

The Business Case for Automating Payments Processing and Cash Application:

Building a Realistic ROI Model and a Winning Business Case

✧ *Develop a business case for justifying investments in an automated cash application (or other) system by focusing on (1) the ways to identify and quantify the benefits from automating cash application, (2) how to calculate the financial return, and (3) how to prepare a business case to gain executive approval.*

The
Credit
Research
Foundation

An Occasional Paper

Executive Summary

Those responsible for incoming payments processing and cash application are acutely aware of the need for automated systems to reduce or eliminate the time required by the numerous manual tasks in the end-to-end accounts receivable (A/R) process. Freeing up resources tied to cash application is one of the single biggest opportunities to enable improvements throughout accounts receivable. Accounts receivable often makes up 40% of the assets on a company's balance sheet, and effective management is critical. Automating as much of receivables management as possible is a proven method to achieve savings, improve days sales outstanding (DSO), and maximize operational efficiencies. Cash application is a process, that due to its low value-add and manual nature, is a prime candidate for automation through technology.

A key challenge in technology purchases is gaining approval from upper level executives who require formal analysis and justification. As a result, it's necessary to have a strong sales mindset when A/R funding requests compete against other projects for capital and operating budget dollars. In many companies, IT projects must be justified through formal processes that detail the spending requirements and estimated savings to produce a return on investment (ROI) measurement. Justifying a project that does not require capital expenditures, such as subscribing to a Software-as-a-Service (SaaS) solution, will often prove easier and less time-consuming for companies where such a solution is a fit.

This whitepaper helps develop a business case for justifying investments in an automated cash application system and will focus on (1) the ways to identify and quantify the benefits from automating cash application, (2) how to calculate the financial return, and (3) how to prepare a business case to gain executive approval. Throughout the paper are examples of actual improvements and efficiencies from payment processing automation efforts, as seen by accounts receivable departments across industries and sectors.

Introduction

CFOs are not always tuned into the issues mid-management must address in their daily functions. A CFO's perspective is focused on a corporate view, taking into account overall financial performance for the organization. Tips for an effective approach to developing a winning case for automated receivables management include:

- **Think like C-Level Management with a focus on bottom line results**
- **Present requests in strong business terms**
- **Recognize the competing requests for capital and resources**
- **Resist the temptation to base justification on the premise of "making life easier"**
- **Quantify technology benefits based on facts, not emotion**

Developing a Business Case

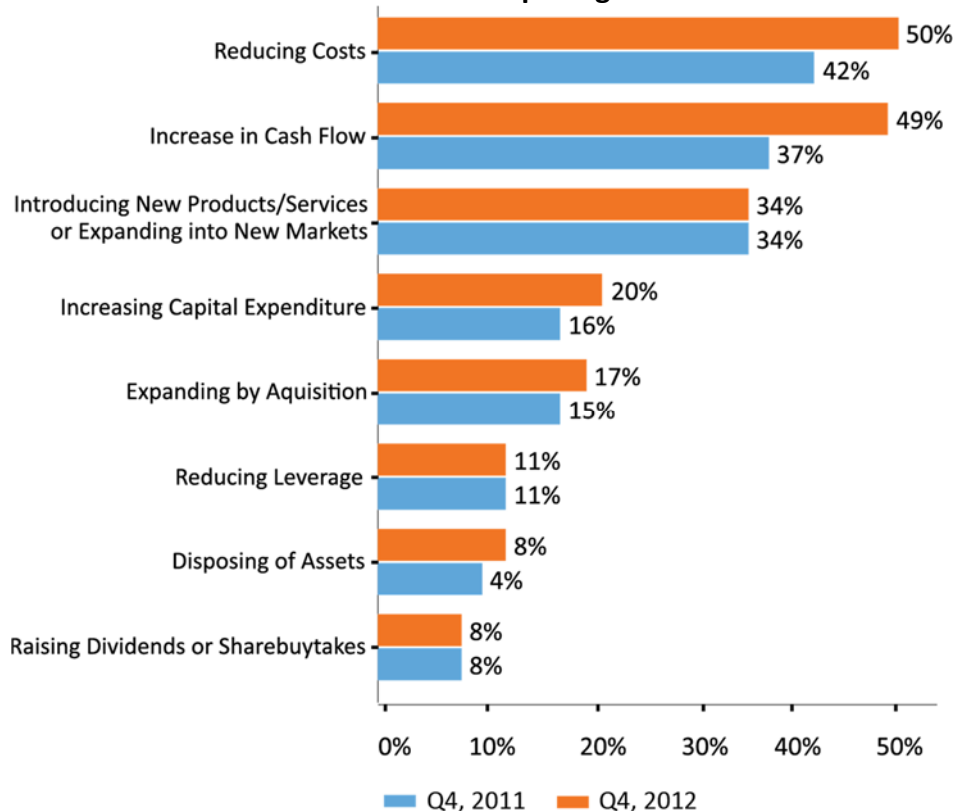
Executive management views successful projects as either making money, saving money or strengthening an existing initiative. Persuasive arguments for technology investments should be

framed in these terms. Unfortunately, departments often attempt to convince management to fund projects based upon statements such as these:

- **“We are stretched thin and so we need better systems to reduce the stress on our personnel.”**
- **“My team is frustrated because of repetitive, manual work.”**
- **“We can serve our customers better.”**

Each one of these “rationales” for bringing in an automated system carries with it an element of subjectivity which can be negated either through more objective thinking or a difference of opinion. For example, the complaint of being stretched thin will most likely be countered by a statement of “you need to do a better job of managing the workload across your department.” As such, the problem may be turned around to reflect poorly on your leadership. The frustration argument may lead to a discussion or a lecture about how personnel need to do a better job of controlling their emotions, or be more appreciative of the jobs they were hired to do. Using emotional approaches to build a case can turn the discussion away from the real issues that you want addressed. While customer service is certainly a noble goal and one that management values, unless you are able to back up your assertion of better customer service this point will be seen as vague. There is no universal measurement for valuing good customer service, therefore it must be defined within each organization. If there are such measurements at your firm, you must show in quantitative terms how you will meet or exceed those indicators.

Figure 1: The Deloitte CFO Survey Q4 2012 (current and past copies can be found at www.deloitte.co.uk/cfosurvey)
Metrics Executives Look for to Frame a Compelling Business Case



So just what are the quantitative metrics and measurements that executives are looking for to frame a compelling business case? To begin, look at the priorities of one very important decision maker, the CFO. CFOs are very concerned with how to “do more with less.” This translates into two important performance indicators: reducing costs and increasing cash flow. Figure 1 shows the results from a 2012 Deloitte CFO survey of their corporate priorities for the next 12 months. The survey looks at the changes from Q4, 2011 to Q4, 2012. Cost control is the leading priority for CFOs; 50% of respondents rated this as a strong priority. Increasing cash flow follows close behind with a 49% acknowledgement. This focus on defensive, rather than expansionary, strategies further underscores the need for a strong, quantitative business case for financial approval.

An important component of a winning business case is to set clear objectives. By establishing clear objectives, you are able to remain focused on what is important and communicate with your executives in a language they will understand. Setting up these clear objectives is centered upon understanding the value from the executives’ perspective. Instead of viewing this activity as “what is in it for you”, communication should be about “what is in it for the executives”. Three good questions that will establish clear objectives are:

- **What is the investment?**
- **What is the value?**
- **What is the payback?**

These three questions build upon each other. The first question establishes the costs of this project while the second outlines the benefits. For a cash application initiative, the benefits are usually best described in terms of savings – savings in terms of human resources full time equivalents (FTEs) that can then be refocused on higher value activities or in dollar terms.

The third question uses the costs and savings as inputs in a financial analysis to determine the ROI. ROI helps management understand, rationalize and prioritize purchase selections, and ensure these choices maximize payback and minimize risk. The three common measurements of an ROI analysis are net present value (NPV), internal rate of return (IRR), and payback period, which is the time required until the savings outweigh the investment costs. This payback period is often referred to as the breakeven point – the time when the savings become equal to the investment.

The Investment and Costs

It is important to understand the full costs incurred on a project to reach the desired business objectives or automation goals. We will discuss how to set these goals realistically in the next section. This includes not only the up-front costs but the ongoing costs to run the system. A cash application project, like any IT project, should be considered in terms of a number of components. Some common ones to consider:

- 1. Hardware** – do we need to buy, install, and maintain new hardware?
- 2. Software** – how much do I pay for the product and how is it spread over the evaluation horizon?
- 3. Vendor setup** – how much will it cost me for the vendor to implement the project with the characteristics necessary to achieve the expected automation targets?
- 4. Internal project resources** – how many resources do my team and the IT department have to allocate to the project; how many hours, over how long, and what will be the impact?
- 5. Ongoing system maintenance** – how will the system be maintained, updated, and managed and what will this cost?
- 6. New system management tasks** – what tasks will my team need to perform and how often to keep the system running as expected on top of their day-to-day business activities?

Software

Typical technology solutions follow License and Maintenance (L&M) or SaaS models. For L&M solutions, you should take into account not just the upfront license costs but also the maintenance costs over the entire period in which you are evaluating the project. For SaaS solutions, there is usually no up-front cost but periodic service charges are often assessed based on usage. It is important to understand all charges for which you are liable, and under what circumstances you incur them so that an accurate estimate of recurring fees over the evaluation horizon is included.

Vendor Setup

Setup costs usually cover any work that the vendor team needs to perform in order for your team to achieve your automation and savings objectives. This would cover installation, integration and configuration time as required. For some systems, custom work may be needed.

Internal Project Resources

It is easy to forget that you will have to dedicate resources, both from your team and from IT to this project. Depending on the integration requirements and system capabilities, more or less time will be needed. Traditionally, the opportunity costs of waiting on IT to staff is not included in the project analysis. However, the less dependency on IT resources, the more quickly and cheaply a project goes live.

Ongoing System Maintenance

For on-premise solutions, you will have to account for update and upgrade costs to take advantage of new capabilities and enhancements. You also need to understand how often updates are released, what capabilities they cover, and what it takes to apply them. Be aware that for some on-premise systems, such upgrades can be significant IT projects on their own and are associated with delays and system down-time. For SaaS systems, the vendor is responsible for any updates, and upgrades are usually seamless and subject to standard service level agreements (SLAs). The cost of SaaS updates is part of the recurring fees.

New System Management Tasks

This is a key area that is often missed in business cases. It takes into account the tasks that the cash application team did not have to perform prior to the installation, but will be required to

perform on a daily (or other periodic) basis with the new system. One regular example that should be considered is the transformation of scanned images. OCR templates will need to be managed for reading paper checks and electronic remittance if the system does not work with email or web portal remittance natively.

The Value – What Will Automation Save Me?

The second piece of the puzzle is identifying what will be saved or increased from implementing the system. Having realistic numbers for before and after is necessary to understand what the system will do for you and from where your savings and optimization will come.

The best practice is to follow an adjusted value stream analysis process:

- 1. Identify process tasks**
- 2. Break key tasks into sub-tasks**
- 3. Identify opportunities for improvement**
- 4. Model the potential benefits**

Identify Process Tasks

The set of high-level tasks for cash application processes will vary between companies. In general, they depend on a number of factors, such as the mix of customers and the way they pay (paper checks vs ACH/EFTs and credit cards), your relationship with your bank and whether you pay for lockbox service and key-in fees, and your ERP or accounting system capabilities.

For cash application, some key tasks can be grouped together based on the payment type and the source of the incoming remittance information. Consider the following remittance sources when generating a high-level task list:

- **Remittance information received through bank lockbox files**
- **Paper remittance received directly (not through lockbox files)**
- **Remittance received through EDI transactions**
- **Remittance received via email, either as part of the email body or enclosed in attachments**
- **Remittance information presented on customer websites and other 3rd party portals**

Break Key Tasks into Sub-Tasks

With the above remittance formats, a common list of associated cash application tasks has been generated for consideration:

Figure 2: Common List of Associated Cash Application Tasks

| Key Tasks | Example Sub-Tasks |
|---|--|
| Process payments and remittance received through bank lockbox services | <ol style="list-style-type: none"> 1. Prepare files, configuration to process with ERP's lockbox processing program 2. Resolve posting errors after initial processing through the lockbox program 3. Read the paper remittance images from the bank files 4. Cleanup remittance to identify correct invoice paid 5. Post payments on-invoice or on-account |
| Process paper remittance (not received through the bank lockbox file) | <ol style="list-style-type: none"> 1. Link the paper remittance to correct payment in the bank payment file 2. Read the paper remittance 3. Clean up remittance to identify correct invoice paid 4. Post payments on-invoice or on-account |
| Process EDI Remittance | <ol style="list-style-type: none"> 1. Prepare the EDI file to process through the EDI processing program or with manually managed excel workbooks |
| Process remittance received via e-mail and e-mail attachments | <ol style="list-style-type: none"> 1. Scan inbox and download remittance information 2. Read and capture the remittance data 3. Link the remittance to the correct payment in the bank payment file 4. Clean up remittance to identify correct invoice paid 5. Post payments on-invoice or on-account |
| Process remittance delivered through customer websites and portals | <ol style="list-style-type: none"> 1. Log-in and download the required remittance document 2. Read and capture the remittance data 3. Clean up remittance to identify correct invoice paid 4. Post payments on-invoice or on-account |
| Process payments without remittance | <ol style="list-style-type: none"> 1. Reach out to payers for clarification 2. Correct illegible remittance information |
| Create deductions | <ol style="list-style-type: none"> 1. Create deductions 2. Cross reference customer reason codes 3. Code deduction correctly with internal reason code |

Identify Opportunities for Improvement

For each of these and other tasks your team performs, review your current process and identify how many hours per week on average your teams spends on each task. This gives you the total FTEs your team spends on applying cash over a given period. This is also the amount of work that needs to be completed with the new system to continue functioning at same level of productivity.

To get to the savings, review how much time your team will spend on each task after the new system goes live in the To-Be process. Make sure to understand any additional, new tasks you will spend time on in the To-Be state. Also, hold your vendors responsible for achieving the operational efficiencies that they promised and that you will use to get the To-Be savings below.

Tasks usually introduced by most cash application systems that should be considered in the To-Be process costs are:

1. Management of templates – Templates are the way traditional OCR systems read data from paper remittance. The issues arise when different formats and more payer information is included on the remittance, and the exceptions that require analyst intervention grow, increasing the number of templates needed. The time spent looking through templates, cleaning up thousands of templates and creating templates on a daily basis should be included in the To-Be process tasks.

2. Converting electronic remittance to paper and fixing errors – Electronic remittance is a great tool and candidate for automation. The data is already high quality and digitized so the accuracy of reading and processing it may reach close to 100%. However, traditional OCR systems that do not process electronic remittance natively but instead rely on transforming it to images or printing and scanning it, introduce the possibility of error by employing a low-fidelity print format and the OCR engine. The work to handle these exceptions should be considered as well.

Model the Potential Benefits

A simple breakdown before and after is shown in the following table in Figure 3. Additionally, you can further break down tasks per the needs of your process.

Figure 3: Task/Activity Before and After Time Breakdown

| Activity Group / Activity | Status Quo (Hours / Week) | With Solution (Hours / Week) |
|---|------------------------------|---------------------------------|
| Payment and Remittance Through Bank Lockbox | | |
| Preparing file for the lockbox program | ? hours | ? hours |
| Resolving errors after initial processing | ? hours | ? hours |
| Reading paper remittance image from file | ? hours | ? hours |
| Check remittance data to identify right invoice | ? hours | ? hours |
| Posting on-account or on-invoice | ? hours | ? hours |
| Paper Remittance Not Received Through Lockbox | | |
| Linking remittance and payment | ? hours | ? hours |
| Reading the remittance | ? hours | ? hours |
| Check remittance data to identify right invoice | ? hours | ? hours |
| Posting on-account or on-invoice | ? hours | ? hours |
| EDI Remittance | | |
| Preparing EDI file for processing | ? hours | ? hours |
| E-Mail Remittance | | |
| Fetching remittance from e-mails, attachments | ? hours | ? hours |
| Reading the remittance | ? hours | ? hours |
| Linking the remittance to payment | ? hours | ? hours |
| Check remittance data to identify right invoice | ? hours | ? hours |
| Posting on-account or on-invoice | ? hours | ? hours |
| Website, Portal Remittance | | |
| Downloading remittance | ? hours | ? hours |
| Reading remittance | ? hours | ? hours |
| Check remittance data to identify right invoice | ? hours | ? hours |
| Posting on-account or on-invoice | ? hours | ? hours |
| Exception Handling (Missing Remittance, Illegible Remittance Information) | ? hours | ? hours |
| Deduction, Dispute Coding | ? hours | ? hours |
| Additional Tasks | | |
| Managing Templates | ? hours | ? hours |
| Resolving electronic remittance OCR processing errors and exceptions | ? hours | ? hours |
| Total | ? hours | ? hours |

This method will help you identify the current demands on your team in terms of FTEs or time spent to process cash. It will also show you the 'To-Be' demands or how much time your team will need to allocate in terms of FTEs to process the same payment volume with the system you are evaluating.

$$\text{Savings (hrs)} = \text{Hours before} - \text{Hours after}$$

Notice that if your team is spending more time to do their work with the system than without, this project does not make sense already.

The final step to calculate dollar savings is to multiply the hours saved per week before and after by the loaded labor costs:

$$\text{Savings (\$)} = (\text{Hours before} - \text{Hours after}) \times \text{Cost per hour}$$

Alternatively, you can use FTEs and the result will be the same:

$$\text{Savings (\$)} = (\text{FTEs before} - \text{FTEs after}) \times \text{Cost per FTE}$$

One additional step you may want to employ when evaluating savings is to consider multiple scenarios and levels of automation. For example, consider a conservative scenario of 30% no-touch automation of paper checks, typical of traditional systems. Include a best case based on vendor claims and a realistic scenario based on conversations with references and other factors.

The Payback - or Should We Do This at All?

The final step in developing the ROI model is summarizing the inputs across all expenses and savings, then the financial calculations are performed. The different expenses and savings usually considered for project ROI calculations are described below.

Capital Expenditure

This includes all costs incurred in the initial setup and the costs associated with the implementation project. These include costs for one-time software licensing, any other one-time software costs, as well as internal resource costs incurred towards the actual implementation. Many software solutions providers also charge a consulting fee for product implementation, which should also be reconciled under this section.

Recurring Annual Expenditure

All recurring fees incurred towards software maintenance and depreciation over the five year useful life are reconciled under this heading. If the software is delivered as a service (software-as-a-service) then the recurring expenses need to account for the annual subscription charged for the software. Software updates and upgrades, which require additional expenditure or internal resource allocation, should be added under this heading as well.

Net Profit Impact

This item accounts for the net operational savings achieved with the software solution being evaluated. It is possible that the solution might result in a net negative impact for operational costs, in which case the value would be considered as a negative. In addition to direct FTE-related operational savings, profit impact may also cover savings from reduction of bank lockbox fees. If an automation solution helps eliminate the data key-in costs – which often form a significant part of the bank lockbox expense statement – those savings should be reflected here.

Getting to the Math

Once the cash flows (net of costs and savings) are laid out by year, the Net Present Value (NPV) measurement for the ROI calculation is used, with a discount rate dependent on your business, to decide whether the project is financially attractive. Typical calculation parameters would differ depending on whether you calculate for a software-as-a-service or an on-premise solution. Examples for both have been provided below.

Figure 4: On-Premise Software – ROI & NPV Example

| A. Capital Expenditure | Year 1 | Year 2 | Year 3 | Year 4 | Year 5 |
|---|--------------------|-------------------|-------------------|-------------------|-------------------|
| Additional Hardware/ Servers | \$ 15,000 | | | | |
| Software Licenses | \$ 60,000 | | | | |
| Other Vendor Fees | \$ 7,000 | | | | |
| Consulting/ Implementation Fees | \$ 40,000 | | | | |
| Internal IT Resource Costs for Implementation | \$ 10,000 | | | | |
| Internal Business Resource Costs for Implementation | \$ 5,000 | | | | |
| Sub-Total | \$ 137,000 | \$ - | \$ - | \$ - | \$ - |
| B. Recurring Annual Expenditure | Year 1 | Year 2 | Year 3 | Year 4 | Year 5 |
| IT Maintenance Cost (On-Premise Software) | \$ 65,000 | \$ 65,000 | \$ 65,000 | \$ 65,000 | \$ 65,000 |
| Software Maintenance Fees | \$ 12,000 | \$ 12,000 | \$ 12,000 | \$ 12,000 | \$ 12,000 |
| Depreciation | \$ 15,000 | \$ 15,000 | \$ 15,000 | \$ 15,000 | \$ 15,000 |
| Sub-Total | \$ 92,000 | \$ 92,000 | \$ 92,000 | \$ 92,000 | \$ 92,000 |
| C. Net Profit Impact Estimate | Year 1 | Year 2 | Year 3 | Year 4 | Year 5 |
| Operational Expense Savings | | | | | |
| (less) Reduction in FTE Costs | \$ 60,000 | \$ 60,000 | \$ 60,000 | \$ 60,000 | \$ 60,000 |
| (Add) Costs for Additional Business Processes | \$ - | \$ - | \$ - | \$ - | \$ - |
| Total Operational Expense Savings | \$ 60,000 | \$ 60,000 | \$ 60,000 | \$ 60,000 | \$ 60,000 |
| Bank Fee Savings | \$ 40,000 | \$ 40,000 | \$ 40,000 | \$ 40,000 | \$ 40,000 |
| Sub-Total | \$ 100,000 | \$ 100,000 | \$ 100,000 | \$ 100,000 | \$ 100,000 |
| Cash Flow | Year 1 | Year 2 | Year 3 | Year 4 | Year 5 |
| Cash Flow Before Tax (C-B-A) | \$ (129,000) | \$ 8,000 | \$ 8,000 | \$ 8,000 | \$ 8,000 |
| Tax @ 40% | \$ 51,600 | \$ (3,200) | \$ (3,200) | \$ (3,200) | \$ (3,200) |
| After Tax | \$ (77,400) | \$ 4,800 | \$ 4,800 | \$ 4,800 | \$ 4,800 |
| Depreciation Add Back | - | \$ 15,000 | \$ 15,000 | \$ 15,000 | \$ 15,000 |
| Net Cash Flow | \$ (77,400) | \$ 4,800 | \$ 4,800 | \$ 4,800 | \$ 4,800 |
| NPV @ 8% | \$ (61,502) | | | | |
| Internal Rate of Return (IRR) | -39% | | | | |

Figure 5: Software-as-a-Service – ROI & NPV Example

| A. Capital Expenditure | Year 1 | Year 2 | Year 3 | Year 4 | Year 5 |
|---|-------------------------|-------------------|-------------------|-------------------|-------------------|
| Additional Hardware/Servers | Not required for SaaS | | | | |
| Software Licenses | Not required for SaaS | | | | |
| Other Vendor Fees | \$ 7,000 | | | | |
| Consulting/Implementation Fees | \$ 40,000 | | | | |
| Internal IT Resource Costs for Implementation | \$ 10,000 | | | | |
| Internal Business Resource Costs for Implementation | \$ 5,000 | | | | |
| Sub-Total | \$ 62,000 | \$ - | \$ - | \$ - | \$ - |
| B. Recurring Annual Expenditure | Year 1 | Year 2 | Year 3 | Year 4 | Year 5 |
| Subscription Fees (SaaS Software) | \$ 80,000 | \$ 80,000 | \$ 80,000 | \$ 80,000 | \$ 80,000 |
| IT Maintenance Costs (Internal) | \$ 3,000 | \$ 3,000 | \$ 3,000 | \$ 3,000 | \$ 3,000 |
| Depreciation | Not applicable for SaaS | | | | |
| Sub-Total | \$ 83,000 | \$ 83,000 | \$ 83,000 | \$ 83,000 | \$ 83,000 |
| C. Net Profit Impact Estimate | Year 1 | Year 2 | Year 3 | Year 4 | Year 5 |
| Operational Expense Savings | | | | | |
| (less) Reduction in FTE Costs | \$ 60,000 | \$ 60,000 | \$ 60,000 | \$ 60,000 | \$ 60,000 |
| (add) Costs for Additional Business Processes | \$ - | \$ - | \$ - | \$ - | \$ - |
| Total Operational Expense Savings | \$ 60,000 | \$ 60,000 | \$ 60,000 | \$ 60,000 | \$ 60,000 |
| Bank Fee Savings | \$ 40,000 | \$ 40,000 | \$ 40,000 | \$ 40,000 | \$ 40,000 |
| Sub-Total | \$ 100,000 | \$ 100,000 | \$ 100,000 | \$ 100,000 | \$ 100,000 |
| Cash Flow | Year 1 | Year 2 | Year 3 | Year 4 | Year 5 |
| Cash Flow Before Tax (C-B-A) | \$ (45,000) | \$ 17,000 | \$ 17,000 | \$ 17,000 | \$ 17,000 |
| Tax @ 40% | \$ 18,000 | \$ (6,800) | \$ (6,800) | \$ (6,800) | \$ (6,800) |
| After Tax | \$ (27,000) | \$ 10,200 | \$ 10,200 | \$ 10,200 | \$ 10,200 |
| Depreciation Add Back | Not applicable for SaaS | | | | |
| Net Cash Flow | \$ (27,000) | \$ 10,200 | \$ 10,200 | \$ 10,200 | \$ 10,200 |
| NPV @ 8% | \$ 6,784 | | | | |
| Internal Rate of Return (IRR) | 19% | | | | |

Notice that some of the elements above will not apply to all systems. We have tried to include some costs elements that our customers have come across. The exact nature of these will depend on the specifics of each solution.

Real World Example

The Coleman Company, Inc. is a worldwide leader in the outdoor recreation industry, developing products for a wide array of markets, including family and extreme camping, backyard recreation and hiking. After an internal evaluation of their receivables management processes, Coleman identified the need for more automation, especially in the areas of cash application and deductions research operations. Their challenge was not uncommon: they needed to improve error-prone cash application functions that were very time consuming. Customers used disparate forms of payment (check, ACH and wire) and remittance detail (paper, EDI, websites and email) resulting in a series of manual processes.

By automating their cash application function from end-to-end, Coleman was able to:

- **Achieve a quick payback on their initial investment**
- **Automate the process of gathering the data needed to auto-apply cash**
- **Leverage standard ERP cash posting**

A leading solution provider in the space was selected to provide their cloud-based Cash Application OnDemand solution. The solution retrieved data from all of Coleman's remittance sources - including paper checks, emails, EDIs, and customer portals - and pulled remittance in any and all formats, eliminating nearly all the manual work associated with this task. The Artificial Intelligence-powered data aggregation engine processed the remittance and extracted data without templates, further reducing the work associated with this task. The technology provided customer-specific rules for invoice matching, residual identification and coding, along with cash discount analysis. No human intervention was required. Finally, a syndication engine produced a single file specific to the ERP system (SAP) for ERP cash posting.

Summary

Some final suggestions for developing your business case include contacting other credit professionals in your network to learn how they have succeeded in gaining executive approval for technology projects. But don't stop there. Understand the budgeting cycles within your own company and be sure to meet any deadlines for capital and operating expense requests. A project which is included in an annual budget plan has a greater chance of implementation than one not budgeted. Additionally, learn from individuals in your own organization who work in other functional areas – sales, customer service, and supply chain – who compete for limited funds and what approach they use for submitting a successful business case.

Below are several of the key points to remember in developing your winning business case for receivables automation:

- **Put yourself into the shoes of your senior management**
- **Think like a CFO - Most CFOs are looking for hard ROI and the initiative should make or save money**
- **Look within your organization at how projects have been approved for funding in the past**
- **Understand the full cost of the solution – from setup and implementation to maintenance and the new system workflow**
- **Know where the savings are – review the current and To-Be process task by task to understand where your team is spending time now and where they will spend time in the future**
- **Create ROI models consisting of costs, savings, timing of cash flows, cost of capital, Net Present Value, Internal Rate of Return, and payback period**

About the Author

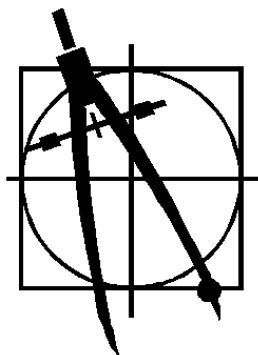
HighRadius provides software solutions to optimize receivables and payments functions such as credit, collections, cash application, deductions and eBilling to automate the credit-to-cash cycle. HighRadius solutions have a proven track record of reducing days sales outstanding (DSO), bad debt and increasing operation efficiency.

Credit Research Foundation

RESOURCES

Visit the Credit Research Foundation website for additional resources

<http://www.crfonline.org>



© Copyright 2016 by the Credit Research Foundation.

All rights in this publication are reserved.

No part of the publication may be reproduced in any manner whatsoever without written permission.

Printed in the United States of America

Credit Research Foundation
1812 Baltimore Blvd. Suite H
Westminster, MD 21157

443-821-3000