The Dynamics of Cost In Document Capture

MicroAdvantage™

document management systems & services

16692-B Hale Ave. Irvine, CA 92606
Phone (949) 955-9111 | Fax (949) 955-9191
Email: microAdv@aol.com
Internet: www.microAdvantage.com

Table of Contents

Putting Document and Data Capture in Perspective

Production scanning sites are typically designed with either document capture or data capture as their primary purpose. The difference between the two is straightforward:

- In a *document capture* installation, documents are scanned and permanently archived. Only a small amount of data is extracted from each image, and its purpose is to serve as a way of indexing the images for later retrieval.

- In a *data capture* installation, the documents are usually forms and the images are not always saved after scanning. Instead, every field on the form is extracted and the resulting data is then stored in a database or line of business application.

In general, capture subsystems of both types are fairly inexpensive to purchase (consisting of little more than scanners and PCs) but extremely expensive to operate due to the ongoing labor cost of operating the scanners and validating the data extracted from the images. This makes them prime candidates for automation and cost reduction.

Production Scanning

Production scanning sites can generally be distinguished by three primary characteristics:

- **High volume**. Production scanning installations typically process 1,000 pages per day or more.
- **Highly structured**. Users perform the same tasks over and over.
- **Mission critical**. If the scanning system goes down, core business processes also go down.

Automation efforts usually pay off only at production sites. At non-production sites, automating the capture process makes little sense, since either the volume of documents is low or else the usage is so random that automation is not practical.

In both document and data capture, savings are measured in seconds, multiplied by thousands of documents. How much can one second save? Consider:

- A clerical worker who is paid $10 per hour costs .28¢ per second.

- If you cut one second off the time of processing one document, and you handle 10,000 documents per day, that one second can save you $28 per day.

- At 5 days a week, 52 weeks per year, this amounts to $7,280 per year.

And that's just for one second. The savings can be truly enormous if you are able to streamline your operation even more.

Of course, operating cost is only part of the story since the long term costs of inaccurate data can also be quite large, involving both business losses and legal consequences. Guaranteeing the integrity of the capture process, while at the same time reducing its cost, is the goal of production capture software.

Understanding the Elements of Production Capture

Production capture encompasses a complex flow of processes that includes scanning but extends much further. In general, production capture includes six operations, namely, document preparation, scanning, recognition, indexing and data validation, QC and rescanning, and release.

Document Preparation

Document preparation is an important first step in assuring a well-functioning production capture process. Key manual tasks include inspecting and separating documents, grouping documents into like categories, and designating the beginning and end of documents and batches.

Scanning

Scanning refers to the actual transformation of paper documents into digital images. Alternatively, existing image files can be imported into the system. Effective scanning requires precise control over a wide variety of scanners and scanner settings, including resolution, contrast, simplex or duplex operation, advanced thresholding options, etc.

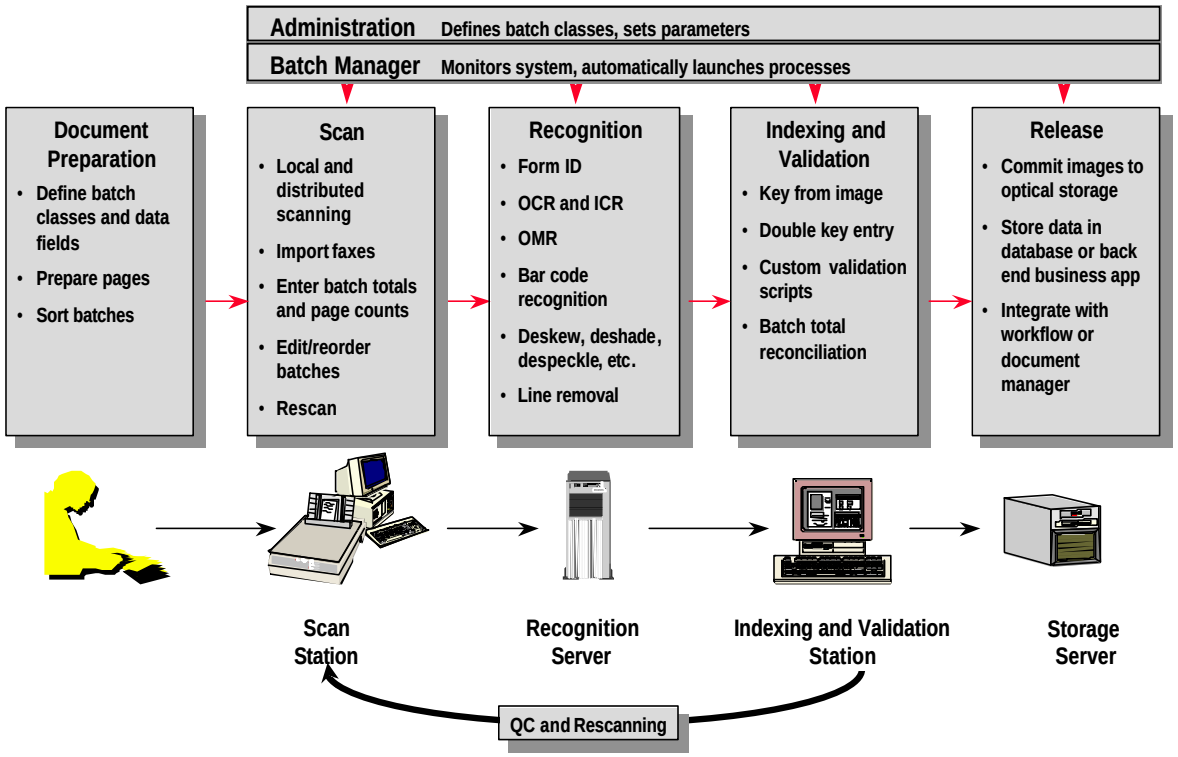


Figure 1. Production Capture Process Overview

Recognition

There are five primary types of recognition performed by production capture systems:

- Form ID is used to automatically recognize different form types
- OCR (optical character recognition) is used to recognize machine printed characters
- ICR (intelligent character recognition) is used to recognize handprinted characters
- OMR (optical mark reading) is used to recognize check boxes, filled-in bubbles, etc.
- Bar code recognition is used to read and extract information from pre-printed bar codes

OCR is the most common type of recognition and is generally broken into two types: zonal and full-text. *Zonal* OCR is typically used on forms, where only specific fields on the form are of interest. *Full-text* OCR is used on free-form documents, such as legal briefs, to read the entire document and then prepare a searchable, full-text index of the document.

Image cleanup is also performed in the recognition step. Techniques include:

- Deskewing, despeckling, deshading, streak removal, and other basic cleanup functions
- Line removal and character reconstruction for use on forms
- Edge enhancement, which sharpens character edges to increase OCR accuracy

The purpose of image cleanup is not usually to make the image more readable, but rather to remove unwanted noise that can decrease the accuracy of automated recognition.

Indexing and Data Validation

Data can be extracted from images automatically via some type of recognition process or manually by a keyboard operator (an operation known as “key from image”—typically used when the accuracy of automatic recognition on a zone is too poor to be useful). In either case, the data must be validated and verified, sometimes by a second independent operator and sometimes via automated processes such as database lookups and built-in business rules.

QC and Rescanning

Quality control entails systematic reviews and checks to ensure that the scanned images are readable. QC includes methods for flagging bad images and explaining why or how images should be rescanned, and can be performed either by a dedicated QC operator or by a “key from image” keyboard operator.

Release

Release is the final stage of the capture process, and consists of handing off batches of in-process images and data to the back end business application. Typically, this is when the document images are written to optical disk or other long-term storage, and the associated data is merged with the document database of the larger system. In addition, the release of a document might trigger a workflow process, initiate the foldering and filing of documents, etc.

Analyzing Capture Costs

The most straightforward way of determining the costs of production capture is to examine both the initial capital costs and the ongoing labor costs of each step in the capture process:

- **Capital equipment** is the most visible aspect of the cost of capture. Capital costs include high-speed scanners costing anywhere from $5,000 to nearly $80,000 each, PCs, specialized image processing accelerators, high-resolution monitors, and so forth.

- **Direct labor** consists of the people who prepare the physical documents for capture, scan the documents, check for quality, perform manual keying and data validation, and integrate the resulting information into the back end system.

The table below summarizes the capital and labor costs involved in each of the steps of production capture.

Table 1. Capital and Labor Costs of Different Production Capture Operations

Capture Operation	Capital Cost	Labor Cost	Comments
Document Preparation	None	High	Purely clerical task, but can be automated using advanced technologies.
Scanning	High	High	Capital costs include the scanner, controller board, and PCs. Labor costs include scanner operators to operate the scanners.
Recognition	Medium	None	Capital costs limited to PCs and, sometimes, accelerator boards. Usually unattended.
Indexing and Data Validation	Medium	Very High	Capital costs limited to PCs. Very high labor costs (typically 2-3 data operators per scanner), but considerable scope for automation.
QC and Rescanning	Medium	Medium	Capital costs limited to PCs. Labor costs include QC operators to inspect images and some additional scanner operator cost.
Release	Low	None	Capital costs limited to PCs. If done after hours, can reuse PCs used for scanning and indexing during the day. Usually unattended.

Scanning is by far the most expensive capital cost, thanks to the scanners themselves and the scanner controller cards. The capital costs of the other operations are generally quite low.

However, as the rest of this white paper demonstrates, labor costs normally dwarf the initial capital expenditures of a capture system. The biggest culprit is the ongoing cost of the operators at the scan and indexing stations, and this is the area that should be targeted most aggressively for cost reduction.

Strategies for Reducing the Cost of Capture

For the balance of this white paper we will be presenting concrete strategies for reducing the cost of production capture, piece by piece and second by second. Not all of these strategies will apply to every installation, but by carefully choosing the ones that can be implemented at your site you will be able to shave precious seconds off your capture process and thousands of dollars from your back room labor expenses.

Use batch processing to speed up scanning

Batch processing is critical to getting the maximum throughput from a high-volume capture system. If pages are fed manually and indexed or validated on the spot, the actual throughput of even a fast scanner can be as little as 5-10 ppm. In a batch operation, by contrast, entire batches are fed into the scanner, then OCRed, validated, and finally released. This "assembly line" operation is far more efficient than manual feeding. Some arithmetic shows why:

- If documents are fed manually, a reasonable estimate for the time to scan and validate one page with three data fields is:

  5 seconds to scan
  2 seconds to switch between scanning and validation
  12 seconds to index or validate

  The total time per document is 19 seconds and the total time for a 100-page batch is 1900 seconds (about 31 minutes).

- In a batch operation, a good estimate for feeding the same 100-page batch is:

  30 seconds to load the scanner
  150 seconds to scan (assuming a 40 ppm scanner)
  30 seconds to load the batch at the index station
  12 seconds to index or validate each document, for a total of 1200 seconds for the batch

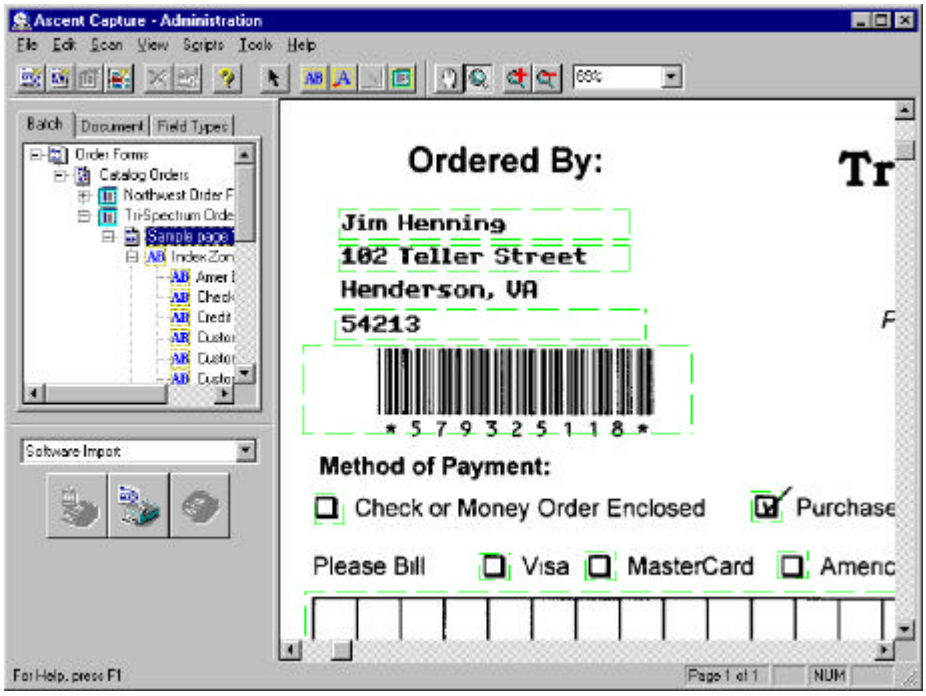
  The total time for the batch is 1410 seconds (about 23 minutes), a savings of 8 minutes.

At a rate of .28¢ per second, you've saved $1.37 for just this one batch. If you process 100 batches per day, 260 days per year, this amounts to $35,620 per year.

Use OCR, ICR, or OMR to automate data extraction

You should always consider using OCR or ICR to automatically extract data from documents. This is especially useful on forms, where information (such as a name or an ID number) is contained in specific locations on the form and can be extracted directly from the image.

The savings from OCR varies dramatically depending on how accurate the OCR is. To decide if OCR is appropriate for a particular field on a particular document, you need to figure out whether it takes less time to key the field manually or to OCR it and validate the OCR results. Here's how to do the analysis:



Ascent Capture allows an administrator to define zones on a form, which can then be used to automatically extract data via OCR, ICR, OMR, or bar code recognition.

1. First, figure out how much time it takes to manually key the field. For example, if a field on a form averages 10 characters in length and your keyboard operators can type 10,000 characters per hour (the most frequently cited industry average), it takes 3.6 seconds to key the field. In real life, you should add about a half second per field, so figure the total time would be about 4.1 seconds. For 100 documents, the total is 410 seconds.

2. Next, figure out how accurate the OCR is on the specific documents you will be using. The only way to do this is to perform tests on real pages, since OCR accuracy varies widely depending on how clean the original documents are.

   In this example, assume that we have good quality documents and the *per-character* accuracy of the OCR engine is 97%. The next step is to figure out the *per-field* accuracy. If each character has a 97% chance of being accurate, the chance of every single character being accurate is .97 multiplied by itself 10 times, or 74%. Therefore, since the chance of the entire field being correct is 74%, the chance of error is 26%.

3. Finally, figure out how long it takes to check each OCR field and how long it takes to correct OCR errors. The time to check is usually about 2 seconds, although testing might provide a more reliable figure for your particular site. The correction time is 4.1 seconds (the same as in step 1), but this is only done 26% of the time. Therefore, for 100 documents, the total time is (2 * 100) + (4.1 * 26), or 306 seconds.

In this scenario, OCR pays off, saving 104 seconds on a 100-document batch. However, OCR is extremely sensitive to both the character accuracy and the length of the field. As a rule of thumb, if OCR accuracy is less than 95%, or if the field is longer than about 20 characters, you are frequently better off keying the field by hand.

Use database lookups to fill in data fields

If your capture software permits it, sometimes you can perform a database lookup to fill in a data field. For example, you might take a last name and a social security number, feed it into a customer master database, and retrieve the customer's first name directly from the database.

This is an especially good technique if the data fields can be used to check each other. In the example above, if one of the first two fields was mis-keyed, the database would reject the query since the name would not match the social security number. This provides a double benefit: the database is checking the accuracy of the first two fields *and* it's filling in the third field automatically.

Reducing prep time #1: Use automatic form ID

In order to use automatic recognition techniques such as OCR, different document types must be defined ahead of time so that the system knows where to look to extract the required data. This means that different document types must be scanned separately, and this in turn requires time consuming sorting of documents into separate batches during the preparation step.

Automatic form ID is an advanced technique in which the system "learns" different form types ahead of time and then automatically recognizes them during scanning. The system can then sort the scanned images electronically and process each document based on its predefined characteristics. Manual sorting of document types is completely eliminated.

Reducing prep time #2: Use auto-contrast adjustment

Form ID helps to reduce prep time, but there are other factors that may force you to sort documents into separate batches anyway. One of the most common problems is that different documents are printed on different types of paper. Scanner settings that work on white paper, for example, will not work with documents printed on light blue paper. In this case, automatic form ID does not eliminate the need for document sorting since the two different document types need to be scanned with different scanner settings.

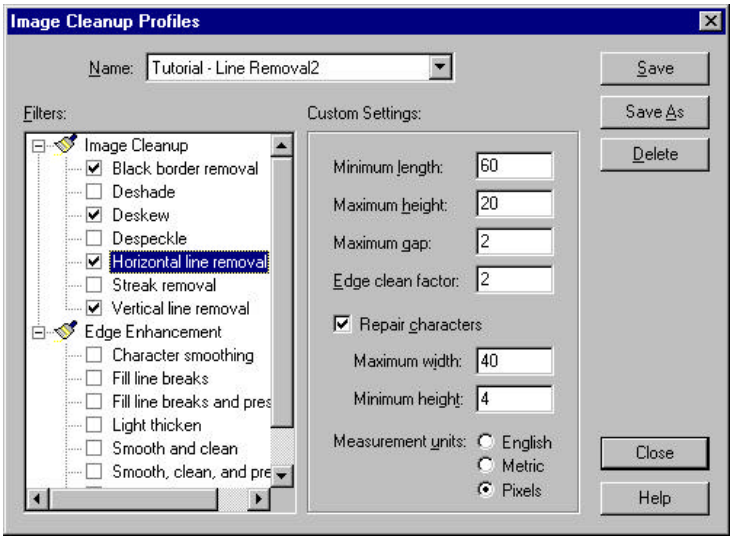
To resolve this, use a scanner that incorporates VRS (VirtualReScan) technology. VRS-equipped scanners perform automatic contrast adjustments on each page in a batch and provide nearly perfect images regardless of the type or color of paper used for each document type. Furthermore, for documents that fall below a pre-defined quality threshold, an intelligent monitoring agent stops the scanner and allows the user to adjust the settings manually. The scanner then continues with the rest of the batch normally.

VRS produces perfect scanned images on nearly any document type, and automatic form ID automates the processing of different document types. The two of them together eliminate virtually all sorting associated with document prep.

Use image cleanup to make images more readable and increase OCR/ICR accuracy

There are several techniques that can make images more readable and increase OCR and ICR accuracy. The most effective ones include:

- *Deskewing.* This technique straightens pages that have been scanned slightly crooked due to mechanical tolerances in the scanner's document feeder. Deskewing can increase the accuracy of OCR by 5-10% or more, which, as we saw above, can be enough of an improvement to make OCR cost effective compared to manual keying.



Ascent Capture supports a wide variety of image cleanup options that make images more readable and increase recognition accuracy.

- *Deshading.* OCR engines are unable to read words against the gray shaded backgrounds that are common on forms. Removing shading allows you to OCR zones that are otherwise unreadable.

- *Despeckling and streak removal.* These techniques remove small speckles and streaks caused by dirt in the scanner feeder or noise in the scanner CCDs.

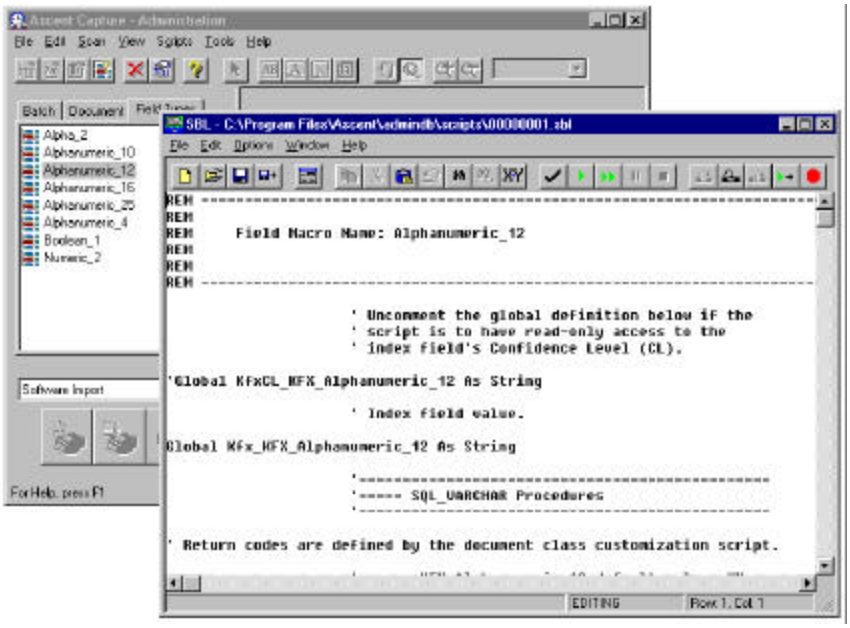
- *Line removal.* On typewritten forms, words are frequently typed so that they cross over the lines on the form, which makes them unreadable to OCR and ICR. Line removal erases the lines on the image and then reconstructs the characters so they can be recognized.

- *Edge enhancement.* This is actually a multiple set of filters that sharpens the edges of characters. The results are usually invisible to the eye, but they can increase the accuracy of OCR and ICR by as much as 5-10%.

Overall, by applying the proper cleanup functions for different document types, you can increase OCR and ICR accuracy by anywhere from 10-30%. This can easily make the difference between using automatic recognition profitably and being forced to hand key every field.

Use validation scripts to reduce manual checking

If the accuracy of a field is especially important, a technique known as *double key entry* is frequently used: the document is indexed (or validated) separately by two operators, and the results are compared. If they don't match, the system displays an error. However, although this technique is extremely accurate and has been used for years in the data entry field, it's also expensive since it doubles the amount of manual keying. Double key entry is usually used only for one or two critical fields.



Ascent Capture includes a scripting language similar to Visual Basic that provides a powerful, but easy to learn, data validation capability.

Another way to increase field accuracy is to use automated scripts instead of (or in addition to) double key entry. A flexible scripting language allows you to perform anything from a simple accuracy check to a sophisticated database lookup. For example:

- A simple validation might check to make sure that a telephone number consists of all digits and is the correct length.
- A more complex validation might compare a city and a ZIP code in a post office database to make sure they match.
- A third type of validation script might display a list so the operator clicks on an entry instead of typing it. Not only is this faster than typing, but it is more accurate as well.

In most cases, validation scripts are not as foolproof as double key entry, but they are often a good substitute, especially if there are several other fields to act as backups. They are also useful tools for increasing the accuracy of fields that are not quite important enough to warrant the additional cost of double key entry.

Use bar codes to automate indexing

Bar codes are far and away the best way to automate the extraction of data from business documents. A good bar code reading package can read multiple bar codes on a page, at any angle on the page, with an accuracy of 99.5%+. What's more, since bar codes have built in error checking, there's no need to have operators check the accuracy of the data. The capture software can do it for you.

Of course, not all business processes lend themselves to bar coding. However, you should consider bar codes if at all possible. They have proven themselves over time to be one of the fastest, most accurate, and most fault-tolerant forms of automatic data recognition.

Use OMR for bubbles and checkboxes

OMR (optical mark recognition) is a technique for automatically reading items on a form that are either selected or not selected. Example include bubbles on multiple-choice test forms, checkboxes on credit card applications, and circled numbers on reader response cards.

Forms often contain dozens or even hundred of these types of fields and keying them manually is extremely time consuming and error prone. OMR is a highly reliable method of automating this process, and good OMR engines can recognize a wide variety of filled in marks.

Use a separate station for rescanning

If you scan a large number of documents, you should consider a separate rescanning station for two reasons:

- It's expensive and disruptive to interrupt the operator of a high-speed scanner. A production scanner runs at 40-100 ppm, and that scanner is idle while its operator spends 30-60 seconds searching for an original document and then another 30 seconds feeding the single page. This is wasteful for a scanner station that could have processed an entire batch of documents in the same time.

- Production stations frequently have only automatic document feeders. Rescan stations usually require flatbed capability so that poor quality documents can be rescanned with greater precision.

As a rule of thumb, perhaps .5% - 1% of all documents have to be rescanned, and each document can take as much as 2-4 minutes to rescan. Why? Consider the steps it takes to manually rescan a page:

- The QC or index operator must reject the page and write a note explaining the problem.

- The scan operator must shuffle through the batch looking for the bad page.

- The page must be rescanned.

- The scan operator must insert the new image into the batch. If the software doesn't allow this, the entire batch must be rescanned.

- The index operator must page through the batch to find the bad page and then re-index it.

To avoid this, be sure that your capture software has an integrated rescan queue. This allows notes to be written electronically and pages to be inserted back into batches and re-indexed automatically. With the proper automation, rescanning time can be cut down to 1-2 minutes per page, a significant savings if your scanning volume is high.

Set up indexing and validation for maximum keying speed

Many fields cannot be automated using OCR, bar codes, or database lookups. For these manually keyed fields, make sure that the data entry screen is laid out for maximum keying speed. For example, do you want to display the entire image on the screen, or just the zone to be indexed? Should the data entry fields be on the left or the right? Top or bottom? Should they all be on the screen at once or should they show up one at a time?

There are no answers to these questions that apply to all cases. In fact, not only will different document types sometimes demand different treatment, but different keyboard operators will be faster and more comfortable with different setups.

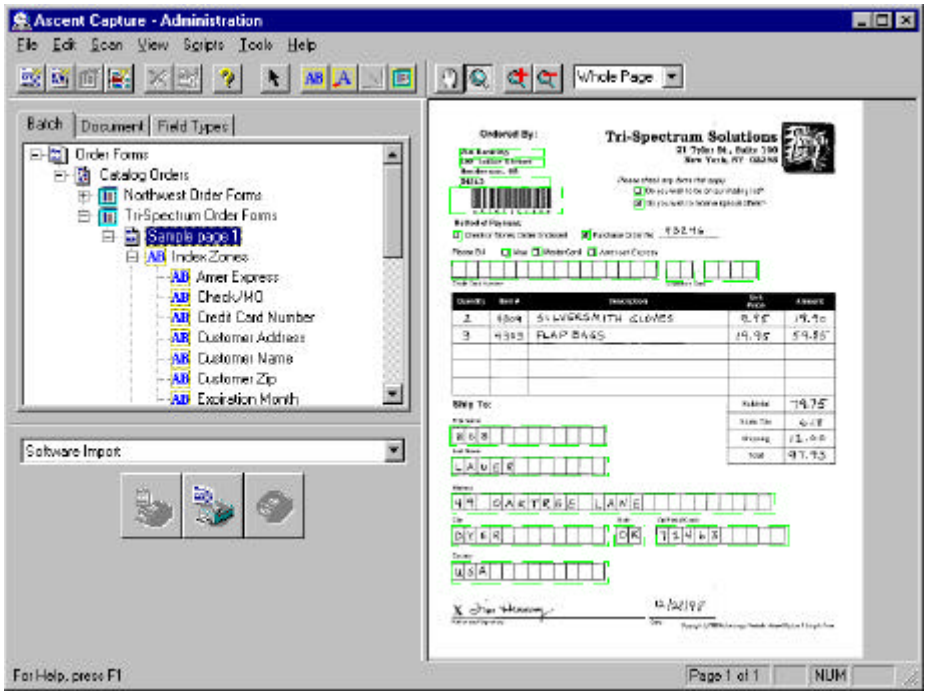
The indexing screen in Ascent Capture allows the operator to adjust the size and placement of both the image window and the data entry window.

Your capture software should be able to handle a wide variety of screen layouts to accommodate different tastes. The payoff can be surprising. Increasing your average keying rate from, say, 9,000 characters per hour to 10,000, reduces total keying time by 10%. If you spend a total of $50,000 on keyboard operators annually, this works out to a $5,000 reduction each year.

Define document characteristics ahead of time

Scanner operators should not have to set up scanner characteristics (resolution, density, contrast, etc.) for each batch. Instead, the capture software should allow an administrator to predefine different classes of documents so that the scanner operator merely picks from a list when a batch is loaded into the scanner.

This technique reduces batch overhead from as much as 1-2 minutes to 30 seconds or less and also helps reduce scanning errors. Instead of remembering different combinations of scanner settings, the operator can simply pick from descriptive names such as "Purchase Requisitions -- Parts Department."



In Ascent Capture, the administrator predefines batch classes, which contain detailed information about how to scan, index, and process different types of documents. The scan operator merely has to pick the correct batch class from a list and then begin scanning the batch.

Use the right batch size to speed rescanning

Every time you put a batch in the scanner there is overhead associated with putting the paper in the hopper and starting up the scanner. This tempts people to make their batches as large as possible in order to reduce the batch overhead.

However, there's a downside to this. If a page turns out to be unreadable and needs to be rescanned, the scanner operator has to find the page within the original batch. If a batch is 100 pages long, this is not too difficult. If it's 500 pages long, the search could take minutes. For maximum overall throughput, you're best off picking a moderate batch size that minimizes batch overhead but also minimizes search time for rescanned documents. A batch size of 100-150 pages is usually optimal.

Use remote scan stations instead of shipping documents to a central site

All too often, production scanning is done at a single central site even if the paper itself originates at multiple remote locations. This creates two problems: 1) scanning is delayed since it takes at least a day for the paper to get shipped to the central scanning site, and 2) the cost of capture is increased since you have to pay to ship batches of paper to headquarters every day (and possibly ship them back to the remote site when scanning is finished).

Unless there are special circumstances (such as legal requirements that require tightly monitored scanning procedures), it is usually cheaper and faster to scan documents at each remote location. You should choose software that allows scanning and indexing to be done either at a central site or at a remote site and that allows finished batches of scanned documents to be transmitted to the central capture site over inexpensive Internet connections (typically via FTP downloads). Remote sites that scan small numbers of documents can usually be set up quite inexpensively using low-speed scanners, low-end capture software, and low-cost dial-up Internet connections. It is not unreasonable to set up a small remote site for less than $2000, a sum that usually pays for itself within a few months by eliminating daily shipping charges.

Reduce the number of index fields (document capture only)

This technique applies only to document capture, not data capture, in which the purpose of the data fields is to act as indexes that allow you to retrieve the document later. There are two primary considerations for deciding how many index fields you need:

- You must have enough index fields to provide redundancy in case a field is mis-keyed. In a document imaging system with millions of documents, the index is the only way to find a document, so a bad index means that the document is lost forever.

- On the other hand, you should reduce the number of index fields to save cost. If an index field takes 4 seconds to type, for example, and you process 10,000 documents per day, that amounts to 40,000 seconds per day, or $112 (at our usual rate of .28¢ per second). Eliminating one index field can save $29,000 over the course of a year! In addition, reducing the number of indexes keeps your database smaller, which speeds up document retrieval times.

Most records management experts recommend at least three index fields per document to guarantee that the document can be retrieved. In some cases, if you use automated techniques to guarantee the accuracy of the indexes, you might be able to get away with less, but three index fields is usually a good rule of thumb. It's enough to guarantee accuracy but not so many as to add unnecessary cost to your capture and retrieval process.

Reduce the length of manually keyed fields

Can you use shorter fields? If you reduce the average length of a manually keyed field from, say, 10 characters to 8 characters, you've shaved .72 seconds per field (assuming a keying rate of 10,000 characters per hour). If you process 10,000 documents per day, that's over $5,000 per year.

Adding It All Up

A simple comparison shows the potential savings of applying these cost saving methods to a typical capture site. Consider the cost of a batch capture system with the following characteristics:

- 10,000 documents per day
- 500-page batches (20 batches per day)
- 40 ppm scanner
- 4 data fields per document, averaging 10 characters each
- 1% rescanning rate
- Manual keying rate of 9,000 characters per hour

The table below compares costs for System 1, which uses little automation, to System 2, which automates the capture process significantly using features discussed in this white paper.

Table 2. An Example of Labor Savings via Automation of Production Capture

Step	System 1	System 2	Comments
Document prep	5 seconds/page *Total: 50,000 seconds/day*	3 seconds/page *Total: 30,000 seconds/day*	Automated via form ID and VRS preprocessing.
Scanner load time	20 batches per day, 30 seconds per batch *Total: 600 seconds/day*	100 batches per day, 30 seconds per batch *Total: 3000 seconds/day*	In System 2, batch size has been reduced to 100 pages, so the number of batches has increased to 100 per day.
Scanner setup time	30 seconds/batch *Total: 600 seconds/day*	None	Predefined batch classes eliminate manual setup.
Batch scan time	750 seconds/batch *Total: 15,000 seconds/day*	150 seconds/batch *Total: 15,000 seconds/day*	
Field 1	10 characters @ 9,000 characters/hour = 4.0 seconds. Add .5 seconds overhead for total of 4.5 seconds per field *Total: 45,000 seconds/day*	8 characters @ 10,000 characters/hour = 2.88 seconds. Add .5 seconds overhead for total of 3.38 seconds per field *Total: 33,800 seconds/day*	This field is still manually keyed, but the average length of the field has been decreased and the keying rate has been increased.
Field 2	Same as Field 1 *Total: 45,000 seconds/day*	OCR field. 2 seconds to check plus .4 seconds to correct (average). Add .5 seconds overhead for total of 2.9 seconds per field. *Total: 29,000 seconds/day*	Using OCR, along with image cleanup to increase OCR accuracy to 99% (90% field accuracy) saves about 1.6 seconds per document compared to manual keying.
Field 3	Same as Field 1 *Total: 45,000 seconds/day*	50 incorrect bar codes per day (99.5% accuracy). 4.5 seconds to correct bad fields. *Total: 225 seconds/day*	Bar codes eliminate checking time (since they are self-checking) and practically eliminate manual keying time for this field.
Field 4	Same as Field 1 *Total: 45,000 seconds/day*	None	This field has been eliminated through careful analysis of retrieval patterns.
Verify Index 1	Same as initial keying. *Total: 45,000 seconds/day*	None	Verification is done via database lookup and is completely unattended.
Rescan	100 rescans per day @ 4 minutes per page. *Total: 24,000 seconds/day*	100 rescans per day @ 2 minutes per page. *Total: 12,000 seconds/day*	Rescanning time has been cut significantly because the smaller batch size reduces the time it takes to search through batches for bad pages.
Total seconds per day	315,200 seconds	123,025 seconds	
Total hours per day	87.5 hours	34.1 hours	
Total hours per year	22,764 hours	8,866 hours	
Total labor cost per year @ burdened rate of $10/hour	**$227,640 per year**	**$88,660 per year**	**Annual savings: $138,980**