SOFTWARE QUALITY CONTROL

C. EDWARD McVANEY
J. D. EDWARDS & COMPANY

The Predicament

Computers have brought space age technology to the ordinary businessman. Their potential seems boundless; the electronic equipment (hardware) is usually superlative; but the computer programs (software) can bring a plague to your organization.

The hardware side of the computer industry seems to be far more advanced than its software counterpart. The gripes and groans about computer failures are not all without substance, and ninety-nine times out of a hundred it seems it is the software that is at fault.

You have to have software! The computer won't work without it! But what do you do? How do you protect yourself? How do you assure a high quality result?

Knowing what to look for, and how to judge quality software, need not be a great mystery. You don't have to be a computer technician to make meaningful value judgments. Your common sense will serve you well if you're willing to do some research.

Before we go much further, we should make a clear distinction between "systems" and "applications" software. Applications software is the term computer specialists use to identify major business systems on a computer. For example:

- Payroll Systems
- Inventory Systems
- Billing Systems, and
- Accounting Systems

Applications software is sometimes referred to as a software system, a software package or just a system. In addition to the individual computer programs (usually ranging from 10 to 100 programs), it would include all the supporting documentation and related professional services including training and on-going maintenance.

Systems software, on the other hand, is a term used to describe the family of computer programs that are used by the computer to do its own work. For example:

- Operating System
- Compilers
- Utility Programs, and
- Communication Monitors

2. Systems Documentation - Documentation is to a computer system what a shop manual is to a mechanic - it is not very important until you need it. Systems documentation should consist of:

- Samples of Each Report
- Illustrations of each Video Display
- Copies of Forms and Documents
- A Glossary of Terms and Codes
- Clerical Procedures
- Computer Operating Instructions
- Program Descriptions
- Data Base Descriptions, and
- Other Appropriate Material

If these items are not readily available and professionally packaged so that they are easy to understand, you've noted a serious weakness in the system. You will be vulnerable to turnover of personnel and empire building if you don't have good documentation.

3. Clerical Procedures - Computer specialists have been known to forget about the people who have to use the system. Check the clerical operating procedures including the video terminal operating instructions to see if they are simple and effective. There's no need to explain the mundane, but at the very least, the unusual aspects of the system should be explained.

Of course, checking for voids goes beyond the particulars we've noted here - video displays, operating reports, input journals, forms, data bank, accounting controls and so on.

(b) Fitness to Your Needs

In their eagerness to sell computer hardware, salesmen have been known to recommend a software system that didn't quite fit. Major misfits are not uncommon.

If you are told that your organization can be changed or adapted to fit the software, make sure you know what will have to be changed. A good software package should provide definable improvements to the flow of work within your business with a minimum of organizational or procedural changes. It had better, because modification costs dearly, not only in terms of time and money but, more importantly, because vital management data often will not be available while the modification is being done. The "data gap" can be filled, of course, by keeping a manual or semi-automated system running parallel with the computer, but we can't imagine any business owner wanting to do that for very long.

(c) Maintenance Costs

Good software packages, like good automobiles, don't require much maintenance. If a computer programmer has to devote a significant part of his time to maintaining a software system, you can bet it is poorly designed or poorly programmed. Conversely, if little or no maintenance is required, you should be encouraged.

The maintenance cost of an acquired software package can often be determined by consulting with other users whose names the vendor has given you as references.

(d) Feature Comparisons

Another effective technique for judging software quality is feature comparisons between comparable systems. This is basic consumerism. These features include:

- Processing Capacity
- Hardware Requirements
- Management Accounting Techniques
- Data Base Concepts
- Timeliness of Reports
- Video Screen Features
- Audit Controls, and
- Others, as appropriate

Many of the little "extras" in a software system are well worth having.

(e) Operating Costs

If a software package does not reduce operating costs (clerical salaries, etc.) or provide significant intangible benefits such as improved management information or better customer service, then something is wrong.

Innumberable software systems have been installed with the anticipation of reduced operating cost and significant intangible benefits only to have just the opposite come true.

Therefore, you should check the "track record" or references of a software system to see if it has performed as promised. While you're checking references, ask about programmer maintenance costs.

(f) Real Response

Video terminal computer systems should provide for almost instantaneous entry of data, updating of records and redisplay of updated records. Many software packages fall far short of this capability. It is not uncommon to find video terminal packages that do nothing more than conventional "keypunching" into the video terminals and then process the data in the off hours. If this is the case, computer records might not be updated until the next morning even though video terminals were used for data entry.

(g) Technical Support

Behind every good computer system there should be good people to provide the technical support that, from time to time, you may need to rely on.

Do these people know and understand your business? Are they competent enough to get things done? What is their track record? Where will they be a year from now when you need them? Is this a one man show or a professional team of computer specialists? You should answer these questions and assure yourself of proper support.

Technical support personnel are all important to the success of your system.

(h) Conventional Technology

More often than you might expect, computer systems are developed using non-standard technology, i.e. offbeat programming languages, homemade systems software, and mix 'n' match equipment. Beware of these approaches because they invariably create inflexibility, overdependence on support personnel, and early obsolesence.

(i) Data Controls

Precise control features are essential to the success of any software package. Check to see that edit controls are installed to check the accuracy, reasonableness and interrelationship of all input transactions. Also check to see that input balancing controls are in effect. The system should have provision for a complete audit trail of all sensitive transactions. This audit trail should balance to master file control totals and reference back to the originating entries.

(j) Predefined Standards

A very effective way to evaluate the quality of a software system is to compare it to independent standards of excellence.

J. D. Edwards & Company has developed such standards, establishing criteria for, among other items:

- Systems Documentation
- Input Controls
- Accounting Controls
- Data Base Design
- Computer Processing
- Report Design
- Programming Structure
- Programming Efficiency, and
- Numerous other items.

Some Classic Examples

We have said that you don't have to be a computer expert to judge the quality of a software package. Let's illustrate how you may judge software quality with simple examples of a computer report and an input document.

(a) Report Quality

Illustration Number 1 represents a construction loan status report for Mr. Charles Louis who is building a new home. This report is a classic example of computerized goobledygook. Note the following deficiencies which could be readily detected by a non-technician.

1. The name and address were typed rather than printed by the computer.

2. The following columnar headings are unintelligible:

   - BR
   - WM
   - TEL
   - TY
   - CD
   - TY
   - MB
   - OV
   - TY DB

3. Note that there are three columns labeled TY.

4. Even if you understood what the codes mean, how would you understand the data? What, for example, does "I" mean in the MB column?

Illustration No. 1

Mr. Charles Louis
1822 N. Woodbriar
Littleton, Colorado 80120

6880 GREEN OAKS DR
LITTLETON COLORADO

FEDERAL SAVINGS AND LOAN ASSOCIATION PAGE 1

ACCOUNT NO. 01-A177000 LOAN IN PROCESS MONTHLY ACTIVITY REPORT DATE= 2/29/76

ORIG AMT 100,000.00 ORIG DATE 2-20-76 TERM OF LOAN 540 INTEREST YTD .00

ACTIVITY DATE	SOURCE HH MM	TEL	TRANS CD	TY M B	O V	TRANSACTION AMOUNT	LIP BALANCE AFTER	TY DR	CHECK PAID TO NUMBER		INTEREST PAID TO	PERCENT
2-20-76	13 02	708	6	LP=	=NA=JF=I=	100,000.00	100,000.00			LOAN SET UP	2-01-76	9.000%
2-20-76	13 02	708	6	LP=	=00=JF=0=	1,091.09=	98,907.00			CLOSING COSTS	2-01-76	9.000%
2-20-76	13 01	168	6	LP=	=00=LC=I=	7,000.00	105,907.00				2-01-76	9.000%
2-20-76	13 02	708	6	LP=	=00=LC=0=	510.19=	105,587.81		282952	COLUMBINE TITLE	2-01-76	9.000%
2-21-76	13 02	708	6	LP=	=00=LC=0=	2.00=	105,585.81		51341	H ROGERS	2-01-76	9.000%

SUMMARY: BEGINNING BAL= .00 RECEIPTS= 107,000.00 DISBURSEMENTS= 1,414.19= ENDING BAL= 105,585.81

Mr. Charles Louis was somewhat perplexed. He felt insecure because he didn't understand what the report said. So he took the report to his CPA and asked him to explain it - he was equally baffled.

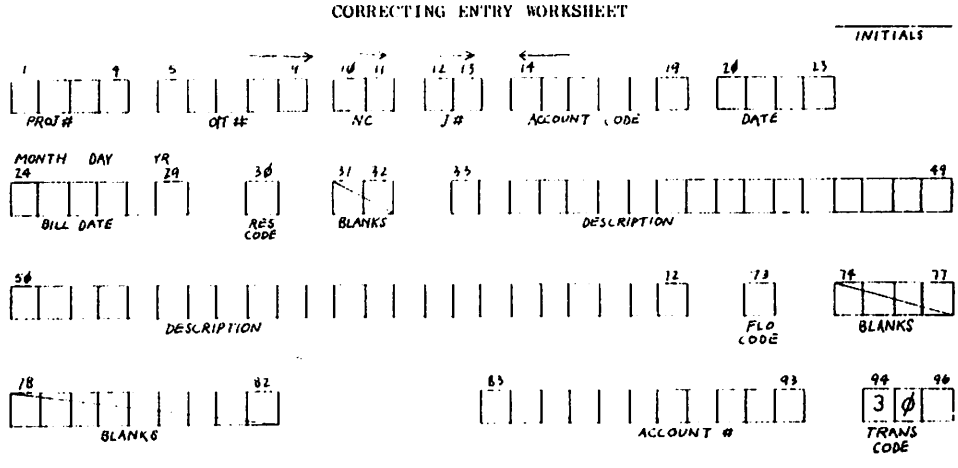
You should not tolerate this kind of poor report design in any software package. In almost all cases, computer reports can be made clear, concise and easy to interpret.

(b) Input Forms Quality

Illustration Number 2 represents a form used to correct a worksheet in a real estate management system. (This form has also reached the status of classic gobbledygook). Note the deficiencies which can be readily detected by a non-technician:

1. What do the arrows mean?
2. What are:
   - OIT#
   - NC
   - J#
   - RES Code
   - Fld Code?
3. What do you enter in the four digit date field?

Illustration No. 2



CC93 - -(dash) over digit for negative

The point is that neither computer input nor output forms need be as complex as many computer people make them. Look at Illustration Number 3, for example, the "Account Master Revisions" form reproduced on the next page. All column headings are in plain English - not computerese. Codes are plainly defined at the bottom of the sheet. There is plenty of room to write, and there are no mysterious arrows or slash marks.

Illustration No. 3

ACCOUNT MASTER REVISIONS

ACTION CODE	ACCOUNT NUMBER	REPORTING CODES: FUND	COST CENTER	OBJECT	SUBSIDIARY	GRANT	PROGRAM: MAJOR	PROGRAM: MINOR	FUNCTION	DEPT	FOOT NOTE	TYPE ACCT CODE	ACCOUNT DESCRIPTION

ACTION CODES

- A or 1 ADD NEW ACCOUNT
- C or 2 CHANGE OLD ACCOUNT
- D or 3 DELETE OLD ACCOUNT
- I or 4 INQUIRY

TYPE ACCOUNT CODES

- 0 ORGANIZATION NAME
- 1 FUND NAME
- 5 COST CENTER NAME
- 6 MAJOR ACCOUNTS
- 7 MINOR ACCOUNTS WITH SUBSIDIARIES
- 8 MINOR ACCOUNTS WITHOUT SUBSIDIARIES
- 9 SUBSIDIARY ACCOUNTS

DATE _____

PREPARED BY _____

Illustration No. 4

LABOR COST ANALYSIS

COST ACCT	ACCOUNT DESCRIPTION	BUDGET QUANT	UN	BUDGET UNIT $	BUDGET AMOUNT		ACTUAL QUANT	ACTUAL UNIT $	ACTUAL AMOUNT	% CMP	PROJECTED FINAL AMOUNT	GAIN-LOSS
00100	KNOCK OUT PANELS		LS		0	TO DATE			55	100.0	55	55
						THIS WK			0			
02001	FINE GRADE WALLS-FTGS-BMS	2,332	SF	.20	466	TO DATE	2,332	.276	644	100.0	644	178-
						THIS WK	0		0			
02002	FINE GRADE SLAB ON GRADE	16,141	SF	.11	2,098	TO DATE	16,141	.100	1,612	100.0	1,612	486
						THIS WK	0		0			
02003	FINE GRADE SIDEWALKS	4,390	SF	.13	571	TO DATE	702		0	16.0	0	571
						THIS WK	0		0			
02004	HAND EXCAVATION-ALL AREAS	1	LS		1,109	TO DATE			606	100.0	606	503
						THIS WK			0			
02005	BACKFILL - ALL AREAS	1,100	CY	2.00	2,200	TO DATE	990	4.254	4,211	90.0	4,679	2,479-
						THIS WK	0		0			
02006	GRAVEL UNDER S. O. G.	202	CY	2.50	505	TO DATE	202	6.039	1,382	100.0	1,382	877-
						THIS WK	0		0			
03001	CAISSONS	360	CY	4.00	1,472	TO DATE	368	.789	290	100.0	290	1,182
						THIS WK	0		0			
03002	GRADE BEAMS & WALLS	101	CY	2.00	362	TO DATE	101	3.101	561	100.0	561	199-
						THIS WK	0		0			
03003	CAISSON CAPS	2	CY	4.00	8	TO DATE	2		0	100.0	0	8
						THIS WK	0		0			
03004	COLUMNS 1ST & 2ND FLOOR	53	CY	7.00	371	TO DATE	53	12.714	674	100.0	674	303-
						THIS WK	0		0			
03005	COLUMNS 3RD & 4TH FLOOR	28	CY	9.14	256	TO DATE	28	1.491	42	100.0	42	214
						THIS WK	0		0			
03006	SLAB ON GRADE	206	CY	2.00	412	TO DATE	206	3.571	736	100.0	736	324-
						THIS WK	0		0			
03007	SLAB ON CORRUFORM	63	CY	2.25	142	TO DATE	63	3.892	245	100.0	245	103-
						THIS WK	0		0			
03008	PAN SLAB 1ST & 2ND FLOORS	614	CY	1.75	1,075	TO DATE	614	3.170	1,946	100.0	1,946	871-
						THIS WK	0		0			
03009	PAN SLAB 3RD & 4TH FLOORS	337	CY	2.04	690	TO DATE	337	3.327	1,121	100.0	1,121	431-
						THIS WK	0		0			
03010	TOPPING - ALL FLOORS	174	CY	2.00	348	TO DATE	87	6.314	549	50.0	1,099	751-
						THIS WK	0		0			
03011	ELEVATOR WALLS	120	CY	2.25	280	TO DATE	128	3.400	435	100.0	435	147-
						THIS WK	0		0			

The "Labor Cost Analysis," Illustration Number 4, shows that computer output needn't be gobbledygook either. This weekly report on a construction project can be easily read by a non-expert. It keeps jargon and confusing abbreviations to a minimum.

The Make versus Buy Decision

In theory, buying a software package should offer the advantages of having:

- Lower start-up costs, and
- Shorter start-up intervals.

In practice, things don't often work out that way. Many experienced data processors are skeptical of these supposed advantages - particularly when modification or changes to the software packages are required. Indeed, the more accomplished computer user often steers clear of purchased software packages all together.

On the other hand, custom designed and programmed software supposedly offers the advantages of being:

- Better suited to your operation needs,
- More flexible,
- Easier to maintain, and
- Less expensive to operate.

But custom designed and programmed software systems are not without peril. If you don't have access to a highly qualified team of systems analysts and computer programmers (either as in-house employees or outside contractors), you shouldn't try to develop a custom system.

A well designed custom system is, in the long run, usually more effective than a purchased system. If your needs are unique in any way, and you can afford to do it right the first time, you should seriously consider custom designed and programmed software packages. After all, you've probably custom fit every other aspect of your business, so why not do the same with you computer software.

What It Should Cost

If software were cheap, the big, experienced users of computers would have known it long ago. Experienced computer users may invest nearly as much (sometimes even more) in software as hardware.

It would be nice if one or two thousand dollars of software were all you needed, but unless your business fits an available ready-made package very closely, you should plan on spending anywhere from 40% to 100% of the computer purchase price on software, depending on your needs.

Once More From The Beginning

When it's all said and done, what you really want is:

- Better Management Information
- Reduced Operating Costs
- More Flexible Business Systems, and a
- Decent Return on Your Investment.

Today's computers have the potential to do all of that in a most pleasing way - but be careful!

Software - particularly applications software - can be the fly in the ointment.

- Check out each component of the system
- Use common sense and be critical
- Ask for professional help in technical areas
- Follow a sound evaluation methodology
- Look for voids, fitness for your needs, maintenance headaches, effective features and operating economics.
- Be very careful about technical support personnel, and
- Use predefined standards or evaluation criteria if you have access to them.

Finally, don't be tempted by low-cost or no-cost software purely because of price - or rule out a custom-design purely because of high start-up cost. But you're not likely to do either if you've been thorough up to the point of cost considerations. Remember that old saying .....

> The bitterness of poor quality remains long after the sweetness of low price is forgotten.