Ten Mistakes to Avoid
When Buying Accounting Software for Use
in the Construction Industry

By Al Blair

After working on both sides of the "accounting systems aisle", I've seen my fair share of mistakes. Some were obvious, some were not. This report shares lessons learned over several decades of providing and using accounting systems. Today, software must be based on the Internet, sold as SaaS (software as a service). Most organizations cannot provide the security, power and flexibility provided by web-based systems. Emphasis however needs to remain focused on the flexibility required to deliver effective management information, not just accounting data. Here then, primarily from an "operating manager's perspective", are ten critical mistakes I would urge you to avoid before subscribing to your new system.

1. Proprietary Database Architecture

Where is your data? How is it structured? If it resides on your office server(s) it may be at greater risk than you realize. In today's world with automatic & extended web backup and redundancy, your accounting & business data is actually more secure in the "cloud" than anywhere else. Importantly, in addition, while most accounting systems use relational databases, and as a result, provide significantly improved integrity, there may still be accounting and business management systems that are not truly "open" and easily accessible. Their tables of accounting data and/or fields of data in their tables cannot be easily accessed by other programs.

What should you look for? Ask to see the underlying database structure, schema and data field descriptions. Do the names of tables make any sense? Are the fields of data described in a manner that would allow your system to be easily integrated with other applications in your organization? In short, can the database be opened and accessed easily? By whom? Only programmers? How about end-users?

Figure-1 is an example of an open database accounting system. Note that the naming convention used for naming various tables actually tells you something. The table of data highlighted in this example is called *td_APPurchaseTypes*. Most users can correctly assume that this table will contain data about accounts payable directly related to the type of purchase that was made for a job.
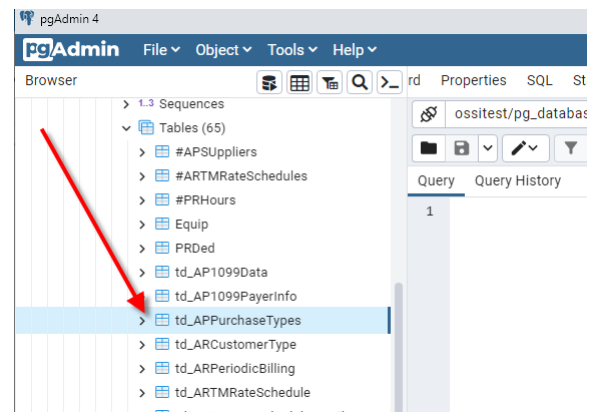

Figure-1

If your prospective new system claims to provide an open database as shown above, then next examine carefully how the fields of data in the database are labeled and described.

Shown in Figure 2 below is an example of various fields of data contained in the *td_APPurchaseTypes* table above.
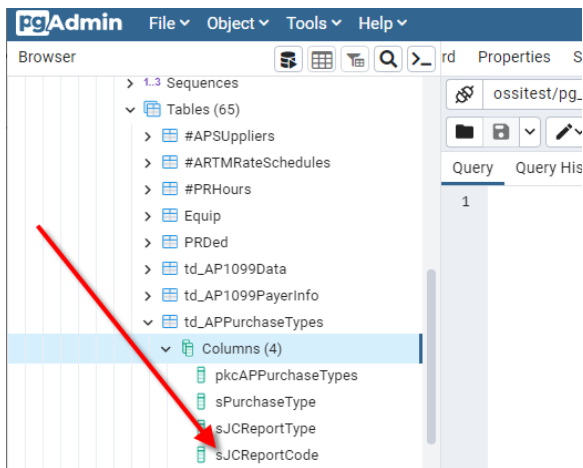

Figure-2

Note that each field of data is fully described. The field of data named sJCReportCode contains the job cost code that you've assigned for a given type of purchase being paid in the AP module in your system.

Why is all of this so important? We have learned after years of work with various organizations of all sizes, that very few accounting systems are purchased off-the-shelf, so to speak, and implemented "as-is". That just does not happen very often, anywhere. Every organization has unique requirements that are typically best reflected in varying management reports and analyses.

Many "high-end" enterprise accounting systems attempt to anticipate these requirements and build into their

products a large number of pre-defined reports. Lower cost systems typically provide a limited number of basic reports. In either case, the real issue is how hard or easy is it to "modify" your job cost accounting system to meet the unique needs of your organization. The key to accomplishing that is an "open" database. We suggest that you avoid systems that have proprietary databases. And watch out for attempts to sidestep this issue by convincing you to use various import/export tools. Other than commonly available web application programs, you should not have to buy anything extra to be able to easily access the data in your job cost accounting system.

eTEK's newer web-based system called eTEK® Online uses an open-source web-based SQL Server database called PostgreSQL (for more info on that database click here). As a result of using this powerful database, you will have access to a number of open-source reporting tools such as Metabase and even "free" tools like Google Data Studio. (We'll talk later about the critical importance of providing ad-hoc reporting to further tailor reports and management information for your business).

2. Lack of Flexibility

Directly related to the need for an open non-proprietary database is the need to make sure that your accounting system can be easily modified and tailored to meet your needs. One of the first things to check in any system you may be interested in subscribing to over the Internet, is how easy it is to email and/or

export any report in the system to commonly used application systems.

Shown below in Figure 3 is an example of an accounting report that provides a direct path to exporting accounting information to HTML for use by other applications such as Microsoft Office or Google Workspace.

Figure-3

The benefit of this kind of flexibility lies in the ease with which information can be shared in a collaborative manner to increase the productivity of everyone.

Other good tests of flexibility are to determine how hard or easy it is to use query tools like those described in general terms in Figure-4 as provided with your system as an integral part of your underlying PostgreSQL database.

Figure-4

The ability to pull data out of your system is critical. Notwithstanding the importance of having an open database, the key is making it relatively easy to get data pulled out into useful management information – and typically, displaying that in graphical forms.

Shown below in Figure-5 (courtesy of Google's Data Studio website) is just a snapshot of what we mean by this kind of test of flexibility.

Figure-5

Flexibility also means being able to call things what you want to call them in your system. So, another test of flexibility would be to find out how hard or easy it is to change the names or labels of reports in the system you might

be considering. Is the system table-driven? Can you easily access the table(s) that provides names of things? After accessing that data, can you change labels and descriptions?

Keep in mind what I said earlier. Perhaps your organization is different, but my experience tells me that the best assumption to make is to assume that the day after you have purchased your new accounting system someone in your organization is going to ask you for new information that has never been produced before.

Assume that your organization merges with another organization. What then? In short, the acid test question becomes how will your system deal with change. Can *you* change your system . . .without spending hours and hours on custom programming work?

We are passionate about the need for flexibility in accounting systems! You should be too. Just make sure that the flexibility you are being sold does not inject another layer of complexity into your world.

Oh, and one more thing: make sure your new system is web-based so you can access it from anywhere at any time 24/7 (and enjoy the benefits of backups and security rarely if ever found in on-premise tractional server/desktop-based systems).

3. System Not Easily Scalable

In practical terms here's what we mean by scalability. Ask yourself this question. If you purchase the system,

you may be considering what would happen tomorrow if your organization doubled or tripled in size?

One of our competitors offers at least four *different* accounting systems for small to medium-sized organizations. They all have different designs that end-users would need to learn how to use.

Even QuickBooks offers varying plans (systems) at varying prices, some online, some on the desktop.

There's a better way. What you need to look for is a system that will allow you to buy *no more or less basic capability than you actually need* to start with, then in case you do need to grow, will allow you to do so *without re-training your users*, regardless of how large your organization may become. We offer one system. Period. It's called eTEK Online.

And the key is scalability based on proven infrastructure open database technology. We have designed our systems so that the same functionality, navigation and user screens are used from our smallest single user system right on up to hundreds of users in a large-scale, enterprise environment … on the Internet. The key to our scalability is our application design combined with the power and flexibility of web-based PostgreSQL.

Our systems have the ability to grow dramatically in numbers of concurrent users, and/or size of database.

The acid-test question comes back to people. People costs, as we all know, are the more significant costs in any system. Scalability means not having to buy a different system as your organization

grows or having to re-train your users every time you expand.

4. Non-Intuitive GUI & Awkward Navigation

The graphical user interface (GUI) and ability to navigate in and around your accounting system is one of the most important keys to making a "difficult" application system a bit easier to use.

Anyone who tells you an accounting system is an easy system to properly set up and start using is either naïve or very much ill-informed. Accounting systems by their very nature are typically a lot more difficult to set up and use than most computer applications. That is why most accounting systems are selected by accountants, Controllers, Chief Financial Officers, and business owners. (We'll talk later about why they often make many of the mistakes discussed in this report).

The key issue is to avoid systems that have a "BIG" ego. What do I mean by that? Systems with big egos typically have spent a lot of time and effort (that you are going to pay for one way or the other) creating their own unique tool bars and menu controls and the like . . . often to accomplish many of the routine tasks like file open, printing, saving, cut, copy and paste . . . that users have already learned to use with their word processing or spreadsheet applications.

Try to select systems that have earned some sort of independent certification related to their ease of use or compatibility with other common desktop applications. One example of that is shown below in Figure-6. This is

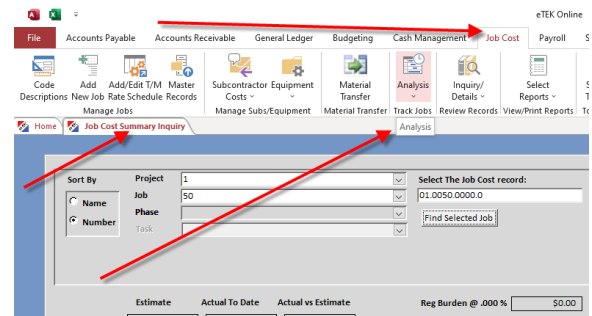a screen that is used to prepare itemized billings.

Figure-6

Look familiar? Sure. Our GUI stems from now decades of fairly close adherence to what Microsoft has adopted and uses in the ribbon at the top of their desktop programs. Just like theirs, ours have various icons and labels to describe the action(s) available to the user.

Note: our tabs across the top from left to right are direct access to the various modules that are integrated with each other and make up our software. Under each tab are logically grouped icons and labels to clearly indicate actions available for the user.

The ability to easily navigate to what you want to accomplish is one of the prime benefits we offer and when comparing to other products we urge you to include an evaluation by your day-to-day users. After all is said and done most systems do the accounting the way accounting must be done. But how hard or easy is it to that? We suggest you take the time to consider whose system will be easiest to use

By selecting a system that provides a consistent and easily understood graphical user interface, and intuitive and natural navigation ability, you will

be focusing everyone's efforts on the application itself. With the natural complexity built into accounting systems simply because they are what they are, practical GUI design goes a long way towards making users more productive.

5. Weak Audit Trails

All accounting systems have to perform accounting functions essentially the same way, namely, in accordance with GAAP (Generally Accepted Accounting Practices). That is assumed. What is not assumed is the presence of practical audit trails built into the system, not so much from the point of view of the accountant or auditor (those are also a given), but equally, if not in some ways more importantly, audit trails on *the use of the system*. Who did what, when . . . to the system?
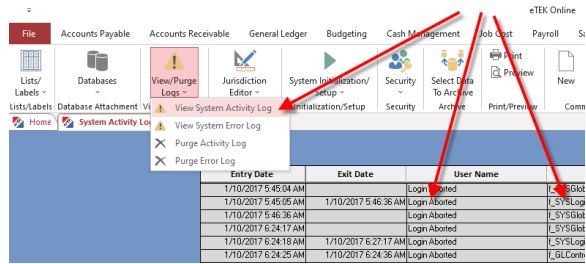
Figure-7

Shown above, merely by way of example, is one approach to providing management (and your system vendor's Support Desk) with a practical internal systems audit trail in the form of an on-line inquiry to a *system activity log*. This kind of audit trail is thankfully found in more and more modern systems and can go a long way towards answering the age-old question . . .. "What were you doing before the system went !@#$%^&*"?

The ability to intelligently fix problems and prevent repeated mistakes starts with the ability to re-create the scenario that led to the problem or mistake in the first place. Too many users underestimate the practical value of system activity logs and/or related audit trails. Don't make that mistake. Make sure your new system has built-in system audit trails that are useful to you, not just traditional accounting transaction audit trails.

6. *Help* Fails to Anticipate Problems

In our own accounting software business, speaking in all candor, I will likely never be completely satisfied with our efforts to provide more effective on-line help, user documentation, training and support.

Most reputable surveys confirm that users of computer systems do not, in fact, feel that the systems they use are as simple and easy to use as they would like. As we add more and more features to our products, the truth is, we have to guard against adding complexity. All of a sudden what we thought was a fairly easy thing to do, and a "cool" new feature to add, winds up becoming too confusing and complex.

We are constantly fighting the battle against complexity in our business. And the ultimate key for us is to focus on providing HELP that really HELPS!

What to look for? Shown below in Figures-8 and -9 are simple examples of how eTEK Online incorporates checklists and trouble-shooting into their on-line help file.
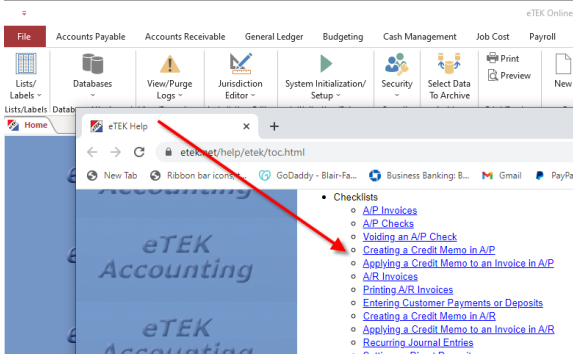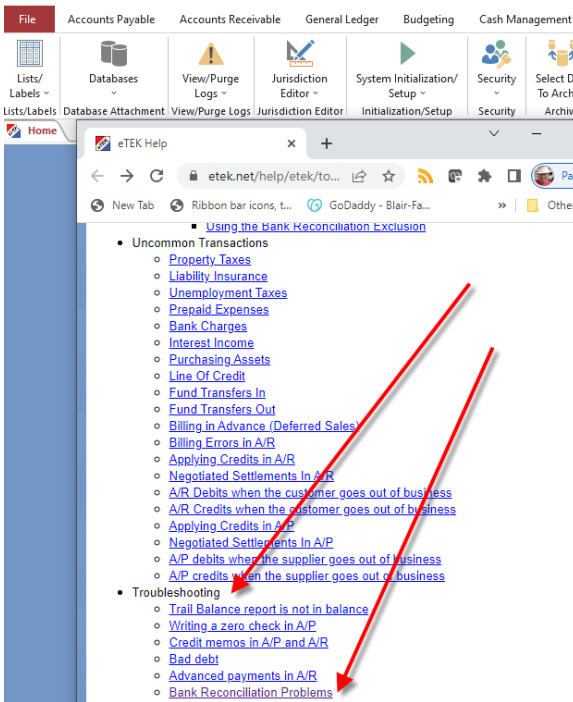

Figure-8

Figure-9

In today's marketplace, I would look for systems where your accounting system provider includes help for their software built-in as integrated part of your system based on same knowledge-base used by their technical support staff so that, beyond the contents of the help file built into your system, you can be assured that your users have immediate access, at any time, night or day, to the latest technical information and updates about your system. We are constantly striving to find better ways to use our web-based system for training and support.

7. Lack Of Compliance With Industry Standards

There are a number of standards that are applicable and important. One is rarely, if ever, discussed in various CPA, Accounting or Financial trade magazines and reviews. It relates directly to flexibility and the ease (or difficulty) in integrating your accounting system with other business applications used in your organization. The issue is: *object-naming standards.*

Without getting bogged down into too much technical detail, please know that in the software industry, the "internals" of software products are almost always evaluated by professional software engineers.

One of the very first things they look for is how all of the various "internal objects" have been named, or labeled by the development software engineers. What they are looking for is logically consistent names that will help them, should they ever be called upon to modify or maintain your system. They want to be able to accomplish their work in as short of period of time as possible, with the least amount of potential for error.

Typical objects contained in the job cost accounting software you are considering consist of the following: tables, views, queries, forms, reports, macros … the list goes on. For simple illustrative purposes, here are the names of two tables in two entirely different commercial computer systems:

pbcatfmt
td_ARCustomers

Can you guess what data exists in the table named "pbcatfmt"? Probably not. Even after I opened this table, since the fields of data were described as "text1, text2, etc.", there was still no way I could tell what information was stored in that database table.

The second table has information about accounts receivable customers. That fact should be readily apparent from the name used. The only portion of the name for the second table that might not be clear would be the prefix "td_". That prefix is used consistently in our systems (and others) to designate a type of table called an attached (or linked) table, well understood by most software engineers.

So why is all of this important? When the day comes that you need changes made, for whatever reasons, that require someone to figure out "where stuff is and what it's called" . . . you do not want to waste their technical skills (and your money) on such tasks. It would be a lot better to have their expertise focused on making the changes you need rather than searching around trying to figure out where the data and objects are that need to be modified or changed.

Compliance with industry standards is much more than just logos and certifications. Don't hesitate to ask for opinions from software engineers you know and respect. I would encourage you to involve them in your evaluation and selection process.

Far too many CFO's and business owners continue to make this mistake. Competent software engineers (hint: look for their certifications) may not know (or really care about) the

difference between debits and credits, but they can sure help you ascertain the degree to which your prospective new accounting system complies with appropriate industry standards.

8. Source Code Not Available

This is one of those issues that is either yes or no. The reason I recommend that you subscribe to an accounting system that includes source code is that this action can provide you with improved long-term security over the investment you make. Regardless of how successful your accounting software vendor may be, your best long-term interests are served when you either have the source code in your possession, or at least have it held by a third party in escrow in the event your vendor either goes out of business, and/or is acquired by another company. One never knows in this business exactly what technical changes may occur in the future that might obsolete your vendor, your investment, or both. Assume the worst. What if you wind up with a system that is no longer supported? If that should occur, you may likely need one of more programmers to help you.

A directly related problem to avoid is buying a system written in an out-of-date programming language. While VBA (visual basic for applications), C# and other languages are often found, in today's world true web-based systems use languages such as Java, Python, and PHP.

In any case, if the changes required in your particular implementation affect how various transactions are computed

and/or processed, including significant revisions on forms (screens) and the "code behind those forms". . . you will need to acquire the source code to your accounting system.

Ask for a non-exclusive source code license agreement that excludes any and all rights to resell the system. If you can, buy from a vendor who will provide a source code agreement for your organization.

9. Is Your System Really Designed for the Construction Industry?

Some of our competitors are not going to want you to learn this …

Look carefully at Figure-10 and note that for every single line item on the AP invoice data entry form shown, there is a separate entry for job cost and general ledger account.

So what does that mean? Too many accounting software programs bog down their General Ledger account number structure (Chart of Accounts) with coding that is needed for detailed job cost reports. When you see that, steer clear. That system was not really designed from the get-go for the construction industry.

Figure-10

eTEK Online automatically ensures that what you have defined and setup for the projects, jobs, phases & tasks you want to track and report on, is synchronized accurately with the accounting codes needed for your financial reports.

At the end of the day, systems designed for the construction industry must not only track costs to the detailed level required, but must tie back correctly into complete profit & loss operating statements.

Directly related to this attribute is the question about reports included in your system. Does the system generate the kind of reports needed by your bankers and bonding agencies? Reports often called Over/Under Billing? Project projections based not only on actual vs. estimated costs, but revenue earned based on estimates-to-complete for given projects/jobs?

Make sure your accounting system is actually designed for the construction industry. From the ground floor up

10. Is the Internet Safe for My Job Cost Accounting?

I don't really have to sell you on using the Internet. Chances are good you already use the Internet for banking, stock investments & critical personal savings and investment accounts.

Today, Internet hosting platforms we use provide better backup, redundancy/recovery; and security, than most organizations of any size can afford to install and maintain.

Sure, there are antidotal stories about failures from time to time, but those are very infrequent and increasingly rare. For our customer, the small contractor in the construction industry, there frankly is little

chance you can afford to protect your office desktops and servers and networks anywhere close to the environmental equivalent to that provided by the web hosting services we use.

Let's be specific. Assume the worst possible case for your business. A storm or tornado levels the office that has your PC's and accounting software/records … and tomorrow a payroll is due along with required government reports, etc.

With your system and data on the Internet you crank up your home PC/tablet/phone and get the job done. Because you have not lost your ability to get the job done. Your system is backed up all the time, typically with redundant servers on either coast and elsewhere.

Trust the Internet for your job cost accounting. It's the very best option available today in my opinion.

Conclusion

I have been asked why this "Top Ten" list does not contain issues that are focused on professional, accrual-based accounting issues of interest to accountants. Here's why.

No accounting software company will succeed if their systems do not process and report accounting transactions correctly. That is a given. Determining whether a system posts detailed debits and credits to the trial balance and then updates the general ledger in a properly controlled and auditable manner, does not really drive today's concerns about selecting an accounting system. Any experienced accountant can quickly tell if the accounting is done properly. But not all accountants necessarily have the perspective that an operating manager has.

It is therefore crucial that you examine accounting systems not just from an accountant's point of view, but rather from an "owners/operating managers" point of view as well. After all, these are the *real clients* of every system we sell . . . because they are ultimately the most important constituency CPA's, accountants, controllers and CFO's must serve.

"Give me the information I need to manage our business better than our competitors. Give it to me in a timely and accurate manner, and in a form I can use."

The above statement from an early mentor, and now close business associate, says it all. It reflects what our customers want from their accounting system - especially with the ability now to collaborate and share information worldwide via web-based capabilities. *Please accept the fact that the Internet has changed forever how information should be exchanged, shared and used! For all kinds of systems!*

We exist to help those who are charged with meeting the information needs of key managers – the people who are actually *managing* professional construction businesses and services. We strongly believe that the "doers" in your organization deserve "a say" in the new system you may be about to subscribe to as Software-as-a-Service running securely on the Internet.

As should be clear by now, we view an accounting system a little bit differently than others. Systems do not plan, organize and control. People do. And more than likely these people are led by "operating managers", not staff personnel and support organizations.

In conclusion, we believe the ultimate test of an accounting system is how well it provides *management information to managers* – not just accounting data to accountants.

I sincerely hope this brief report helps you make sure your next accounting system has the "open-ended-ness" and flexibility you need. We certainly do not have all of the answers. No one does. But if you have found at least one idea in this report that will help you make a better-informed buying decision, then our primary objective has been met.

Al Blair is the founder of eTEK International, Inc.

He has several decades of experience in information technology for large-scale business and government organizations, as well as commercial-grade independent software development firms.

He has taught and consulted in certified client/server rapid application development methodologies and business process re-engineering.

For more information about *eTEK* International and their products and services please call 800–888–6894 or visit http://www.etek.net