mind corporate goals and policies. These two aspects can be difficult for corporations to balance in the anarchic tangle of the Internet. For example, as soon as program architecture, or open-source code, for a popular application is available for manipulation, the original developer no longer has control over the future of his creation. This is difficult for the corporation to appreciate when millions of dollars were spent on development. The innovative spark of the countless number of programmers online far surpasses the corporation's ability to control their creation. This is a reality in the information age that cannot be ignored.

From this freeware 'hacker' ethic, however, originate the brashest innovations, and developers can use this mystique to their advantage to create a rich development environment. Innovations such as Tcom, now an integral part of commercial applications like MineSched, began as a way to circumvent Microsoft application portability only by licensed partners. Such enthusiasm is critical to remain robust and innovative. Any scripting language that remains stifled by the controlling corporation is defeating its own purpose, which is to promote end-user experimentation. The most creative programmers, who have no patience for stumbling blocks and bureaucracy when cyberspace is full of free and easy options, will likely ignore such languages.

With respect to product customization for specialized sectors such as industrial minerals, utility of customization is a critical consideration. Specifically, the scripting language and GUI toolkit must be easy for the user to learn (assuming some programming experience), and tools, extensions, and tutorials must be freely available. Time spent in trainings or on the phone with tech support is not time brainstorming and programming, and if there is too much hassle involved the programmer may not take the time.

Because most custom solutions start small and progress through subsequent generations, the time to maturity to the commercial marketplace and/or availability to other users can be considerable. The fact that computer usage for mine planning in many sectors is still new is also a factor. Therefore, it will be some time before robust and valuable solutions for industrial minerals users are available in the greater marketplace though new applications are on the horizon. Vendors, however, are discovering that development of industry specific extensions is a good way to introduce their products into new market sectors.

Though the focus of prior examples has been aggregate and cement, other industrial minerals sectors have their own unique requirements that require customization. For example, geostatistics and mine planning for kaolin mining has unique considerations as pertains to the utilization of the important 'Brightness' parameter. Commodities such as garnet are characterized for sale in terms of their use for industrial purposes, with a subset of suitable quality for gems. Ornamental stones such as scoria are characterized by subjective colour parameters that cannot be measured in a laboratory, but whose occurrence in a quarry can be mapped by an experienced practitioner. Even with respect to aggregate, planning requirements for a railroad ballast quarry would be different than those for a concrete aggregate site.

The futures of individual scripting languages can be suggested, but not predicted, by their present success. Tcl has had the earliest success, and industry specific developments are already under way. Python's acceptance could be comparable to Tcl's, but usage is fairly new in mining circles. VBA's control by Microsoft has the potential to be a hindrance, but it could prevail as applications become more complex and performance efficiency becomes important. Datamine's COM Server Interface approach should become more popular, simply because it is the most flexible approach and is inclined toward web applications, which is the inevitable future of mine computing applications.

Acknowledgements

A number of people generously provided information and insight for this paper. Guy Sande (Couer d'Alene Mines Corporation) gave insight into the practical implementation of VBA in Gemcom Desktop Editions. Tim Carew (Gemcom USA) and Jennifer Palma (Gemcom Software/ Vancouver BC) provided information on VBA. Abdullah Arik (Mintec) provided a wealth of information on MineSight's implementation of Python. Rob Vallis (Hemlo Mines) described recent innovations of Tcl in Surpac. Jim Butler (MineMax) provided information on Tcl

implementation in iGANTT. Dave van de Ven (Digital Mining Services Australia) yielded a wealth of information on Tcl, Scl, GUIDO, and customization approaches. Special thanks to Andy Lapworth (Datamine UK) for valuable insight on the values of the COM Server Interface approach, and Albis Duran (Datamine Peru) for detail on specific applications in the cement industry.

References

1. Tcl Developer Exchange. History of Scripting. 2001. Retrieved on 12th March 2002 from the World Wide Web: http://www.tcl.tk/advocacy/scriptHistory.html
2. IBM UK Laboratories. The Rexx Language. 2002. Retrieved on 12th March 2002 from the World Wide Web: http://www 2.hursley.ibm.com/rexx/
3. LOMAX, P. VB & VBA in a Nutshell: The Language, Sebastopol Calif., O'Reilly & Associates. 1998. 633 pages. 1998.
4. GETZ, K., and GILBERT, M. VBA Developer's Handbook, San Francisco, Sybex. 1997. 922 pages.
5. Microsoft Developer's Network. Microsoft VBA Licensing Partners. 1999. Retrieved 4th January 2002 from the World Wide Web: http://msdn. microsoft.com/vba/companies/company.asp.
6. MineMax. iGANTT—A Robust Gantt-chart Based Tool for Production Scheduling. 2001. Retrieved on 12th December 2001 from the World Wide Web: http://www. minemax.com/prod_iGantt.html
7. Tcl Developer Xchange. History of Tcl. 2001. Retrieved on 17th March 2002 from the World Wide Web: http://www.tcl.tk/advocacy/tclHistory.html
8. Surpac Software International. Third Party Development Opportunities using the Surpac Command Language. 2000. Retrieved on 12th January 2001 from the World Wide Web: http://www.surpac.com/Business opportunity.htm
9. Digital Mining Services Australia. The Ore Is Out There. 2002. Retrieved on 12th February 2002 from the World Wide Web: http://www.z-files.com
10. MineSched International Ltd. MineSched Home Page. 2002. Retrieved on 14th April 2002 from the World Wide Web: http://www.minesched.com
11. HUANG, C.T. Tcom—Access and implement Windows COM objects with Tcl. No date. Retrieved on 10th November 2001 from the World Wide Web: http://www.vex.net/~cthuang/tcom/
12. BUTLER, J. Personal communication: Technical Director, MineMax Resource Optimization Software. 2002.
13. VAN ROSSUM, G. Foreword for 'Programming Python'. No date. Retrieved on 10th April 2002 from the World Wide Web: http://www.python.org/doc/essays/foreword.html
14. VAN ROSSUM, G. The Whole Python FAQ. No date. Retrieved on 10th November 2001 from the World Wide Web: http://www.python.org/cgi-bin/faqw.py?req=all.
15. Mintec, Inc. Mintec, Inc. Web Site. 2002. Retrieved on 14th May 2002 from the World Wide Web: http://www.mintec.com
16. Mintec. MineSight Planner promotional brochure, Tucson, Mintec, Inc. 2002. 5 pp.
17. Mintec. MineSight Operations promotional brochure, Tucson, Mintec, Inc. 2002. 7 pp.
18. ARIK, A. Personal communication: Senior Mining Consultant, Mintec, Inc. 2002.
19. Microsoft Developer's Network. Microsoft Visual Basic for Applications—ISV Overview. 2001. 2001 Retrieved on 17th May 2002 from the World Wide Web: http://msdn.microsoft.com/vba/prodinfo/backgrounder.asp
20. Gemcom Software International. Gemcom Software International, Inc. Integrated Resource Management Solutions. 2002. Retrieved on 10th February 2002 from the World Wide Web: http://www. gemcom.bc.ca
21. Gemcom Software International. Gemcom for Windows User Manual; Volume I: Core. Vancouver, Gemcom. 1998. 1502 pp.
22. LAPWORTH, A. Personal communication: General Manager, Datamine Software Ltd. 2002.
23. Earthworks Corporation. Earthworks Home Page. 2001. Retrieved on 17th March 2002 from the World Wide Web: http://www.earthworks.com.au
24. Datamine. Web Based Scripting Language. No date. Retrieved on 14th March 2002 from the World Wide Web: http://www.datamine-studio.com/brochure/scripting.htm
25. HACK, D.R. and VAN DE VEN, D.P. Customized Surpac Tcl extensions for the aggregate industry. In I. Gonzalez (Ed), CDROM Proceedings of the Fourth International Symposium of Computer Science in Mining (INFOMINA 2002). Lima, Instituto de Ingenieros de Minas del Peru
26. DURAND, A. Solucion integrada para el control de calida en canteras de calizas para cemento. In I. Gonzalez (Ed), CDROM Proceedings of the Fourth International Symposium of Computer Science in Mining (INFOMINA 2002). Lima, Instituto de Ingenieros de Minas del Peru.
27. DURAND, A. Personal communication: Manager of Technical Support, Datamine Peru. 2002.
28. Gemcom Software International. VBA Object Library Specifications. Vancouver, Gemcom. 2000. 96 pp.