ROI for your Software Project

Basing your Return on Investment Analysis on Sound Financial Principles

By Lubo Zizakovic, President
Insidus Corporation
www.insidus.com
August 2004

Like any project, investment or business venture, the ultimate decision on whether to proceed or not with the purchase/construction and implementation of a software application will depend most heavily on the financial analysis. If the application does not add measurable value to your organization in terms of benefits outweighing costs, then the project should never gain approval. The merits of any business venture always boil down to the "numbers", and there are plenty of valuation tools and methodologies floating around that describe how to calculate the return on investment (ROI) of your proposed software application. Unfortunately, much of the financial advice about software ROI floating around is flawed because it is not based on good financial principles. I am not exactly sure why some of those flawed analyses are proposed. I can only suspect that that those analyses are kept fairly simple because there is a general presumption that the decision makers who analyze software purchases do not have a high level of financial acumen or that the advice is coming from someone who lacks financial acumen themselves. Either way, the methodology used to evaluate a software project does not differ fundamentally from the methodology used to evaluate any business project. ROI is a standard measure of project profitability that is the discounted profits over the life of the project expressed as a percentage of initial investment. It has become such a crucial business tool that several techniques have been developed to measure it. We will talk about some of these methods such as the payback period, net present value, and internal rate of return among others.

For those individuals who believe that simplistic financial analyses are fine for evaluating projects and other investments, I recommend reading a book named "Big Bets Gone Bad: Derivatives and Bankruptcy in Orange County" by Philippe Jorion which chronicles the largest municipality bankruptcy in U.S. history when in December 1994 Orange County, California Treasurer Robert Citron lost $1.7 billion of Orange County's investment portfolio due to his misunderstanding of financial derivatives. The dollar amount implications will most likely not be as large regarding your application decisions, but you certainly do not want to be negligent in your actions like Mr. Citron was. Performing the proper financial analyses are only slightly more complex than the simplistic approaches that are prevalent, and quite easy to grasp even for those people who did not major in Finance. So why am I so hung up on numbers and proper financial analyses? Well, my MBA was in Finance and I started out my business career as an investment banker, so getting the "numbers" right is always important to me. It should be important to you as well because excluding important financial principles can lead to poor investment decisions.

So what is ROI and what does it represent? The ROI calculation results in a value that represents the benefits received from a project against the total costs of the project. The problem is that there are many varying opinions on how to calculate ROI. We will review the ROI calculation a bit later.

For those individuals who always thought that those number-crunching financial wizards are the pinnacle of quantitative accuracy, this may come as a bit of a shock, but financial forecasting accuracy relies on reasonable assumptions more than anything else. This means that if ten different individuals value a project or business venture, you will most likely receive ten different valuation amounts. The main difference between accounting and finance is that accounting is backward-looking and finance is forward-looking. Looking at and recording the past accurately is obviously much easier than accurately predicting the future. Many financiers will make reasonable and general assumptions about future cash inflows and outflows. This is fine for a "quick and dirty" back of the envelope valuation. I have always been a financier who breaks every line item down to the most granular level in making future assumptions. For

example, one valuator might assume that payroll costs will increase 10% each year and extrapolate forward. I would rather look at what the human resource needs will be next year, in two years, three, four, etc. and break salaries and benefits down by individual hire to support the proposed growth of the business. I will also model proposed salary increases, associated recruiting and training costs, severance costs for employee churn and any other future cost items associated with employee headcount. It is easy to see that my assumptions would be more accurate than just predicting a 10% year-over-year increase. The drawbacks are that accuracy takes time and it is difficult to predict anything accurately past three or four years. The advantages are a high degree of confidence in your analysis and the ability to justify and adjust any line item when questioned. The whole point is that the accuracy of your valuation will be directly correlated to the effort that you put into your assumptions about future events.

Before we get to actual financial valuation methods, let's talk about assumptions that must be made for your financial analysis, because these cost/benefit assumptions are the basis for any financial valuation. The one thing that you always must keep in mind is that costs and benefits must be recognized on a timeline when they are expected to occur.

Costs or Cash Outflows

When compiling costs or benefits, it is extremely important to include every item that would be impacted by implementing the proposed application. Some items are not blatantly obvious, so some real thought needs to go into this process. You will also find that it may seem impossible to attach dollar amounts to certain items such as management making better decisions as a result of having access to an enterprise business intelligence system. This is where finance becomes more art than science. You cannot predict with any accuracy such a line item, so you plug in a conservative and reasonable amount. Do not omit the item because doing so would skew your analysis to the negative.

Be careful to only include costs that would be incurred if the application were implemented. Obviously, this would include the cost of developing or purchasing the software application, but if implementing the application required your organization to purchase a new server to support the database back-end, then include this infrastructure cost. Do not count sunk costs such as infrastructure items not associated with the project (ie. leveraging the existing network servers).

Keep in mind that some costs are one time and others are recurring. For instance, you purchase hardware or software once but might pay maintenance on it every year. For initial project costs, consider them payable today, or in finance terms, time zero on a timeline. If maintenance and service contracts are prepaid annually, then include those first year costs at time zero as well. Costs incurred during the course of the year are recognized at the end of that year on the timeline. Costs or cash outflows may come in various form such as the cost of the software or the development of software, hardware (if necessary to support the application), the ongoing cost of maintenance and support if applicable, the cost of your employees time on the project if applicable, consulting fees if your proposed application requires significant customization to fit your organization, training costs (employees time and trainer costs if applicable) and any other cost that would be incurred if the application were approved.

Benefits or Cash Inflows

Direct Benefits

You'll discover that quantifying benefits can be much more difficult than quantifying costs simply because it is sometimes very difficult to measure the real benefits of using a technology that you are planning to deploy. Direct benefits tend to be easier to measure and predict than indirect benefits. An example of a direct benefit would be selling old technology that you are replacing.

In cases where real measurement cannot occur, you must rely on conservative and reasonable assumptions. In order to check that your assumptions are actually reasonable, you should discuss your assumptions and benefit amounts with other team members. Sometimes you will find that what you thought was reasonable before checking with others, no longer seems reasonable. Gather as much information or input as you feel comfortable with and make an informed assumption.

Indirect Benefits

Identifying indirect benefits and predicting the associated dollar impact will be your most difficult and time consuming task. How do you measure indirect savings? If an employee saves time or is more productive, what is that productivity worth to the company? If management could make a better decision because of the implementation of a business intelligence system, how is the dollar impact of that decision measured? Obviously predicting these dollar amounts will rely as much on art as they will on science, but this is where you make your conservative and reasonable assumptions.

Whenever possible, quantify your indirect benefits. If the new application would decrease 4 full time equivalent headcount once implemented, then apply those salaries and benefits cost savings to each year of your analysis (don't forget to include assumptions for future salary increases). If audit time is reduced from seven weeks to two weeks as a result of the system, calculate the employee hours saved over five weeks of an audit process and determine the dollar amount. Do not forget to apply annual savings to each year of your analysis if applicable. If the system saves five weeks of audit effort this year, then it will do so next year, the following year and every year that the system is used. You must also be careful not to double count interrelated benefits. Please remember to always write down your assumptions as a footnote to your analysis. You will be asked about your assumptions and it is highly unlikely that you will remember exactly what they are.

Below is a sample list of benefits. The list is certainly not exhaustive, but hopefully it will spark start your thought process on how the application deployment might impact your organization.

- Decreased full time equivalent headcount
- Decreased training requirements
- Reassigning human resources
- Elimination of errors
- Increased production
- Efficiency gains
- Time savings from automation
- Decreased employee turnover
- Decreasing/eliminating paper costs
- Better Management Decisions

- Job Satisfaction
- Opportunity Costs
- Strategic Competitive Advantage

It is important that you examine every aspect of your business that might be impacted by the deployment of the application and try to determine a dollar amount on a timeline for each benefit.

Financial Analysis Tools

The biggest issue that I have with many ROI analysis tools that I have reviewed is that the proposed analyses do not take the time value of money into account. The time value of money is a simple concept that everyone understands, and that is that a dollar today is worth more than a dollar tomorrow. It is the basis for discounting future cash flows to achieve a value at time zero, or today, but I've seen many simplistic analyses that omit discounting. There are software ROI tools out there that do incorporate the time value of money. I would definitely advocate the use of those tools over ones that do not account for the time value of money, but as you will discover shortly, just about every financial tool used to value software ROI is interrelated.

Net Present Value (NPV)

In simple terms, NPV is the difference between an investment's market value and its costs (sum of the present values of the net annual benefits minus the initial dollar investment required). The "P" in NPV means that we will derive a single dollar value for the investment today even though the life of the project may span many years. We begin by estimating the future cash flows that the new software will produce. For purposes of these calculations, any cost saving, incremental revenues or other benefits to the organization as a result of deploying the software are considered "inflows" and are positive numbers. All costs are considered "outflows" and are negative numbers.

As a general rule of thumb, technology investments are considered obsolete after three years. Depending on the investment, a three year time frame might not be reasonable. As a decision maker, would you agree to spend a million dollars on a new enterprise level CRM system if you knew that its useful life was three years and that in three years you would have to replace the entire system? Technology does change rapidly, but you will not replace every technology every three years unless your organization had money to burn and absolutely had to stay on the "bleeding edge" of technology. The hypothetical example below assumes that the lifecycle of the software application is 10 years. There may be upgrades and additional development along the way, but for all intents and purposes, management believes that the application will be used for the next decade. Numeric cost savings assumptions were made as well.

Time (years)	0	1	2	3	4	5	6	7	8	9	10
Initial Cost	(100,000)										
Inflows (Benefits)											
Strategic Advantage Over Competition		10,000	10,000	10,000	10,000	10,000	10,000	10,000	10,000	10,000	10,000
Reassigning Human Resources		15,000	15,000	15,000	15,000	15,000	15,000	15,000	15,000	15,000	15,000
Time Saving From Automation		20,000	20,000	20,000	20,000	20,000	20,000	20,000	20,000	20,000	20,000
Decreased Training Requirements		5,000	5,000	5,000	5,000	5,000	5,000	5,000	5,000	5,000	5,000
Decreased FTE Headcount		50,000	50,000	50,000	50,000	50,000	50,000	50,000	50,000	50,000	50,000
Better Management Decisions		50,000	50,000	50,000	50,000	50,000	50,000	50,000	50,000	50,000	50,000
Decreased Employee Turnover		30,000	30,000	30,000	30,000	30,000	30,000	30,000	30,000	30,000	30,000
Increased Production/Job Satisfaction		5,000	5,000	5,000	5,000	5,000	5,000	5,000	5,000	5,000	5,000
Elimination of Errors		5,000	5,000	5,000	5,000	5,000	5,000	5,000	5,000	5,000	5,000
Opportunity Costs		20,000	20,000	20,000	20,000	20,000	20,000	20,000	20,000	20,000	20,000
Outflows (Costs)											
Maintenance Fee		(10,000)	(10,000)	(10,000)	(10,000)	(10,000)	(10,000)	(10,000)	(10,000)	(10,000)	(10,000)
Software Support Fee		(5,000)	(5,000)	(5,000)	(5,000)	(5,000)	(5,000)	(5,000)	(5,000)	(5,000)	(5,000)
Additional System Requests		-	-	(25,000)	-	-	(25,000)	-	-	(25,000)	-
Net Inflow		195,000	195,000	170,000	195,000	195,000	170,000	195,000	195,000	170,000	195,000
Net Cash Flow	(100,000)	195,000	195,000	170,000	195,000	195,000	170,000	195,000	195,000	170,000	195,000

Figure 1.0

The hypothetical example above is quite simplistic in its assumptions (ie. no adjustments over time for salary increases or reduced headcount, no increases in maintenance or support costs over time, etc.) in order to keep the example as simple as possible. Once your cash flow projections are laid out by year (initial cost of the software is at time zero, or today and cash inflows and outflows start at the end of year 1), we need to discount all of those future cash flow estimates so that we can get a value today. This process is also referred to as discounted cash flow (DCF) valuation. To discount the cash flows, we must determine a discount rate, or your required rate of return. This rate is also known as your cost of capital. This white paper will not go into the various methods that could be used to determine the discount rate, so for simplicity's sake, let's presume that we would be happy with a 15% return, so 15% will be our discount rate. We will discount each future cash flow (years 1-10 in our example) so that we get a value for each cash flow today. Keeping in mind the concept that a dollar today is worth more than a dollar tomorrow, cash flows farther out will be discounted more heavily than those that occur earlier. Each cash flow is discounted back to today's dollar terms using the formula below where r = discount rate and n = year.

$$\frac{\text{Cash Flow Amount}}{(1+r)^{n}}$$

Years	1	2	3	4	5	6	7	8	9	10
	$\frac{195,000}{(1+0.15)}$	$\frac{195,000}{(1+0.15)^2}$	$\frac{170,000}{(1+0.15)^3}$	$\frac{195,000}{(1+0.15)^4}$	$\frac{195,000}{(1+0.15)^5}$	$\frac{170,000}{(1+0.15)^6}$	$\frac{195,000}{(1+0.15)^7}$	$\frac{195,000}{(1+0.15)^8}$	$\frac{170,000}{(1+0.15)^9}$	$\frac{195,000}{(1+0.15)^{10}}$
	=	=	=	=	=	=	=	=	=	=
	169,565	147,448	111,778	111,492	96,949	73,496	73,308	63,746	48,325	48,201

Figure 1.1

As you can see, the $195,000 amount in year one is worth considerable more in today's dollar terms than the same $195,000 amount 10 years out. Since our initial investment amount occurs today (when we pay for the software), we do not discount that amount because it is already in today's dollar terms. If we sum the initial cost (-$100,000) and all of the discounted cash flows, we get a NPV of $844,307. Interpreting the NPV result is very simple. Your accept/reject decision is based on whether the NVP amount is positive or negative. In this case, our NPV is positive, so we should go ahead with the software purchase because it will

provide a return to the organization above our required 15%. Unfortunately, NPV is not widely used as a decision making tool for projects because many companies are fixated on a percentage of return. The NPV technique should really be used to determine if the project gets a thumbs up or a thumbs down on a financial basis. If your organization is looking at multiple application options, many projects might get a green light. So how do you decide which one is the best option?

Internal Rate of Return (IRR)

The IRR is the discount rate that makes the NPV of a project equal zero. The IRR provides us with a single rate of return that summarizes the merits of the project. For our example above, when you set the NVP to zero, the IRR is 193%, which is well above our required rate of 15%. This tells us that we should undertake the project. Just one quick word of caution about IRR calculations, they work for conventional project cash flows, meaning that the first cash flow (initial investment at time zero) is negative and all of the remaining cash flows are positive. If the cash flows are not conventional, then you might have multiple rates of return where the NPV is zero. Many computer financial packages do not compensate for this problem and will return the first or lowest IRR. In the case of non-conventional cash flows, use the NPV method because strange things can happen to IRR. If the cash flows for multiple projects are conventional, then the IRR method is a great way to compare projects because you can talk in terms of rates of return instead of dollar amounts (NPV).

Comparing two or more projects in rate of return terms allows you to evaluate the project on a level playing field. This means that project “A” might return a large dollar amount as shown in the NPV calculation and project “B” might return a much smaller dollar amount, but project “B” might have a much higher rate of return, or IRR. As you can see, IRR is just another way of looking at NPV in percent terms. For either calculation, you must still set up your timeline of costs and benefits such as the one shown in Figure 1.0. IRR does have a practical advantage over NPV in certain circumstances. NPV cannot be estimated unless we can determine the appropriate discount rate or cost of capital, but IRR can still be estimated. If we performed an IRR calculation and found a 50% return, we might be inclined to approve the project because it would be very unlikely that a required return would be that high.

If you want to calculate the IRR quickly, open an Excel spreadsheet and lay out your projected net cash flows over time (see Net Cash Flow line in Figure 1.0) and use the IRR function. Do not use the discounted cash flows from Figure 1.1 because you have already made a discount rate assumption and IRR determines the discount rate for you which makes the NPV of the project zero, or the point where you would be indifferent to accept or reject the project.

Payback Period

The payback period is the amount of time needed for an investment to generate cash flows to recover its initial costs or another way to look at it would be the point at which total benefits equal total costs. This concept is widely understood and used, particularly when evaluating IT projects. If we look at our sample above from the NPV section, we see that at the end of year one, we have $195,000 in cash flow which is greater than the initial cost of the software (-$100,000). This means that the payback period occurs at some point during the first year (100,000/195,000 = .51 years or just over 6 months).

The payback period method has some serious shortcomings. It is calculated by simply adding up the future cash flows. There is no discounting of cash flows, so the time value of money is

ignored. Without discounting the future cash flows, your project will look much more attractive than it really is.

Not only will the project look more attractive, but since there is no required rate of return used, the risk level of the project is never captured. This means that a very risky project is treated the same as a low risk project. I read another paper recently stating that the payback period provides an indicator of risk, presumably that a project with a shorter payback period is less risky. This statement is absolutely false. Risk is captured in your required rate of return. It is an easy concept, the higher the risk on any investment (think of your stock portfolio), the higher your required return will be. How quickly an investment recovers its initial amount invested does not measure risk. For example, you could buy a lottery ticket today for the $10million draw tonight, or put a sum of money on the roulette wheel at a casino. If you win, you would have recovered your initial investment very quickly, but the extremely quick recovery of your investment in no way measures the risk involved in the investing activity. Your required rate of return on the lottery ticket is astronomical and if you do not discount your future cash flows, then the risk of your investment actually paying off is never captured in your analysis. The biggest shortcoming with the payback method is that there is no economic rationale for determining the correct cutoff period. An arbitrary cutoff period must be chosen, so you need to decide whether 2 years is acceptable, or 4 years, or 5, etc. The payback period also tends to bias the user toward short term investments as it ignores cash flows beyond the cutoff. A project that takes a few years to get up to speed and then creates phenomenal returns would be rejected strictly on its cash flow profile.

With all of the shortcomings of the payback period method, it is easy to see why you should put very little weight on the analysis results other than as a very general guide when looking at two fairly comparable projects. If payback period is a valuation metric that your organization tends to look at, you can and should perform a discounted payback period analysis where you determine your discount rate and discount the future cash flows before performing the payback analysis. This would eliminate some of the shortcomings mentioned above, but very few individuals ever perform such an analysis in practice. Use the discounted payback method only as a quick and dirty valuation method to value a project on the “back of an envelope” and as just another valuation metric in conjunction with the others mentioned here. Do not base your accept/reject decision upon it.

Profitability Index (PI) or Benefit/Cost Ratio

The PI is the present value of future cash flows divided by the initial investment. The present value of future cash flows is another term for discounted cash flows calculated exactly as shown in the NPV section. The PI would be larger than 1 for positive NPV projects and less than 1 for negative NPV projects. For our example above, the PI would be 9.44. The PI measures “bang for the buck”. It tells us that for each dollar invested, the organization receives $9.44 in value. The PI and the IRR valuation methods are obviously very similar to the NPV method. They just present the results in a different fashion. An attractive NPV project will also look attractive on an IRR or PI basis and vice versa. IRR and PI allow you to compare multiple projects on a level playing field. Just be careful, because it might make more sense for your organization to pursue a high NPV project even though it carries a lower IRR and PI than another comparable project. Earning a 500% return on a $10,000 project might not add much value to your organization, whereas earning 22% on a $5 million project would add considerable value.

Return on Investment (ROI)
ROI and the financial tools above are designed to help you build a business case to support your technology decision by evaluating the real impact to your corporation's bottom line. ROI and other financial analyses should not be your only evaluation criteria. For example, a higher ROI project might not have a very intuitive GUI. If most of your end users are not very tech-savvy, then they might not adopt the system. This type of event might be impossible to predict. If you cannot accurately reflect that dynamic in your assumptions, then you might invest hundreds of thousand of dollars into a system that users refuse to use. Non-quantitative input is required here and should be equally as important to your decision making process as the financial analysis.

Be careful when talking about ROI as many people tend to talk about ROI without including time horizons. Every time you use ROI as a comparison of two or more projects make sure you are comparing "apples to apples" meaning that the ROI figures being compared are for equal time-frames.

So what is ROI? If you try to look up the definition or ask someone, you will get a variety of different answers. For our purposes, ROI is a return ratio that compares the net benefits of a project, verses its total costs. As such, the ROI calculation represents the relative value of the project's cumulative net benefits (benefits less costs) over the analysis period, divided by the project's cumulative total costs, expressed as a percentage.

$$\text{ROI} = \frac{\text{Average Net Benefits}}{\text{Initial Costs}}$$

In our example from Figure 1.0 above, we find that our return is 844% over the life of the project. Whenever anyone talks in ROI terms, they are speaking on an annual basis, so our 844% return has an ROI of 84% per year over a ten year period. If we were to perform the same ROI analysis on the first three years only, we would see a total return of 329% or an ROI of 110% per year over the first three years. It makes sense that our ROI is higher for a shorter time frame because discounting has less of an impact on cash flows in the early years. The important thing is the time frame. If you compare a project with a three year life cycle with another project that has a five year life cycle, make sure you compare 3-year ROIs but do not forget to consider the 5-year ROI as well, otherwise you will be discounting to zero the benefits of those final two years. This is why time frames are important to know when comparing ROIs for decision making purposes.

Taxes and depreciation can also have an impact on your financial analyses, but that discussion is beyond the scope of this white paper. Use the above financial valuation techniques to calculate an ROI for your proposed software projects. The analyses will force you to think about how the software will impact your organization and consequently will guide you so that you make the right software decisions for your organization with confidence. Ultimately, you must weigh financial returns against many other factors such as risk, timing, platform issues, user acceptability, competitive issues, feasibility considerations, etc., but you cannot build your business case without a solid financial analysis.