The Importance of Power/Precision Data Entry to Document Imaging

A White Paper

VIKING SOFTWARE SOLUTIONS

A Division of Phoenix Software International

6660 S Sheridan Rd., Suite 202
Tulsa OK 74133-1730
**(918) 491-6144**
Fax (918) 494-2701

Table of Contents

Introduction

One of the often overlooked and underestimated elements of a Document Imaging project is the automation and human cost of lifting data from the images. Systems engineers frequently respond to data entry requirements with automation-only solutions that fail to recognize the accuracy demands of most applications. We have all heard horror stories about lost images due to improper indexing and major errors due to inaccurate invoices, claim forms and etc

The off-the-shelf key entry component of most document imaging systems is seldom up to the job. They lack the attention to data entry cultures and automation costs that are critical in high volume, high accuracy data entry operations. Ongoing labor costs for data entry are often the largest element of the operating costs. Unnecessary operating costs are the result of failure to provide state of the art data entry methods and techniques. Controlling these costs can mean the difference between economic success and failure of imaging projects.

The diagram below shows how the data entry components relate to other front-end components in a generalized document imaging system.
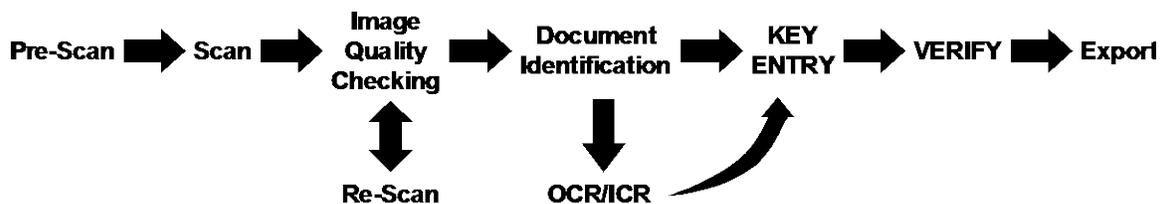
Most data entry modules for image systems were designed by Windows programmers and are excellent for casual use. Nevertheless, they are inefficient for high volume, high-speed data entry. For decades the national average for keyboard data entry has been about 12,000 keystrokes per hour. Anything less is unacceptable to organizations concerned about costs.

The professional data entry people in your organization can offer valuable insights into these issues and their importance. Be sure to involve them in the evaluation process.

Many document imaging systems are designed to do most of the indexing and data entry by a "keyless" method using ICR (or OCR) engines for automatic character recognition. In well-controlled circumstances this technology will reduce the key-entry labor dramatically. However, in the real world, using uncontrolled documents, recognition rates seldom approach the advertised rates. Unrealized expectations and undetected errors can cause big problems. The remainder of this paper will address the issues and explain the importance of data entry methods and techniques that will be required by successful document imaging systems. The discussion is divided into five categories:

- **Importance of Data Accuracy**
- **ICR/OCR Accuracy Issues**
- **Operating Costs**
- **Application Development Costs**
- **Other Issues**

I.   Importance of Data Accuracy

A. Cost of Quality

Errors in data cause a variety of problems and raise the costs in several areas. The cost to recognize and detect errors is not trivial. Further costs are incurred to correct the data errors. The largest cost components are the hidden costs that affect other departments.

B. Error Detection

Detecting data errors in programs often takes as much, or more, of the analysis and programming efforts than the main logic. The earlier an error is detected, the cheaper it is to correct it. Fifty years ago it was common to "key verify" punch cards. This technique is still one of the best methods for detecting errors. Re-key verifying selected data items, combined with programs that look for invalid data, can detect most data errors.

C. Error Correction

Correcting errors programmatically is the cheapest way. Doing it with a single keystroke when the data is initially keyed is the cheapest manual method. Conversely, it costs hundreds, or thousands, of times more to create update transactions to fix errors that remain undetected until later in the process.

D. Hidden Costs

The hidden cost of errors is much higher. For example, customer service problems increase proportionately to the number of mistakes that result in billing errors or shipping the wrong product, or potential costs of litigation resulting from bad data. You know the many hidden costs in your business.

II.   ICR/OCR Accuracy Issues

A whole specialty knowledge area has developed around how to design and tune character recognition applications. The state of the art is constantly improving, but it still requires very scarce technical resources to be successful. It is beyond the scope of this paper to discuss ICR and OCR technology. However, we will discuss some of the error detection and correction issues.

Many ICR systems claim 95% to 99% data accuracy. This sounds great on the surface, but in fact may be unacceptably low. When one understands that this rate is on a "per character" basis and that ICR engines reject characters that don't meet image quality specifications, these advertised rates become increasingly unappealing. Only 95% accuracy means 5% errors. A typical data input transaction consists of more than 60 characters. How many applications that have an average of three errors on each transaction would be considered successful?

A. Error Types

There are two classes of recognition errors, unrecognized characters (called "**rejects**") and erroneously recognized characters, or "**substitution**" errors. Rejects need to be corrected. Substitutions must first be detected and then corrected.

ICR engines can achieve very high recognition rates when the documents are properly designed, printed and controlled. Exploitation of corporate data and use of "fuzzy logic" expert systems can further boost ICR results. Nonetheless, a 1% error rate on a printed page with 3,000 characters means there is an average of 30 errors on each page. This would be unacceptable for a typist,

but may be adequate for information that will only be read occasionally and if the human reader or text classification engines can deduce the correct information from the context.

However, today's ICR/OCR systems have unsatisfactory error rates for most real-world documents. It is a continuously developing technology, but it is projected to improve only incrementally for the next few years. Furthermore, data to be processed by a computer must be very accurate for most applications. The true measure of the value of ICR is the cost of doing it plus the cost of error correction. This is highly application dependent. Many ICR projects are highly successful, but others cost more than simply keying the data.

1. Reject Correction Costs

On the surface, reject reentry seems simple and straightforward. An image of the rejected character is presented to the keyer who depresses the correct character key and the program moves to the next reject character. Reject reentry modules must address three problems:

- missing characters
- extraneous characters
- key entry speed

a. Missing Characters

Sometimes the recognition engine cannot separate adjacent characters and treats them as a single unreadable character. In this case the operator must be able to insert extra characters. The operator must take extra time to diagnose the problem, which affects the throughput. This process must be ergonomically efficient to achieve maximum keying speeds.

b. Extraneous Characters

Smudges and other extraneous marks on the paper may appear as an additional unrecognized character. The operator must decide what the problem is and delete the extraneous character. Again, the ergonomics of the program design strongly influences effective keying speed.

c. Correction Entry Speed

Data is usually keyed from paper documents at average rates between 10,000 and 15,000 keystrokes per hour. Keying from images should be slightly faster since the keyers do not have to remove their hands from the keyboard to turn the pages of the source documents.

With some systems, reject reentry keying rates may be only 5 to 10% as fast as full keying rates. A 1994 Imaging Magazine study show an average reject reentry rate of 386 characters per hour, or about 3% of the full keying rate. One reason for the slower keystroke rate is that presenting keyers with only a single character at a time prevents them from building any rhythm. Smooth keying rhythm is essential to fast keying.

d. Two Cost Factors

We see that two factors influence the cost to repair rejects:

- Accuracy of the recognition engine, and
- Reject reentry speed.

The facts show that if the recognition engine is only 90% accurate and the reject reentry rate is only 10% as fast as full keying, it is just as cheap to key the data. On the other hand, increasing the recognition to 95% will make it very worthwhile. Increasing the reject reentry rate also leads to substantial cost reductions.

Since reject characters often occur together, the ability to simply re-key the entire field, or even the entire record, greatly speeds up reject reentry rates. This is because the operator can often key the entire field faster than repairing several rejects. This capability is very valuable for forms and documents that are subject to being damaged, marked, stained, or wrinkled.

Reject repair programs for high volume applications must be well designed to be cost effective. The value of recognition engines depends on many factors and it also varies from one application to another.

2. Substitution Error Costs

We must first consider the error detection component. Correcting rejects is easier than substitutions because the recognition engine finds the errors for us. With substitution errors we have to find the errors that fooled the recognition engine. Some errors can be detected programmatically with spelling checkers, data base lookups, deduction from other data, balance calculations and semantic analysis. Programmatic error detection is highly application dependent.

a. Manual Detection Methods

There are also two manual error detection methods. Proofreading (or sight verification) is the most common method. It is not especially accurate because the mind has a way of fooling our eyes. Nevertheless, it finds many errors. Surprisingly, continuous proofreading is not as fast as key entry. Some reject reentry systems provide for selected proofreading while the rejects are being corrected. This can be a valuable feature.

Re-key verify is the time proven method of manual error detection. Over decades of use it has been proven to be about 99.9% accurate. The cost is similar to the cost to key the data. However, usually not all of the data has to be verified which saves labor. Data elements that can be programmatically validated, or whose accuracy is not important, do not need to be key verified. Good data entry programs have this capability. Verifying only sample portions of the data is a statistical method used to detect problems with equipment and personnel.

b. Correcting Substitution Errors

Once the error has been detected, the issue of how it is to be corrected must be addressed. Some systems allow the error detection program to "mark" substituted characters as a rejected character and send the data record and its image back through the reject reentry process. Re-key verify programs should provide an efficient method for immediate error correction. Some are more efficient than others.

B. Other Factors That Influence Recognition Rates

Recognition accuracy depends on many factors. Good quality documents with well-delineated text is the first requirement. Paper color and type also affect the quality of the scanning and resulting image.

Good text delineation implies that cursive handwriting is beyond the scope of today's technology except for very highly specialized cases where the system can "learn" the unique characteristics of the author's handwriting.

1. Document Recognition

Document recognition is clearly a prerequisite for formatted forms. If the document type is unknown then the recognition engine will not know where to look for the data. This may mean that not only does the type of form need to be recognized, but also the particular version of it. In these instances, character recognition doesn't help much with "content recognition". When forms recognition is difficult, it may be cheaper just to key enter all the data. Or, plan to do manual entry

in the beginning until satisfactory document recognition technology is in place, the forms are redesigned, and the old forms have been removed from circulation. The point is, plan your image data entry for reality.

Character recognition technology is changing rapidly. Proprietary systems that are locked into a single recognition engine may result in less than optimal performance today. They risk falling behind the technology curve in the future. No one knows what will be the hottest engine next year.

2. Recognition Technology

Some engines are much better on some kinds of applications than others. If you have a variety of documents it may be desirable to consider multiple engines.

III. Operating Costs

Operating costs go on for the lifetime of the image project. Typically they will be orders of magnitude higher than the acquisition and implementation costs. Some operating cost elements associated with data entry have been previously mentioned. The four principal issues are:

- Maintenance Costs
- Data Accuracy
- Keying Speed
- Controls and Auditing

A. Maintenance Costs

What are the annual maintenance fees for licensed software?

What will it cost to maintain the hardware? Systems that depend on highly specialized components not only cost more to acquire, but their maintenance costs will be much higher than standard components. For example, high performance workstations and monitors usually must be maintained by the manufacturer, while common PC's are inexpensive to maintain, replace and upgrade. Power consumption and other environmental factors must be considered.

How much will it cost to maintain and enhance products developed in-house?

The tremendous growth of the packaged software industry speaks to the fact that it is usually much cheaper to maintain licensed software than to maintain systems developed in-house. This is especially true for dynamic new technologies like imaging. You expect your software vendor to keep up with the advances in technology and improved methodology. Are you willing to make a similar commitment of in-house resources? If not, you will soon fall behind and you will not achieve the cost savings to which you are entitled and which management expects.

B. Data Accuracy

Previously, we discussed the importance of this topic. Now we will discuss some techniques and methods the data entry software should offer to allow you to maximize the accuracy of your data. These are features that will be found in a good key-from-image program.

1. Validation Features

Character Sieves. The first line of defense against keying errors is to eliminate invalid keystrokes as they occur. For example, the system should not allow alpha characters in numeric fields and provide for even more sophisticated single character validations.

Field Edits. As soon as a field is entered, it needs to be edited. You need a wide variety of edits, ranging from simple numeric range checks to database lookups and computed values. The common functions should be provided by the system and there should also be a mechanism for easily adding custom edits that are unique to the application.

Field edits should have multiple levels of capability. For example, the first level date validation is to insure the month is in the range from one to twelve. A deeper level detects future dates and unreasonably old dates.

High Speed Table Lookups. Comparing field values with database tables of acceptable values and other information is one of the most important types of field edits. Table lookup with substitution is a valuable "keystroke saver" that improves productivity. However, this process must be extremely rapid and efficient and never delay the keyer. Table lookups that disrupt keying rhythm may be counterproductive. This requires very good programming techniques and well-designed systems to be effective.

Computed Values. To achieve maximum productivity and accuracy the system must provide for flexible and fast field edits to compute values based on one or more data fields and database values.

Check Digits. Many fields, such as account numbers, have a self-checking digit in the number that can be used to detect errors in the value. Dozens of check digit algorithms are in use and the data entry module must be able to handle them all.

Field Duplication. A common method of keystroke reduction is the capability to duplicate the values in previous fields and from previous documents. Often the documents are grouped by location, date, account number and other fields that can be duplicated. Default values are a special case of duplication. The system should provide for a flexible and powerful way to duplicate data. The keyer and the system both should be able to duplicate information from other fields.

Context Sensitive Help. The system should provide on-line help to guide the keyer in the rules for entering data. Help messages should present information for the data field being keyed. It should require a minimum of keystrokes to request and terminate the help messages. Multilevel help messages with increasing orders of detail are even more useful. It must be easy to modify and change the help messages because data entry forms are subject to change and the needs of the keyers may not be well defined initially.

Optional and Required Fields. Some fields must always be entered and the system must enforce this requirement. Other fields are optional. Some optional fields are seldom entered so keyers will be more productive if the default is to automatically skip them. But there must be an efficient way for the keyer to enter the fields when necessary.

Intelligent Field Skipping. The best systems have the ability to dynamically skip fields based on the data entered in previous fields. This feature speeds key entry and reduces the opportunity to make errors.

2. Re-key Verify

As mentioned previously, the time proven technique for improving data accuracy is to independently re-key the data and compare the results. Accuracy is much better when the keyer doing the verify step is not the one who originally entered the data. To be cost effective, this process must be very efficient. Not every field needs to be verified, so selective verification is required. The system should recognize certain conditions for dynamically skipping verify fields to further reduce the effort.

Usually the verify keyer is more experienced than the original entry keyer and should be able to correct the data. The correction process must be very efficient in order to minimize the time to verify data. For example, single character errors should be easily corrected when they are detected. Likewise, an entire field may have to be replaced.

Re-key verify is also used as a means to measure the results of programmatic validation techniques and tune them for improved performance.

C. Keying Speed

There are many factors that influence keying speed. There is no single overriding factor, but a whole series of little things that couple together to allow the keyer to reach their maximum potential. Interestingly, studies have shown that the fastest keyers are also the most accurate. This means that these ergonomic factors are important even if you do not expect blinding speed from the keyers.

Fully keyed data entry is accomplished at average rates of 12,000 keystrokes per hour. Superior operators will key at 50% or more above the average rate. They can only do this if the data entry system is well designed, allows the keyer to maintain rhythm, and minimizes hand movement. The techniques and methods for high speed key entry are evidently not widely understood and appreciated. A new generation of analysts has grown up with a mouse and has no concept of what is involved in keying at 20,000 keystrokes per hour. Production data entry is very different from the casual data entry we use with spreadsheets, data bases and other transaction-oriented applications.

1. Keying Rhythm

Rhythm is a very important factor in high speed keying. The best operators talk about the importance of maintaining a smooth rhythm to the keying. Furthermore, fast keyers actually look ahead of where they are keying. There is a discernible time lag between the eyes and the fingers.

2. Keypunch Style Keyboard

The fastest keyers traditionally have used the so-called "029 keypunch" keyboard layout that has the numeric keys underneath the right hand. This allows numeric characters to be keyed at the maximum rate without moving the hands from the home keys. This keyboard has proven to be the fastest for alphanumeric data. Good data entry systems emulate this capability on a standard keyboard.

3. Enter Key vs. Tab Key

Data entry applications have used the ENTER key to signal the completion of a data field. This is a large key under the strong right hand. Windows applications use the TAB key for this purpose. It is a small key located under the left hand. This use of the TAB key makes it unavailable for "tabbing" over optional fields.

Removing the hand from the keyboard to manipulate a mouse to perform special functions will not be nearly as efficient as assigning the special functions needed by data entry keyers to a function key that can be depressed with a single keystroke.

4. Statistics

Statistics should be kept regarding the number of errors and who made them. This is useful both for evaluating the keyers and to refine the system. Reports should be available that are organized by operator and by job. It has often been said that, "if you can't measure it, you can't manage it"

Traditional data entry systems have measured the keyer's entry rate in keystrokes per hour. This metric must be carefully considered as it may not be applicable when entering data from images or when correcting rejects from recognition engines. However, it can be a good comparative measure for keyers doing the same work. A better metric may be error-free records per hour, or documents per hour. Documents per hour is better for evaluating changes to the system.

It is not possible to compare average keystrokes per hour between different systems. The method of allocating the time to keyers varies from one program to another, so it is an "apples and oranges" problem.

D. Control and Auditing

The data entry supervisor must be able to know what jobs are in the system and where they are in the process. They must be able to control the work assignments to keyers. Controls must be flexible because priorities change as deadlines approach. The system must be auditable to ensure that all the work coming in has been processed and is not lost. It is desirable to know if it is being accomplished in a timely fashion.

1. File/Batch Navigation

The system must provide facilities for the keyer to view the various images and data records in a batch of work. They should be able to easily go backwards to review previously entered work.

2. Search Mechanism

A flexible search mechanism is needed to find records and images based on the contents of the data.

3. Modify Existing Data

The keyer must have the ability to change data that was previously entered. This includes both key entered data and data from recognition engines. There should be safeguards to prevent data from accidentally being changed.

4. Character Insert/Delete

The keyer must have the ability to insert and delete characters. The most efficient mode for high-speed data entry is type-over mode where characters are replaced by typing over them. However, when correcting existing data it is desirable to select character insert mode so that characters can be easily inserted and deleted from the text. Good systems provide both modes.

5. Image Display Speed

The system should never prevent the keyer from reaching their maximum keying rate. Some keyers are capable of speeds in excess of 20,000 keystrokes per hour. Simple tests, such as holding down a repeat key, can often be used to determine if the data entry system can display images fast enough.

Image display speeds are often the limiting factor and they should be studied to determine the effect on overall productivity. Image display speed depends on many factors, such as:
- CPU speed of the workstation
- Display monitor
- Image resolution
- Display parameters: e.g., color, grey, etc.

- Decompression algorithms

IV. Application Development Costs

The costs to develop the data entry component of document image systems vary widely. Some systems provide easy-to-use tools that allow you to quickly and easily create the applications. Other systems use closed and proprietary technology that requires extensive training and sometimes you must use their consultants to set up the applications.

Systems that use the same application development procedure for keying and reject reentry are simpler to use and also have a consistent interface for the keyers.

V. Other Issues

There are many other issues that affect the performance of an image data entry system. Space does not permit a thorough examination of these factors and they will only be mentioned briefly.

A. Operating System

The most common operating system for an image data entry platform is Windows. Most of the imaging components are designed for Windows. The familiar interface makes it easy to learn to use systems and to integrate components. Although the Windows graphical interface is inherently more complex and inefficient than traditional charter based interfaces, today's high speed PCs generally offset this disadvantage. However, standard Windows dialog boxes are inadequate for professional data entry operators.

It is crucial that the key-from-image module provide the features and functionality to enable the data entry operator to key at their maximum potential speed. Rates far in excess of 12,000 keystrokes per hour are routinely recorded by professional data entry operators. Be certain that your system will provide this level of performance.

Interactive terminals connected to a multi-user UNIX system can be used as data entry workstations. However, they are usually not price competitive. In today's technology, a PC network in front of the Unix or Linux system is most common.

Block mode terminals attached to mainframe computers are not nearly as productive as character interactive systems and they cost much more to procure and operate. Data entry is a highly interactive activity.

B. DBMS Compatibility

The data entry system must provide some compatibility with the DBMS used by the rest of the image system. It may only need to be able to import/export data files.

C. Proprietary Issues

Systems that are tightly integrated to proprietary hardware and software components have advantages and disadvantages. On one hand, proprietary systems tend to be the most uniform and there is a single point of contact. On the other hand, systems composed of "best-of-breed" components are more likely to have the best performance and lowest cost. As in all things, there are always gray areas in between these two extremes to consider.

D. Multiple Image Formats

The trend in the industry is towards TIFF formats, but there are many other formats in common use. Good data entry modules can handle a variety of formats interchangeably.

E. Workflow

The data entry component must provide workflow capability for images and data that are in the various stages of data entry. It should be mostly automatic with manual overrides by the supervisors.

The system should also be compatible with the workflow mechanism used by the rest of the document imaging system. Hooks and handles should be available to provide the proper interfaces.

F. Flexibility and Adaptability

The system should be flexible to accommodate growth and change. New applications will arise and new technologies will be added to the system. Recognition engine technology is on a steep upward curve. The best engine today may be superseded by something much better tomorrow. Data entry components must adapt to rapid change. The system should be able to integrate with stand-alone data entry components.

G. Importing Data

A flexible method to import data into the system is clearly needed. Systems that are locked into single data formats and databases are of limited value. This is especially true in the case of multiple recognition engines.

H. Exporting Data

After the data entry function is complete, the data must be exported to the downstream applications and databases. Portions of the data may be exported in different formats. A flexible export module is required.

VI. Conclusion

Data entry often contributes the largest cost component for document imaging systems. The costs can be classified in the following three categories:

Data Capture Costs
- Error Detection Costs
- Error Correction Costs
- Hidden Costs

Well-designed systems can minimize these costs, but attention to detail and time-proven ergonomic techniques are essential. Keyless data entry technology is improving, but manual data entry and reject reentry is going to be required for years to come. Organizations will not be well served by glossing over this issue. Rather, they should be certain that the data capture components of their document imaging systems are well suited to their requirements. Superior data entry modules are available as off the shelf products.