Two perceptions of software maintenance performed by an on-site contractor

by BERNARD NARROW
NASA Goddard Space Flight Center
Greenbelt, Maryland

and

JOHN KELLY
Lockheed, EMSCO
Greenbelt, Maryland

ABSTRACT

Software maintenance is a difficult task under the best of circumstances. Having work performed by an on-site contractor adds an additional layer of complexity to the customer's task. This type of relationship places greater emphasis on formal work procedures and detailed reports of the work in progress. It also promotes the use of performance norms for evaluating contractor performance. These factors are all on the positive side. However, such a relationship also calls for a special awareness of contractor ploys calculated to increase their performance evaluation.

From the contractor's point of view, being on-site imposes a more disciplined environment and places special importance on the manner and means of dealing with the customer. Another special feature is that the contractor receives formal feedback from the users, through periodic performance evaluations, indicating how well the software maintenance group measures up to expectations.

This paper describes the lessons learned by one customer and one on-site contractor.
CUSTOMER'S VIEW

contracting is likely to become more pervasive due to
basis in the federal government (a la the Office of
ment and Budget Circular A-76) as well as in state
ments and municipalities. An account is given here of
uge data processing facility with extensive experience
acting out software maintenance has learned to cope.
ata processing installation is a large, multisystem gov-
facility, comprised of a mix of manufacturers and
cluded are on-line systems, database systems, batch
and intercomputer systems. Types of hardware in-
M (370, 4341), UNIVAC (1100, Varian), SEL (32),
well (Sigma). Altogether, there are 20 stand-alone
that require software maintenance support. A large
of the applications run on these systems deal with
data; however the operational mode is akin to a
multistep production process. Other applications relate
action control, cost accounting, inventory control, and
e maintenance.

ZED vs. DECENTRALIZED SUPPORT

ally, our technical control over the work performed by
tractor was split along application and functional lines.
lo several independently run units, both on our side
the contractor's side. However, we exercised overall
stewardship over the contractor's activities for evalu-
performance. Arrangement, while providing us with a close working
ship and a strong grasp of the technical details, natu-
ally to parochial viewpoints on both sides. If a key sys-
son in one area resigned, contractor personnel could
cally call upon another area for temporary assistance
of reluctance by the latter to dilute their level of
Support problems, resulting from poor management,
enced or inadequate numbers of personnel, and the
ned to be prolonged and not pursued aggressively.
ly 1982, we reorganized to centralize all software
ance within a single unit. The contractor's organiza-
o was reconstituted on a centralized basis. A number
fits—some obvious and some not so obvious—were
, including

Uniform reporting of maintenance activities
Uniform and tighter configuration control
more effective communications channels
improved response in correcting or resolving problems
eparation and independence of programming and test-
groups

6. improved documentation (due to configuration control
oversight)
7. more effective control within the program library
8. more effective establishment of priorities and better
allocation of resources
9. more consistent manner evaluation of contractor per-
formance and determination of award fees
10. more availability of the information to build a central-
ized database for deriving work performance metrics

ESTABLISHING AN EFFECTIVE WORKING RELATIONSHIP

Because software maintenance cannot easily or readily be
translated into a set of well-defined products, the connection
between customer and contractor needs special emphasis. This
is a critical factor in determining the quality and cost
effectiveness of the support provided by the contractor. The
key elements characterizing the customer-contractor relation-
ship can be labeled as the three Cs: credibility, coverage, and
clout. Credibility hinges largely on the competence of cus-
tomer personnel. The level of competence should be such as

...
credibility in that it is meant to ensure that the technical discussions are substantive and are more likely to flush out causes, rather than treat symptoms.

Clout is a two-edged sword. It can be and should be used both to reward the contractor for better-than-expected performance and to penalize for below-expected performance. One way to accomplish this is by way of a cost-plus-award-fee contract, with the award ranging from 0% to 10% of the cost. Expected performance results in a fee in the 4% to 6% range, thereby leaving ample allowance for award level variations based either on positive or negative factors. Another important consideration is the level of management—both sides—that is involved in or is made aware of the fee determination.

On the customer side, this should mean the top person in charge of the data processing facility; on the contractor’s side, it should mean at least one level above the on-site manager, depending on whether software maintenance represents part or all of the contract. If the latter, the involvement should be at least two levels above the contract manager.

What You Need to Know

What do customers need to know about the contractor’s activities in order to monitor and evaluate the contractor’s performance effectively? In our case, we have stipulated that reporting should be at the functional (or third) level with system and project reporting being the higher levels.

For our purposes functional reporting was broken down as follows: validation and assessment of the effects of a proposed change (prior to approval by the configuration control board); programming (analysis, coding, and unit testing); acceptance testing; and implementation. This breakdown is predicated on the objective of closely monitoring the work in progress so as to be conversant with current problems and to assess effectively whether proper and timely actions are being taken to resolve them.

Some might argue that on a routine basis it is only necessary to monitor the contractor’s activities at either the system or project levels and thereby reduce the cost of monitoring. It is further argued that either periodic or unannounced audits can be made to determine the contractor’s performance at the functional level. The problem with this argument is, that substantive deficiencies are uncovered by an an n customer monitors are not in a position to assess idently whether the contractor is taking the proper or measures—and doing so in a timely manner. Waiting next audit takes place to make such a determination effective way to deal with such problems.

It should not be inferred that effective monitoring an an item-by-item review. One suggestion is to have an placed in the margin of a report to highlight those i which actual hours exceeded estimated hours or for data change was made since the last report period. Thi the monitor’s attention to the items that require clos vation and that should be accompanied by a written nation.

A complementary tactic is to specify the t important items in a separate report, which is distrib higher level of management than is the full detailed Specifying the detailed items to be reported on is the battle. The reports must be reviewed carefully f acy, completeness, and currency. Contractor person prone to adopt a casual attitude toward reporting if not held closely and consistently accountable for th contents. Figures 1 and 2 are sample formats of mont mary reports by system, showing, respectively, the cl status of all work in progress and the actual hours en by type of job.

Games Contractors Play

Wherever there are performance-type contracts, the inclination to “shade” the reporting of activities and e a way that is advantageous to the contractor. Althou ing can, in reality, be a euphemism for fraudulent it is more likely to manifest itself in more subtle odious forms. Also, on-site contractors are less like gage in these practices than off-site contractors, be the more personal relationship in the former case.

Noted here are both known and suspected tactics tractors have used. These tactics represent an over illustration drawn from a number of different contractor.
Creative bookkeeping

To prevent actual hours from exceeding estimated hours on a given job, time is charged to "miscellaneous." A variation of creative bookkeeping is where the contractor in the process of trying to correct an error takes a shortcut, e.g., bypassing testing, in order to stay on schedule. Should this in turn cause additional errors, these are reported as new errors and are disposed of expeditiously. This, of course, leads to "favorable" measured performance.

Technical obfuscation

When analysis or diagnosis of a persistent problem does not turn up anything definite, or when an embarrassing event occurs, the contractor might try to talk his way around it. Jargon and vague but technically imposing reasons might be offered to convince customer monitors that the problem is not due to any fault of the contractor.

All in the family

Here, contractors try to be particularly responsive to the customer monitor's pet projects. This is coupled with ego-boosting tactics, which together are an attempt to foster the impression that "we are all family" and we ought to be protective of the other party's interests. A variation of this game is to seek the company monitor's advice and suggestions about how to handle a given problem. This tends to compromise the company monitor's objectivity in assessing the contractor's performance.

End-around play

Should the customer monitors prove rather astute in dealing with the contractor's games, or if the customer monitors are frequently critical of the contractor's performance, a play can be made to a higher level of management. An attempt is made first to establish a close liaison with higher management and then to convince them that the monitors are biased and unreasonable.

Old standbys

Briefly noted here are the more familiar excuses and tactics used by software personnel.

1. overly generous padding of estimates to perform jobs
2. blame it on the vendor's documentation
3. blame it on the operating system
4. blame it on the hardware
5. blame it on the person no longer employed by the contractor

Performance Measurement

Under an incentive-type contract, it is necessary to face the issue of performance measurement squarely. First to be addressed is the formulation of which elements and factors are to be evaluated and measured. The candidate elements are those from which one can derive the desired factors. Examples of such factors include management, productivity, responsiveness, timeliness, communication, planning, and initiative. Factors such as management, communication, and planning are highly subjective in nature and are evaluated in an indirect or an event basis. Others, such as productivity, responsiveness, and timeliness, are adaptable to objective measurement, and these are the ones discussed herein.

Before qualifying a set of metrics for performance evaluation, it is necessary to define and establish a database. In our case, pertinent information is collected from all jobs including application software changes, operating systems maintenance, and special software tasks. Information about these jobs is collected from the individual programmers, and entered into a database. Weekly reports compiled from this information are carefully reviewed both by the contractor supervisors and the monitors to assure complete reporting and overall accountability. A list of the metrics that we observe is shown in Table I.

Each of the metrics in the table can be further categorized by computer system, language, type (i.e., systems or application software), and so forth. Such breakdowns enable comparisons to be made within the given category; e.g., how does the average time per fix for system A compare with that for system B?

After a sufficient amount of time has elapsed to compile a substantial database and to analyze and interpret the derived metrics, the final step can begin. This is to establish the norms for each of the selected metrics. Here again, contractor personnel should participate in this determination in order to arrive at a set of norms that is deemed to be fair and reasonable to both parties.

Such objective performance measures can be weighed and coupled with the subjective factors referred to earlier so as to arrive at the contractor's overall technical performance assessment.

THE CONTRACTOR'S VIEW

Interfacing with Customer Personnel

The role of the software professional within a company that performs facility management services is somewhat different from that of a programmer nestled comfortably in a corporate structured arena. Being on-site readily exposes a casual or sloppily managed working group and calls for an awareness or presence that should be calculated to command the respect of the customer. Sloppy personal demeanor, unoccupied desks, persons reading newspapers, and so on, are perceived by the customer as indicators that the contractor is unreliable, unprofessional, or underworked. In effect, the contractor has two "bosses"—the on-site customer as well as company management. This presents a unique dilemma—how to please both factions and maintain proper professional perspective (and sanity) in successfully fulfilling job requirements.

Acquiring the confidence of customer-monitoring personnel is an important goal that must be achieved quickly if successful performance ratings are to be attained. The ability to grasp the technical jargon and the complexities of the customer's subject matter makes customer communication a natural extension of the monitor's working environment.

When special requirements are addressed, the contractor should obtain customer concurrence on how the workload should be adjusted to satisfy all affected users. Too often, additional task requirements are accepted by the contractor without informing the customer of current manpower con-
strains and the effects of new tasks on current completion schedules. The contractor must not be afraid to oppose additional customer requests and should be prepared to convey to customer management that in reality there is no free lunch. When possible, suitable alternatives should be recommended.

Effective communication of task performance appraisals is an area that requires special contractor attention. The customer needs appropriate status information to provide a sufficient base for pointing out shortcomings, giving plaudits for tasks well done, and recommending an appropriate award fee. Formats for contractually required reports should be determined mutually, at the beginning of the contract, and should be reviewed periodically for possible alteration to respond to changing customer management reporting requirements. In addition to these reports, regularly scheduled status meetings between software management and key customer-management-technical-monitoring personnel should be established. These meetings, which are by design less formal and in the nature of committee sessions, are multipurpose. They not only provide a forum for presenting firsthand status information, but also are an excellent opportunity for discussing customer priorities and perceived deficiencies prior to their being written into the customer’s evaluation report. Another helpful measure is to provide a self-evaluation—representing the software management’s view of task performance—to the customer for consideration in determining periodic award fees.

*How Work is Divided and Allocated*

As noted in the first part of this paper, we are a centralized software organization, responsible for maintaining more than 50 software systems functioning on more than 20 mainframes, and for all developmental work. Major functions are separated into applications programming, systems programming, and software acceptance testing. By definition, applications programmers are responsible for maintaining the production software (primarily FORTRAN coding, with some assembly language) and the systems programmers are the caretakers of all operating system software. Systems analysts, however, provide the necessary expertise for assuring the validity of both new and modified software through the development and execution of detailed acceptance test plans.

Because of the size of this organization—approximately 85 software professionals—the numerous specially developed computer systems, and the frequency of software changes attributable to data-related and user requirement variances, it is difficult to impose conventional software management techniques. An internal task-tracking system has been developed to monitor several hundred tasks ranging from discrepancy reports (something doesn’t look right) to change requests (modifications to accommodate specific problems or requirements). Included within this range are customer-initiated tasks (often new requirements) and tasks generated internally by software management (usually related to normal maintenance activities, such as evaluating release tapes for existing operating systems). Due to the high volume of tasks, complexities of interorganizational interface, and management requirements for up-to-date status reporting, a full-time administrator is employed to maintain and coordinate all transactions and report generation attributed to this tracking system.

Assignment of programmers to support each system can often be a difficult process. Software management must be prepared to evaluate the overall complexity of the system, be familiar with the intricacies of various program components, and be knowledgeable about the stability or volatility of the software. These variables are then matched against individual programmer experience profiles to determine the most appropriate manpower allocation.

*Acquiring and Retaining Technical Personnel*

Our typical maintenance programmer has almost five years of college training and more than six years of technical experience. Turnover, however, is surprisingly low in our case, because of an unusual phenomenon known as incumbency. Many of our software professionals have selected this area because of the nature of the work—it is highly scientific and very interesting; the physical plant is conveniently located and easily accessible; there is no charge for parking; etc. Even though the contract is bound by a prenegotiated amount of time, the technically oriented employee has little fear of losing a position due to contract expiration. Obviously, even under a new contractor, the job must continue to be performed. Who else, other than those currently doing the job, could satisfy customer requirements with no untoward effect on daily operations? Of course, if there is a new contract awarded, management must be sensitive to the apprehension programmers are likely to exhibit during the recompetition and, if necessary, the changeover periods.

Programmers, like many other skilled professionals, consider themselves creative and take special pride in developing “eternal” systems. There exists, then, an innate stigma attached to the label of “maintenance programmer.” This is a difficult but not insurmountable hurdle for software management to overcome. One of the ways to maintain good personnel morale is by offering diversification in mainframes, operating systems, and programming languages. For example, in our case the opportunity to use FORTRAN, assembly language, or PL/I may be found on IBM (370/145 and 4341) using VS1, VM, or MVS; IBM (Series-I) using EDX; UNIVAC (1100/82) using EXEC-8 38R2; SEL (32/77 and 32/75) using RTM and MPX-32; VARIAN 77 using VORTEX; SIGMA 59 using BPM and CP-V; and various other special purpose image-processing systems.

Although the term maintenance is used to describe the main functions, many tasks require such extensive systems analysis prior to making appropriate changes that the programmer receives as much challenge and satisfaction as if the program was actually being developed. Another factor is training. In order to keep the staff abreast with state-of-the-art developments, management encourages formal vendor-supplied training classes. Specific analytical and systems-oriented techniques and skills are addressed in these courses. Attendance at user and general conferences is also an added
incentive provided to the programming professional for acquiring and dispensing information.

Dealing with Newly Developed Software

Almost all software maintenance groups encounter the problem of assuming responsibility for new software developed by another organization. In our case, this problem is compounded by the fact that the new programs are developed by another contractor. To deal effectively with this situation requires getting involved well before the software is delivered. Plans and interface definitions should be mutually agreed upon and include acceptance testing, documentation, and formal sessions for acquainting the maintenance personnel with the inner workings of each program.

The development of the acceptance test plan requires extensive communication between the maintenance and the development groups. Program design walk-throughs are highly recommended for this purpose, as well as for familiarizing the maintenance personnel with the software. This should be done prior to the turn-over of the program since afterwards development personnel are reassigned to other tasks and often are not easily accessible.

As on-site contractors, we need to be particularly concerned with the way information concerning our dealings with development personnel is presented to the customer. Group interaction problems, such as competing for computer time, should if possible be transparent to the customer. When these problems need to be brought to the customer’s attention, it is best to avoid a finger-pointing session. Such sensitivity and awareness contribute measurably to harmonious relations with the customer.

CONCLUSION

Overall, the use of on-site contractors can be a viable and effective means for accomplishing software maintenance in a large data processing facility. To achieve these ends, however, calls for a proper appreciation by both the customer monitors and the contractor management personnel of the factors and considerations described herein.

REFERENCE

1. Office of Management and Budget Circular A-76. This circular has been incorporated into the Federal Acquisition Regulation as Subpart 7.3, effective April 1, 1984.