M in Malta

by Chris Bonnici

What follows is a look at the M market in Malta. It attempts to demonstrate what the major contenders in M think about the language they use. In Malta, M is promoted by three software houses, Apex Ltd., Databyte Ltd., and Intercomp Ltd. All companies use MSM as their programming platform. For this article, I spoke to Chris Briffa (Intercomp Ltd.), Harry Restall (founder and former Managing Director of Intercomp), Joe Ross (Databyte Ltd.), and Adrian Borg (Apex Ltd.). Names are sorted in order of interview for this article.

All of the companies I contacted for this article use M to create programs which are sold to end users. One can find a few companies that have their own in-house development, but in almost all cases, the expertise of these companies was initially used to implement the first M system.

There are a few considerations that one should keep in mind when interpreting this article; Malta is a very small country, consisting of a small group of islands—Malta, Gozo, Comino, Cominotto, and Filfla, (only Malta and Gozo are inhabited, with Comino housing a score of people in all)—located in the Mediterranean Sea. The total land area is 316 sq. km. (122 sq. mi.). The population is about 400,000. Malta has a very active economy and has recently been promoted to a developed country status. Most companies reflect this small scale. A recent study showed that 75% of companies in Malta are classified as small according to EU standards. The largest companies (ignoring Government employees) employ a few thousand workers.

One would be incorrect to assume that because Malta is small, that software developed here is in any way a minute version of the software one would expect in any other region. While the quantity of data processed is small in relation to larger countries, the algorithms that do the actual processing are no different (in fact it is sometimes more complex). The hardware necessary to process and store data is normally the younger relative of that in other countries. Being small, a percentage of Maltese firms normally represent and deal with, or are subsidiaries of, foreign companies. This means that these entities must normally abide by certain procedures required by their foreign counterparts. All this must be seen in light of a rapidly changing economy with new laws and regulations coming out rapidly and with very little pre-announcement. A typical example is the implementation of VAT in Malta in January 1995. Different market sectors were still being briefed up through the third week of December 1994. As clearly seen, the average company must navigate a very dynamic route.

The Beginning

In August 1980, the computing scenario in Malta started to take off. Few large software companies existed (PC's had not made their debut), and government of the time was not really keen on computers (I feel that it was afraid of technology.). Harry Restall, the general manager of the newly formed company Intercomp Ltd., was looking for hardware and software to commence operations. In the U.K., he met with representatives of the British company CDS and purchased Data General equipment under an OEM contract from them. (Malta, being so small could not purchase the minimum quantities specified by Data General.). CDS was using MIIS on their systems. MIIS had a good interactive teaching package, and after going through it, Intercomp decided in favor of this language.

The price of a typical system was very competitive, and by the close of the year, Intercomp had two systems running in a financial institution specializing in property loans and in Malta's largest periodical/newspaper distribution company. A payroll system was developed soon after, followed by a fully integrated accounting system. Besides selling the software, Intercomp started offering data processing services for companies that could not afford their own hardware.

Intercomp moved to M in 1982 when they started using SWTP (South West Technical Products), though soon after, the company established solid contacts with Wang and Motorola and used this computer hardware on small and medium to large systems respectively. When Wang filed for chapter 11, Intercomp became a Dell distributor and started using these computers instead of Wang.

Joe Ross was at Intercomp for about 6 months to help introduce Wang in 1982. His company was formed in 1985 as a partnership and became Databyte Ltd. in 1986. Adrian Borg was a manager at Intercomp. In 1990 he left to form Apex Ltd. Both companies deal exclusively in software, yet today

both have well-established relations with hardware suppliers and can offer existing as well as new clients a complete solution.

The Ups and Downs of M

All those interviewed agree that M encourages one to be productive. In fact all companies claim that while they do offer off-the-shelf packages, they consider one of the major selling points to be the customization of existing software and tailor-made contracts. All claim that teaching or porting a programmer to M is relatively easy (although experience is an asset looked for).

Amongst the merits of M are claims of how efficient and stable the platform is. An example given was how M could withstand certain "shocks" other environments would crack under. "If the hard disk spins, we will normally try and will succeed in most of the attempts to fix it," was a quote I feel should be passed on unaltered. While all those I spoke to made it clear that they did recommend that their clients back up regularly, all told me that while others would probably close the conversation at "restore the last backup you have," these companies would attempt data recovery if the client requested it (for example, the last security copy was a week old or the failure took place towards the end of a business day).

The fact that M is portable is another benefit and the people I spoke to did mention how important this feature is when a client moves from one operating system to another. While in Malta the absolute majority of M systems run on UNIX, DOS, or Windows 95 platforms, if a company decides to shift, it can do so without the hassle other platforms may pose.

When I asked whether the cost of developing a system in M was less expensive compared to other suppliers (using different development tools) the answer was no. A system in M costs as much as, or even more than, a system written in a 4th GL. The points earned in increased productivity were normally spent on other aspects of a system.

Systems were classified into two categories: those with 8 or fewer users which we called small systems and those with more than 8 users, which for convenience rather than accuracy, we termed large systems.

In small systems, many competitors use tools in which the spin-offs are royalty-free. Even though an M run-time license is quite inexpensive, it still constitutes an added cost. Also many 4th GLs (even text-based ones) come with certain extras as standard that are completely lacking in M. A report generator was given as an example. Trying to convince a small company owner that these tools will, in all probability result in a wasteful consumption of paper, is very difficult, especially if in that same day, another two competitors have boasted about this feature. Besides, if the owner does want a simple columnar report with a few subtotals and everything totaled up, this has to be written and billed for in M. For small companies, the enormity of a full-blown report generator that necessitates any form of programming is not seen as necessary, for the reason that "these companies do not have the manpower and the time to delve into programming themselves—it would be a waste."

On large systems, M still suffers from what has been mentioned above, but due to a larger number of users, costs may be spread out more. The factor working against M is hardware. With the price difference between a PC and a dumb terminal so ridiculously low, the days when the complete hardware and software solution were cheaper than that of a competitor's are gone. Besides, many of these companies want to use networks for the various benefits entailed in this setup. The fact that M requires a lower computing platform than the most efficient competing product no longer carries any weight. If Microsoft Office can't perform adequately, the hardware won't be considered. Besides, even if a company was interested in low-end computers, where in the world (i.e., according to periodicals) would you find a 386 computer? For a few hundred dollars more than a 486, a company considering new hardware would opt for the mid-range Pentium-class computer. This computer is capable of running M as efficiently as other programming languages. Clients rarely consider the idea that maybe, in a few years' time, the latest version of their base software will no longer run efficiently while M will.

When I said the word "consultants," everyone agreed that the majority of these people were hurting M a lot. Those who opposed the implementation of software written in M were classified into two main groups, those who know that M exists (and might have used it a few years back) and those who don't know anything about it.

The first group know M as a 1960 language, period. They did not follow up on it and don't have an idea of what developments have taken place, and therefore, feel that M is a language stuck in the mud. Other programming tools have evolved during that time and logically (theirs) these products provide better tools and more features that benefit their clients.

The second group consists of those who don't even know there's a language called MUMPS or M. A talk with this group would reveal (after a little bit of explanation) that these people associate M with some early version of BASIC. They don't intend to demonstrate their lack of knowledge to their customer (bad publicity; it would ruin the "know-it-all" image), and therefore, are not willing to assume the responsibility of recommending this product. They will simply rule out the language, no reason given. If they build a certain confidence with the company quoting in M, a helpful piece

of advice is "when I'm consulting don't quote me in M; it's a waste of time." A language that runs efficiently on a 286 and does not require a network definitely has to be something not worth trying.

The smaller the user, the more they expect a GUI interface. Yet, now even large users seem to want some of their software written with a graphical interface. The fact that this is not yet possible means that people are putting off purchasing a system driven by M. The interface is that part of a program that probably most influences a client's decision on whether to go for a product or not. Just as a cake sells if the topping is nice, a lousy program with a nice interface has a better chance of being purchased than a masterpiece with poor packaging. Considering that most clients will barely have the time and resources to thoroughly go through a product for a duration of time sufficient to get below the icing, M companies simply can't compete. While the GUI version of M is still in its beta version, a commercial product is moving to its second or third release.

All the companies I talked to say that as far as their clients are concerned, they are buying software and the majority couldn't care less about it. The only exception to this rule is when a client is being recommended by someone else, but this could probably be ruled out as it acts both ways. The fact that M is an ANSI standard carries very little weight. Many don't know what ANSI means, others don't relate the term to any form of benefit. "Isn't Microsoft Office an ANSI standard?," is a question sometimes put forward. The reason for this probably lies in the misuse of the term "standard" and the use of phrases such as "market trends" (80% use so-and-so product) in general computing periodicals that are the main form of information for consultants and others who are considering computerization.

What Now?

To speak negatively is inherently human, yet progress (in any aspect of life) can be achieved only if one puts forward suggestions that lead to improvements.

The area of improvement most commonly suggested is that of advertising (more specifically a sustained and repeated advertising campaign). The advertising should be aimed at projecting M as it really is; a good, efficient, multi-user, multitasking, up-to-date, generic programming language, with solid roots, not bound to any one industry and not owned by any one company (that some day might fail and leave its clients stranded). Unless more people are aware that there is a language out there that is good, M's market potential will be limited. The line of thought is that advertisers get reviews, and reviews are read by consultants and potential clients who circle that topic for more information and may ultimately recommend or consider a system written in M. While the people I talked to realized that not all companies can afford hefty advertising budgets, one suggestion was to increase the role of the Mumps Development Committee (MDC) to include marketing. For example, develop an advertising campaign for M with the lower half of the list of advertising companies participating in it. The MDC would set up follow-up information for those who want to know more. The MDC could also assume the role of setting up reviews of products. The reasoning behind this is that if proprietary products sell mainly through advertising and the positive reviews associated with this advertising, why shouldn't M do the same?

I was also told that some publications have sections that list new products and will do so either for free or for a very nominal fee. M companies should actively look for such opportunities and should (in their own interest) make use of such forms of advertising.

The MDC was often mentioned as the body to tackle these group ventures mainly because it was appreciated that the high costs of promoting one's wares single-handed may be unattainable for individual companies and the MDC was already formed and already had a list of members.

A "salesperson/educator" was another suggestion. Supported by the MDC, this person's job would be to go around informing people about M. Whether the group happens to be business people or students, the job would entail demonstrating the potential of M.

A point of clarification is that no one has suggested that the MDC as it exists now stop doing what it already does; expanding the role of the MDC is seen as more sections (or departments) are added on board. I asked why the MDC and not MTA's or MUG's? The answer is that there is only one MDC and many of the others. Given a good communication channel such as what is possible with the Web, these groups could help the MDC in arriving at the best possible outcome, customized for the country or region.

Other comments involved attempting to improve the time from when a language enhancement is forwarded to when it becomes a standard. An example was quoted from an issue of *PC World* (the actual date could not be recalled). In the article, Microsoft chairman Bill Gates and the Microsoft Money team decide to make changes to Money and buy a bank all in less than an hour. The GUI version of M was mentioned most often. By the time the final release appears, it will still lack some of the objects present in other languages.

Another interesting suggestion, this time aimed at implementors and software houses is an "Access version of M." This version would be mainly interface-driven (as is the case with Microsoft Access) yet the core would be M. The package would be marketed and sold as with Access. Besides

increasing M market share, it could help increase awareness of M. A user with no knowledge of programming whatsoever could build a database and perform all the functions without having to type a single line of code.

Shareware/freeware code zones were also suggested as a means of increasing the popularity of M. Recently in newsgroups, the topic was suggested by the MTRC (M Technology Resource Center). A new user could access code snippets making his evaluation of M more interesting. A new user with no knowledge of M is expected to learn M by jotting down his first code himself. Such libraries (that should ideally vary from Astrology to Zoology in breadth of topics) would give a person more incentive to download M. Many people passively considering a language will normally first opt to download code to look at programs, modifying them slightly. Development is normally a second stage. Such a project may be extended to calling binary modules where ready-made modules can be called within program execution.

The idea of having languages that cater to and encourage third plug-in modules would also help to boost the popularity of M. The example quoted was (again) that of the GUI version of M. While the language implementors may want to stick closely to the ANSI-approved standard, they or third parties might make available to the market the non-approved modules that increase functionality. When an updated version of the language specification comes out, conversion programs that convert from the non-approved version to the approved version would get existing code in line. All this activity would surely generate new markets built around M.

The above paragraphs might give the impression of anarchy, and if one looks at other programming languages, the case does sometimes arise. The megabytes upon megabytes of code, some of it of low quality with only a small percentage being of envious quality, both give rise to popularity. Code junkies will spend hours downloading everything they find and telling others about their treasures. This environment constitutes one of the most fertile patches for the growth of a language.

The Future in Malta

The impression I got from those I spoke to was that "M people" have always taken up a defensive role. They seem to be happy with the one large fort they hold and passively defend only it. The problem with this approach is that outside attacks are constant and on the increase. Medical software not written in M already exists, and each new crack might ultimately end up bringing down the fortress leaving bits and pieces spread around the place.

M's future in Malta looks a bit gloomy mainly because Intercomp Ltd. seems to be on the road to moving out of M. Chris Briffa, Managing Director, told me that Intercomp is actively looking at other products and has already released their first non-M software (Delphi front end running on an Interbase engine). Considering the fact that Intercomp is one of the pioneers of computing in Malta, the company that brought M into Malta, and the company that has the largest base of M installed locally, clients can resist, but for how long? While the other two contenders might be able to cash in on the companies who want to stick to M, a number of clients will go the Intercomp way, reducing the overall installed base.

The other gentlemen I spoke to were more optimistic (though they are waiting for the official release of the MSM-Workstation for Windows) and at the time of writing showed no intent of moving out of M. From what they've seen of this product, they see that it has retained the true spirit of M; productivity and efficiency.

Prior to going to print, I discovered that the software house Megabyte Ltd. had won a tender for a hospital system written in M. The system will run on ISM. I spoke briefly to Megabyte's software manager, Mr. Julian Sant Fournier. Mr. Sant Fournier did confirm the hearsay but when asked whether Megabyte was considering moving to M, he answered in the negative. Megabyte is the local distributor for Oracle and is very active in promoting it (I've just received a letter from this company asking me to check from a list of about a dozen Oracle talks/demonstrations together with an offer for a free subscription to Oracle magazine). I am of the impression that the client had requested the software specifically and Megabyte was able to supply it.

Conclusion

Micronetics has announced that MSM-Workstation for Windows will be released this fall. I would like to thank all those who set aside time for the interview that made this article possible. A thank-you is also due to Frank Falzon and George Portelli who helped with editing and historical accuracy of this text. **M**

Chris Bonnici lives on the island of Malta and is the MIS manager for the Malta branch of Royal Insurance. He is involved with analysis and design and also plays an active role in programming. He has an advanced degree in computing and holds the post of Consultant for the Association of Insurance Companies. His email address is: Chribonn@keyworld.net.
