Dollars and Sense:

Evaluating Remittance Processing Software

By: L.H. "Skip" Kaiser and David A. Schmidt

Abstract

Most new releases of accounts receivable software include automated remittance batch processing (auto-remit) capabilities. Auto-remit invites you to import cash receipts as an electronic file, rather than manually entering them on-line. The development of this electronic file is usually outsourced to your EDI partner or lockbox bank.

The prospect of acquiring auto-remit capabilities is seldom a primary reason for installing new financial software. When accounts receivable software modules are being evaluated, auto-remit capabilities are often just another "add-on" or additional functionality that comes with the package. Not surprisingly, the attitude toward this feature often is, "now that we have it, let's try it." Thus, the question of whether the benefits from using the A/R software's auto-remit function justify the expense of developing an electronic cash receipts file is a secondary consideration.

When the question is finally addressed, in most cases the answer will be yes. However, the economic incentive is usually small because few customers provide the correct invoice numbers with their payments or pay exactly the amount owed. Though A/R software can identify invoice numbers being paid when there is an exact match with the remittance data and determine whether a cash discount is for the correct amount and within terms, over 75 percent of cash receipts may still end up in the re-work file pending manual application. A/R software simply does not have the capability to drill down into both the remittance advice and the A/R open items database to find any matches that are not obvious.

This is where remittance processing software earns its keep. Rather than relegate unmatched cash receipts to your A/R's re-work file, remittance processing software uses a customized battery of sophisticated matching algorithms to identify exactly what the customer is paying and then automatically process any needed credit or debit memos to finish the job. Without detailed system knowledge, it is difficult to recognize all the manual tasks an auto-remit application must replace in order to maximize cost saving and efficiency benefits. By analyzing the tasks performed during the cash application process, the underlying dynamics that enable remittance processing software to match over 90% of cash receipts with the unpaid items in your A/R each and every day can be fully understood.

Manual Tasks

The tasks involved in applying customer payments to a business' open receivables involve three general areas: matching payments to customers and invoices, determining what to do if the remittance differs from the receivables, and properly coding all exceptions.

The individual tasks are:

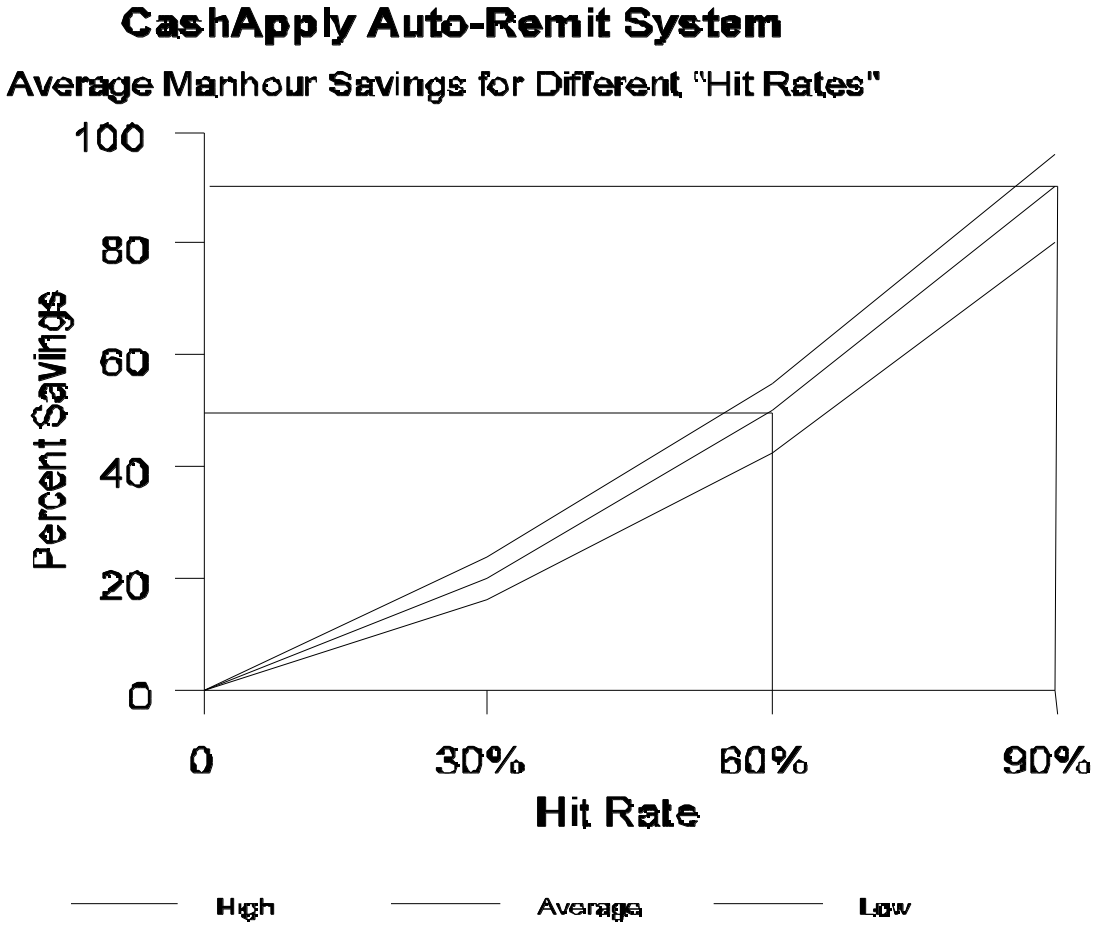
1. Identifying the payee and invoice
2. Reconciling payments to receivables if remittance details are incomplete
3. Determining if prompt pay discounts were timely
4. Deciding whether to charge a customer back for underpayments
5. Preparing debit memos for unauthorized deductions and affixing reason codes
6. Preparing credit memos for authorized promotional adjustments or payment deductions
7. Preparing credit memos for overpayments and payment on account

In an automated environment, remittance detail from electronic cash receipts and lockbox capture are matched to a database of unpaid receivables to ensure cash is applied accurately. The same seven steps that comprise the manual process must also be incorporated into the automated algorithm. The problem some companies run into stems from relying upon the auto-remit capabilities of their A/R software, or worse yet having to write a complicated interface program, and then only being able to match around 60 to 65 percent of the remittances to their A/R. Buying automated remittance processing software, with hit rates exceeding 90 percent, will be much more economical in the long run.

The Higher the Hit Rate the Better

Hit rate measures the percent of remittance detail items that are posted automatically, without operator input, into the A/R. A system that achieves a high hit rate is therefore posting a high percentage of customer cash receipts into the A/R without much manual intervention.

Typically, most cash receipts include some data that exactly matches open receivables without requiring any further processing beyond the first task listed above. The remaining remittances, which are progressively more difficult to match, involve one or more of the other six tasks. What this means is that an A/R software module yielding a 65% hit rate has merely skimmed the cream off the top and left the hardest tasks, all involving data correction, to be processed manually. The bulk of the remittances can be processed quickly, whether that is done manually or automatically. While there is a benefit from automating repetitive tasks, greater benefits derive when automation assumes the processing of more difficult and time consuming chores.



Furthermore, the higher the hit rate from an auto-remit solution, the greater its marginal utility. The accompanying graph shows the wide difference between the hit rate and the labor saved from manual cash applications. Whereas a 60% hit rate yields roughly a 50% reduction in labor, a 90% hit rate can achieve a 90% decrease in labor. In other words, only 50% more matches reduce labor an additional 80%. Clearly, when automating remittance processing, a prime consideration should be the ability of the auto-remit software to achieve a high hit rate.

The Source of the Savings from a High Hit Rate

In any auto-remit application, outsourcing data acquisition (keying remittance details) provides a 35% savings right off the top. To get to a 90% hit rate, you must also employ

complex matching algorithms to determine the correct invoice numbers and automatically create debit and credit memos for small dollar write-offs. Without sophisticated solutions to these two phases of the data correction process, you

will not realize a substantial reduction in the manpower required for cash application. Roughly apportioned, complex matching algorithms contribute about 25% of the efficiency found in a 90% hit rate environment, while the final 30% comes from automatically creating debit and credit memos.

The level of manpower reduction between the auto-remit capability of A/R software yielding typical results and that of a high end solution is very significant. The ability of auto-remit software to find a match, and if an exception is identified, to quickly, easily and effectively deal with it, is critical. Software that achieves only a 65% hit rate derives most of its benefits from outsourcing data acquisition and applying matching algorithms, not its automation features. Using the accompanying "Check list of Time Saving Components that Should Be Included in Auto-remit Software," to identify the strengths and weaknesses of your current or prospective A/R software will help you understand why your hit rate is probably less than 90%. Of course, 100% cash application manpower savings are unlikely because uncorrected cash receipts need to be resolved, deduction coding may require operator attention, and judgement calls must be made in order to match incomplete remittance data with open receivables.

Auto-remit Solutions within the Context of Financial EDI

As the number of electronic transactions grows, it becomes both tactically essential and economically justifiable to implement systems capable of handling a broad range of electronic cash receipts and remittance details. The accompanying diagram (Optimal Cash Receipt Processing System) shows how a company can receive cash by check through its bank's lockbox, by ACH, by Fedwire, by VANs, and by VABs. Additionally, the capability of receiving Financial EDI data directly from customers provides the flexibility to separately receive remittance detail that lists invoices, credits and other pertinent information from the actual payment. This allows EDI remittance data to be pre-processed in anticipation of the receipt of the physical check.

Financial EDI has quietly carved out a niche among those who recognize the significantly improved cash flow that results from EDI based remittance advice and electronic funds transfer, especially when that data flows cleanly into the A/R system. The most substantial benefits from Financial EDI result from reductions in Days Sales Outstanding (DSO), interest expense, and in labor. The reduction in DSO (typically 10%) and interest results from the shorter collection cycle realized with EDI transactions (customers pay on-time according to their written EDI agreement) translating into more cash and lower receivable balances. In addition, more and more companies are becoming EDI capable, adding pressure to those who have not yet adopted this technology.

However, to get maximum benefits from EDI requires restructuring the fundamental work flow for applying cash. Some companies that have been forced to accept EDI transactions from their trading partners contracted with a VAN, hook up a PC running EDI translation software and then print out any EDI

transactions for manual re-keying into their enterprise systems. This clearly defeats the intent and negates the benefits of EDI.

Case Study: Financial Software's Auto-Remit Capabilities Fall Short

When In-Sink-Erator (ISE), Racine, WI, replaced their legacy mainframe software with J.D. Edwards, there was some hope this package's auto-remit feature would be powerful enough to also replace ISE's stand-alone remittance processing software. However, early on in the evaluation process Paul Clausen, ISE's credit manager, realized that the J.D. Edwards accounts receivable module "didn't provide enough functionality." Clausen claims:

> It lacked extensive matching algorithms
>
> There was no way to handle items that could not be matched, except to send them to rework
>
> It was inflexible in regard to grace periods associated with prompt payment discounts

Because of these three critical factors, ISE chose to continue using their stand-alone remittance processing software, CashApply, and interface it with the J.D. Edwards A/R module. It has been a good decision, too. According to Clausen, the software is able to at least identify the customer account number for all but 5 percent of all remittances. As a result, manual support of the automated remittance process typically requires less than _ hour each day for a function that required a full time cash applier when remittance processing was handled manually.

Of course, this is an unusual case. ISE had already automated remittance processing, using CashApply and their bank's lockbox department to replace a manual cash posting function. In a sense, Clausen's staff had already seen the two extremes, and found that any departure from a highly efficient remittance processing environment was an unacceptable step backward that would put an undue burden on staff resources. Nor would the negative effects have been limited to the amount of labor and time required to post cash. Clausen's collectors would have lost a day or more visibility as to payments received. Either they would have had to revert to the practice of manually noting payments on accounts or risk calling accounts that had already paid, both activities counterproductive to effective collection processes.

Operational Efficiencies: A By-Product of Auto-Remit Processing

When utilizing software to replace any manual system, careful consideration should be given to other benefits and costs. In the case of auto-remit software, a number of questions demand answers:

1. Can payment history be compiled and exception reporting provided?

2. How much operator training is required and is the software intuitive to use?

3. Are electronic interfaces with your lockbox bank, EDI partner and A/R built into the software or is additional programing required?

4. How many high cost computing resources are required to run the software?

With this in mind, auto-remit software needs to effectively support the following operational aids:

Payment History - When customers ask questions about how their cash payments were applied against their account or if deduction analysis is necessary, comprehensive research tools save time and effort. In addition, the ability to drill down into the payment history database to identify changes in payment timeliness and customer abuses, improves future collections

Ease of Use - A good on-line tutorial and easily understood set-up controls to describe your cash application parameters (such as discounts and grace periods) contribute to user productivity.

When these operational aids are combined with sophisticated matching algorithms, manual intervention is minimized. Additional efficiencies are then achieved on the processing side through:

The Electronic Interface - The transmission of data from the bank or directly from your customers via EDI or the Internet must be seamless and transparent. By the same token, auto-remit software must produce a file that can be accepted by the A/R system.

Minimal CPU Cycles - Matching algorithms can eat up CPU time on a mainframe or in a client/server environment, so it is important the auto-remit software use minimal CPU cycles. Software that processes these algorithms off-line has a natural advantage in this area.

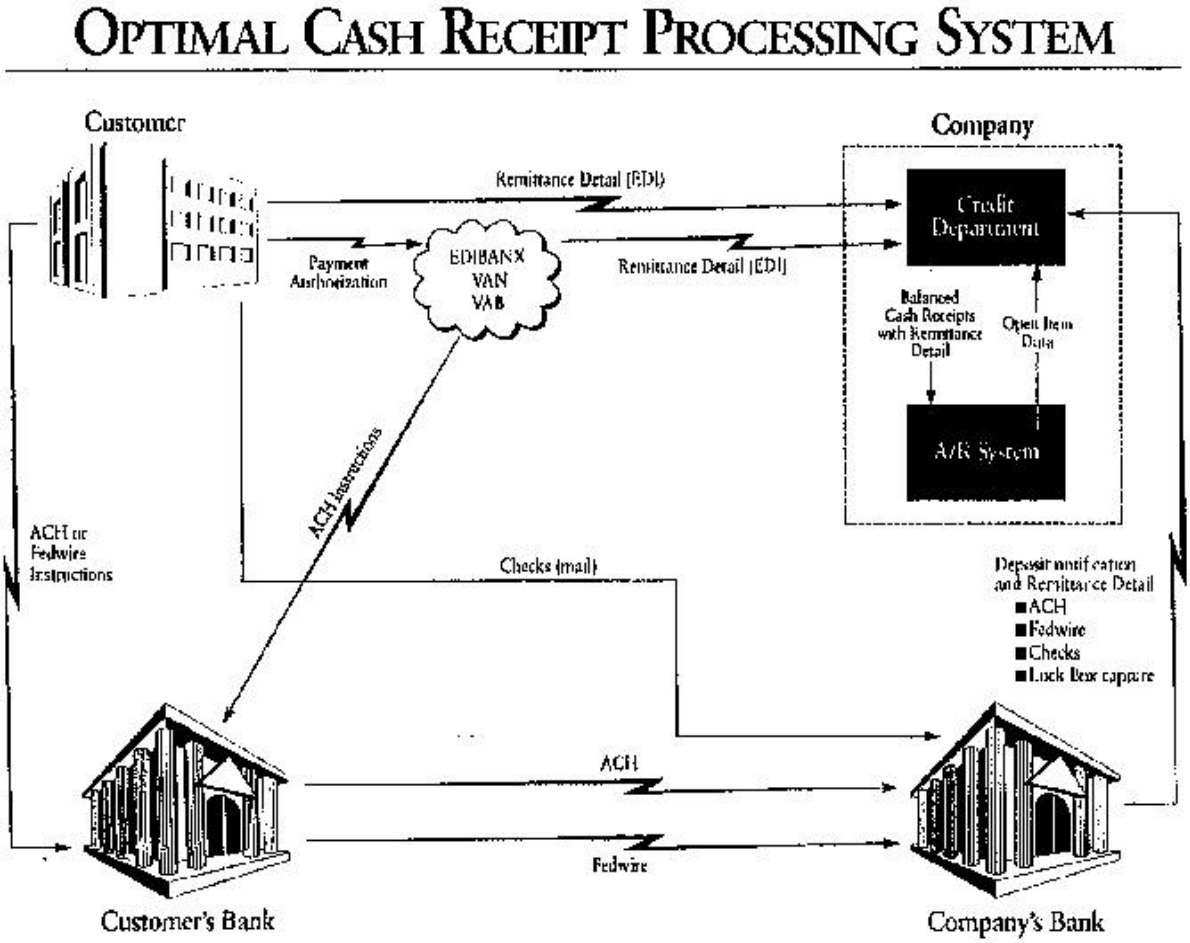
The typical A/R software module that can also handle auto-remit applications does not contain all these features. Though the auto-remit capabilities of A/R software do provide some incremental benefits, dramatic improvements such as those realized by In-Sink-Erator are unlikely. As a result, achieving significant benefits may prove elusive, especially in a high deduction volume environment, which by nature will tax a company's clerical resources.

Justifying an Auto-remit Solution

Fortunately, it is not as difficult to justify restructuring or to actually automate the cash application process as it might appear. Unlike the other areas of EDI, Financial EDI can be implemented economically by itself. It is important to avoid the trap of equating the automation of cash application tasks with reengineering, a more extensive process that changes the way tasks are performed. By itself, EDI simply allows a company to apply cash receipts and

the accompanying remittance detail electronically, thereby eliminating manual processing.

Restructuring the cash application process with the aid of relatively inexpensive computer software is not only possible, but necessary, because the handling of exceptions associated with the correction and verification of electronically received customer payments must also be automated. Since the key to selecting the right remittance processing software, is the solution that best



minimizes manual operations, the most practical, cost effective solution is fully functional, stand-alone remittance processing software.

Time Saving Components That Should Be Included In Auto-remit Software
Data Acquisition
Electronic Files created by bank from lockbox deposits, ACH payments, and Wire transfers
BAI format accepted with no interface required
EDI 820 remittance advise accepted with no interface required
Warehousing of EDI 820 data for subsequent matching to cash receipt
Translation of customer deduction codes to your company's codes
Customer data such as MICR numbers and tolerances maintained in an electronic file
Data Correction
Debit memos created for unauthorized deductions/short payments
Credit memos created for overpayments and payments on account
Cash discounts verified and variations within tolerance written off
Matching algorithms to address your customer's data correction requirements
Automatic matching of credits with corresponding debits
Online correction of cash receipt data
Data Transmission
Create export file for A/R processing with no interface required
Add general ledger coding and replace reserved characters as required by A/R
Research and Analysis
Any check or paid invoice/memo can be viewed for at least 12 months after the deposit date
Ability to monitor small balance write-offs and tolerance amounts by customer
Trend analysis and exception reporting of late payments by customer

Skip Kaiser is the President of Accounts Receivable Processing Company, Omaha, NE, the developers of CashApply auto-remit software. He can be reached at 402-392-1768 or goarp@hotmail.com. Dave Schmidt is Principal of A^2 Resources, Yardley, PA, commercial credit consultants specializing in cost reduction and systems improvement. He is the editor of The Credit Manager, a monthly newsletter, and a contributing editor with Business Finance, formerly

known as Controller Magazine. He is also co-author of “Power Collecting: Automation for Effective Asset Management” (John Wiley & Sons, 1998). He can be reached at 215-321-5444 or 75564.3253@compuserve.com.

Printed in the United States of America
Credit Research Foundation
8840 Columbia 100 Parkway
Columbia MD 21045
410-740-5499