

Getting on top of IT

C. Verhoef

*Free University of Amsterdam, Department of Computer Science,
Amsterdam, The Netherlands*

x@cs.vu.nl

Abstract

Information technology is becoming the largest production factor in many organizations. Yet most executives encounter insurmountable problems in governing information technology: IT is on top of them. New research shows that executives can reverse this situation, without having to master the intricacies of IT itself.

Introduction

We all know that you don't need to unfathom the innards of a boiler in order to tap hot water. Likewise, there is no need for an executive to obtain a degree in civil engineering before deciding on building the company's new headquarters. In general, governance at the top level should be detached from the underlying technicalities that are subject to governance. Of course, you do need certain input for executive decision making, e.g., specific financial and strategic information. Ideally, executive staff members *do the math* so that the company leaders can focus on what they do best: cutting the corporate Gordian knots.

The IT-proponents made many executives believe that for information technology things are entirely different. Current paradigms indicate that the executive responsible for IT understands the hottest acronyms, deals with jargon ridden techno talk, and—based on this—prudently decides. Frankly, this is the same as assuming that the latest materials, strength calculations, and the best concrete mixtures are the key input for the board to decide on its new headquarters. While it is obvious that the latter technicalities are not the proper ingredients to decide on a new HQ, it is common practice in the IT-world to myopically focus on exactly such aspects.

To alleviate the problems with IT governance the Chief Information Officer (CIO) was invented. The CIO is the sublimation of an IT-savvy technologist and a business visionary. This officer is thought to be the designated person who can be trusted with the IT assets—just like the CFO is trusted with the company's financial assets. But how much trust is given to CIOs? A consistent stream of data shows that the average tenure of CIOs is only about two and a half years. The prime reason for *making them available to the industry* is: failed IT projects. No wonder that some people think that CIO stands for *Career Is Over*. This pattern is confirmed by Standish Group who is collecting data on IT failures, cost overruns, and successful IT projects. Their consistent findings are that 30% of all IT projects is cancelled, 50% are runaway projects, whereas only 20% is on time, within budget, and has the desired functionality. In absolute numbers for the U.S.: in 1995, \$81 billion was spent on cancelled projects, and an additional \$59 billion was wasted on cost overruns. This deplorable reality makes us think of the CIO

as the unicorn of the corporate world: while many and diverse powerful properties are attributed to the CIO, these hardly ever materialize in real life.

Still, we are talking significant figures: suppose that Standish group is right, and 30% of your IT budget could be cut by not initiating sure-failure IT projects, and another 50% may be saved by taming runaway IT projects. If only you knew how to identify the good, the bad, and the ugly, you could avoid buying into triple D performing IT projects, exploit the triple A performing initiatives even further, and nurture the others to improve their performance. But how?

IT portfolios

You could use something similar to how you would manage a security portfolio. The 1990 Nobel Laureate Harry Markowitz was the founding father of security portfolio management. His portfolio analysis starts with collecting and interpreting information concerning individual securities. And it ends with conclusions concerning entire portfolios. The purpose of security portfolio analysis is to optimize the portfolio to meet the objectives of the investor. For the crop of IT systems in an organization this is not different: for IT projects we also need to understand the relationships and procedures by which relevant information about IT systems is transformed into conclusions about IT portfolios to meet the objectives of the IT investors.

The idea of IT portfolio thinking is not new: McFarlan published in the early 1980s a qualitative method to manage IT risk as a portfolio in Harvard Business Review [5]. Recently the trade-press picked up the IT portfolio idea again and reported on substantial cost savings by revealing (and removing) redundant IT projects [1]. In addition, the federal U.S. government requires an IT portfolio approach in the Clinger Cohen Act of 1996. A long string of reports on value destruction through public IT spending eventually led to this act, literally stating that IT investments must:

reflect a portfolio management approach where decisions on whether to invest in IT are based on potential return, and decisions to terminate or make additional investments are based on performance much like an investment broker is measured and rewarded based on managing risk and achieving results

Only no one knows how. But, popular belief is that Markowitz's work is applicable somehow. This belief is so powerful because the goals of managing security and IT portfolios coincide. However, the means to reach these goals are entirely different. A major reason why Markowitz's modern portfolio theory (MPT) does not transpose to IT portfolios is that IT systems become illiquid after adoption. The nature of securities is that you can invest and disinvest in them and implement a decision without prohibitively large costs, and often in a matter of seconds. Modern portfolio theory essentially asserts that investing and disinvesting in securities is enabled at all times. The heart of security portfolio management is to optimize the security *selection* process. An IT investment is a different beast: it sometimes takes years to construct and implement IT throughout an organization. After implementation, it is no longer possible to exchange the IT solution for another, even if the chosen solution is no longer optimal: it became part of the corporate DNA. For instance, recall the Year 2000 problem: about 30% of all the IT budgets in 1999 was spent on its solution. According to security portfolio theory, every infected system should have been immediately junked because of underperformance. But this did not happen, because of the illiquidity of IT.

As Markowitz himself noted in a recent interview [3] about applicability of his work to manage portfolios of projects: “I would be cautious about applying MPT to corporate projects as though these are liquid assets.” And “The company (or Division) manager cannot pretend that he is selecting liquid assets subject to a budget constraint.” Let us explain the illiquidity a bit more. The nature of successful software assets is that they are continuously adapted to changing business needs, government regulations, other technology etc. This process induces a constant accumulation of crucial business knowledge. Unfortunately, this incremental knowledge-growth is a *write-only* process, adding to the illiquidity. By encoding valuable business rules, programmers actually encrypt the logic, and throw away the key: the original system documentation (if available at all) is usually not updated. If you think this is outrageous, consider this: there are no certification demands on computer programmers. In other areas it is common that only graduates are certified: for instance, physicians and surgeons are without exception doctors of medicine. If a certified computer programmer is someone with a graduate degree in computer science, then at least 87% of the computer programmers in the U.S. is not certified [6, p. 168] (about 12% has a graduate level, but not necessarily in computer science). Despite the best intentions of both the computer programmers and the people hiring them, reality is that understanding an IT system now amounts to understanding the execution behavior of the code by reading it. The programmers jargon “use the force, read the source” indicates that only Jedi Knights will succeed. Indeed, it is like trying to understand humanity by solely reading the human genome. All this does not add to the badly needed transparency to properly govern IT.

Why is IT on top?

Without proper governance, steering, and management controls, it is hardly possible to get on top of IT as an executive. Adding to the situation is the strong belief that it takes one to manage one, so that professional managers easily feel somewhat insecure when IT decision making is at stake. Typically, executives take their *technical man* with them like an amulet protecting against black magic. The formal chain of command is of course that the final decision is up to corporate management, but the actual power is often with the ones deeply entrenched in the software assets, like voodoo priests who influence the cause of action in the background. So, in many practical cases, the *technical man* is implicitly in command. And this line of command and control turned out not to be very effective: benchmarks reveal that 75% of all organizations world-wide is on the lowest level of the IT development maturity scale—the Capability Maturity Model (CMM).

CMM level	Meaning	Frequency of Occurrence (%)
1 = Initial	Chaotic	75.0
2 = Repeatable	Marginal	15.0
3 = Defined	Adequate	8.0
4 = Managed	Good to Excellent	1.5
5 = Optimizing	State of the art	0.5

Table 1: Distribution of organizations over CMM levels.

In Table 1, taken from [4, p. 30], we show a recent distribution of organizations over

CMM levels. Level 1 means anarchy: no repeatable process is in place. In particular, there is no overall metrics and measurement program. Just like with securities, you need data on individual IT projects to aggregate this into conclusions on entire portfolios. But in an IT-anarchy, there is no ready availability of data, so how can you get the management information you need? The intuitive answer is to establish a metrics program. But, this has shown not to work: since 1988, the ratio of starts to successes has remained remarkably consistent at about 5 to 1, or four in five metrics programs fail to succeed [7]. Are executives doomed in that they will never master IT decision making?

The short answer is: “no”. Extensive research on quantitative methods for IT portfolio management and hands-on application of such techniques in practice show that you can bypass the deficiencies of immature IT departments by utilizing public benchmarks as a surrogate. After a jump-start you can establish more precise internal benchmarks and finally you can craft a company-specific set of macro-economical formulas modeling the important financial aspects of the company’s IT. This will support corporate leaders in getting on top of IT. Let’s take a look at how this works by giving a few examples.

IT-synergy scrutinized

Paul Strassmann (a former CIO of the U.S. Department of Defense) once wrote: “Credible financial analyses are necessary before top management can act with an understanding of the consequences of any decision.” To get a feel for the subject matter, let’s see how executives can be supported with such financial analyses.

An IT synergy project is the amalgam of similar IT projects to be carried out as a single overall IT project. Usually, you have to rely on your local IT-guru’s opinion how synergistic the combined project is going to be, but new research shows that you can get an impression yourself pretty quickly. Suppose that two business units of a company need almost the same information system. Upon asking we get the estimated development schedules: about 12 months each—after all they are fairly similar. We calculated that the costs of each IT project are most likely going to be \$780,000. Doing this twice then will cost \$1.56 million. A synergy project has real synergy if the total cost and risk will be below that of doing things twice. So we calculate the most likely time a single information system project of \$1.56 million is going to take: 14.5 calendar months. So for twice the price, you win only two and a half months. In addition the risk of failure of the 12 month projects is 13% each, and this risk increases with 3.5% to 16.5% for a 14.5 month project. Now how realistic is it that the variation points, the additional communication, ownership issues, the additional risk, the postponed deadline, and more issues are solved in less than 2.5 months? Not likely, so don’t do it.

The good news is that you need zero IT knowledge to calculate the above numbers, i.e., you don’t need the archetypal *technical man*. Moreover, the quantitative data gives you the support you needed to come to a final decision.

If the amount of similar systems increases, there is a cut-off point where synergy costs are going to pay off, but simultaneously the risk of failure for the *grand* IT project will increase. Quantifying both cost and risk will then help in making an informed decision. Of course, more input is taken into account for decision making: the business criticality, the risk profile of the company, the deepness of the pockets, whether economies of scale are feasible, etc. As many executives experienced, accurate numbers are not routinely supplied by the IT departments. It is more likely that the *technical man* is

convinced that you save in the order of \$780,000 by combining the two projects: that's why it was called an IT synergy project in the first place. So, how can top management get this kind of support to (dis)invest in IT based on sound financial arguments? In fact, not different than deciding on other assets. For instance, to manage the intellectual property of a company, most executives seek the advice of patent lawyers to develop a sound strategy. Similarly, to manage the financial assets of the company, financial specialists provide top management with information on risk and potential return, so that the best possible investment strategy ensues. Most top executives will not apply modern portfolio theory, option analysis, or other sophisticated analysis tools themselves. Instead they hire executive staff for this.

For information technology, a similar structure is necessary. Ideally, to manage the software assets of the company, a corporate information technology department provides top management with the company's accumulated data on cost, risk, and potential return, so that informed decisions on software can be made. We use the word software here on purpose. The reason is that when people think of IT assets, immediately a hardware focus comes to mind. And although hardware costs are substantial, the software costs surpass them by an order of magnitude. One estimate is that only 9% of the total IT costs are attributable to hardware—the rest is spent on software. Overall, tailor-made software is the dominating cost factor.

CredIT, ergo sum

In the old days, the philosophers debated whether it should be *credo, ergo sum* (I believe, therefore, I am) or the other way around: *sum, ergo credo* (I am, therefore, I believe). Thereby characterizing the important thought patterns of that era about human existence, religion, and their interdependencies. Along the same line, we caught current thought patterns in information technology by the phrase: *sum, ergo credIT* intended to mean *I am, therefore, I create value with IT*. This summarizes the manifest thought patterns of the information age about corporate viability, information technology, and their interdependencies. Undoubtedly, information technology pervaded in all the veins of society, and almost every business process is computerized. But, this “*carpe diem*” spending on information technology is not necessarily leading to sustainable value creation. On the contrary, it can easily lead to bankruptcy, as the dot com hype painfully illustrated [2]. Therefore, we can no longer afford ourselves to make added value subordinate to information technology and spend in blissful ignorance without pondering on the value creation proposition. The landscape has changed, and reversal of the sky-is-the-limit adage is imminent: *credIT, ergo sum*. This articulates the idea that you first need a sound business case, before embarking on investing in information technology. We think of this adage as: *I create value with IT, therefore, I am*. To implement this “*carpe dime*” adage, an investor's perspective on IT is mandatory. Only then, you can come to grips with cost, risk, and potential return. This necessitates fact-based reasoning, as is common with managing other assets. For instance, financial assets often have a rich source of data, just think of historical stock market information that is available for security portfolio analysis. For information technology there is no such thing (due to its immaturity). But we do need at least *some* data to understand cost, risk and return of IT. We experienced that in most organizations, the following data points for most of the IT projects are available:

- initiation date;

- delivery date;
- project costs.

Even this used to be a problem in the *sum, ergo credIT* age, but since the tide has reversed, at least you need to come up with some data before management approves. Still, these sparse data points do not come on a silver platter (but they can be recovered). When you establish a corporate IT department that systematically gathers such information, we experienced that this situation rapidly improves. But getting more detailed relevant data remains a problem. To compensate for the lack of data, we use (public) IT benchmarks as a surrogate. From this minimal data set plus these external IT benchmarks it is possible to obtain fairly accurate management information, supporting decision making. For that you need to establish and nurture executive staff that can deal with the salient issues, just as you need specialists for financial asset management.

To give you an idea of what your executive staff could have done to come up with the data of the synergy example, see the side-bar “Do the math”. The point we like to make there is that given only minimal input data from the IT department, it is possible to come up with relevant information to support rational decision making. The formulas in the side-bar are not displayed to encourage top managers to learn them by heart, but to substantiate the claim that via them, you can get on top of IT, without an IT-voodoo seance.

Side-bar: Do the math

The input data for the synergy example is threefold: the duration d in months of both projects: $d = 12$ months. Furthermore, we assume a fully loaded daily rate r of \$1000 for IT-personnel, and 200 working days per year (abbreviated w). Note that from the IT-department we only need the estimated durations of the IT projects. The other inputs are easily retrieved from the accountancy department. We used the following formula to calculate the benchmarked total cost of development:

$$tcd(d) = \frac{rw}{1800} \cdot d^{3.564}$$

where tcd stands for *total cost of development*. With a scientific calculator you can then estimate the most likely costs: $tcd(12) = 779757.30 \approx 780,000$ U.S. dollar. If you have two such projects, we simply take twice the amount: \$1.56 million. We use another formula to calculate for a given IT-project costing c U.S. dollar, its benchmarked duration:

$$dd(c) = \left(\frac{1800c}{rw} \right)^{0.28}$$

where dd stands for development duration, r is the daily rate again, and w the number of working days per annum. So, $dd(1559514.608) = 14.495 \approx 14.5$ months, i.e., for \$1.56 million you have a 14.5 month IT-project. This small difference in duration, combined with a large price difference is relevant input for decision making. Apart from the cost and time-to-market, we also came up with a chance of failure. We used the following formula for that:

$$cf(d) = 0.4805538 \cdot \left(1 - e^{-0.007488905 \cdot d^{1.506090}} \right)$$

where cf is short for chance of failure, and d is again the duration of the IT project. Note that e is a special mathematical constant available on all scientific calculators (mostly denoted as e^x). If you go the pain of typing in all the weird constants, you will find $cf(12) = 0.130$, which means that the risk of failure is 13%. Likewise, for the synergy project that takes 14.5 months we find: $cf(14.495) = 0.165$, so an estimated risk of failure of 16.5%. This example, the formulas, and their derivations stem from our elaborate treatment elsewhere [8].

Seismic IT activity

The synergy example showed the nuts and bolts of how to support executives with data relevant for decision making. Now let's look at a more complex IT investment example where we will encapsulate the machinations of your executive staff, but only present the aggregate information. Mathematical derivations and statistical analyses for the present example plus an elaborate treatment of the underlying technicalities can be found in the paper [8].

Suppose a large fictitious organization thinks of substantially investing in IT. For instance the executives are so sick and tired of the IT situation, that they plan to overhaul the major systems to improve the situation once and for all. Think of a new ERP system, some CRM implementations, payroll systems, various Intranet applications, an enterprise web-server, and an abundance of supporting smaller IT projects. In Table 2 we summarized this IT investment impulse.

proj. #	<i>tcd</i> \$mn	ready months	retire months	cost all. dev. \$mn/month	cost all. ops \$mn/month
50	15	27	126	27.5	5.49
10	30	33	146	9.0	1.81
6	75	43	176	10.5	2.10
3	150	52	203	8.6	1.73
2	300	63	234	9.5	1.90
1	600	77	260	7.8	1.56

Table 2: IT investment impulse.

In total, 72 IT-intensive projects of varying size were identified. The only data points we have for all the new IT projects are estimates of their total cost of development. These amounts are in the second column (*tcd* stands for *total cost of development*). In the first column we give the number of projects with the same price. So we have 50 projects of \$15 million, 10 of \$30 million, etc. The total cost of development for the entire IT impulse is, therefore, \$3.15 billion. Using formulas similar to the ones we already used, we calculated the most likely development durations for all IT-projects. These are in the third column; so a \$15 million project is ready in 27 months, a \$30 million project takes 33 months, etc. Then we estimated the lifetime of these projects: a 15 million dollar IT system will retire after 126 months, which is a little over a decade, a \$30 million IT-system retires after 146 months, and so on. Then we calculated the cost allocation per month for both development, and keeping the systems running: the minimal operational costs. So to build the 50 IT systems of \$15 million, you need to allocate 27.5 million dollar per month, for 27 months. After development,

you need to allocate \$5.49 million per month to keep those 50 systems operational. Likewise for the other IT systems (viz. the last two columns).

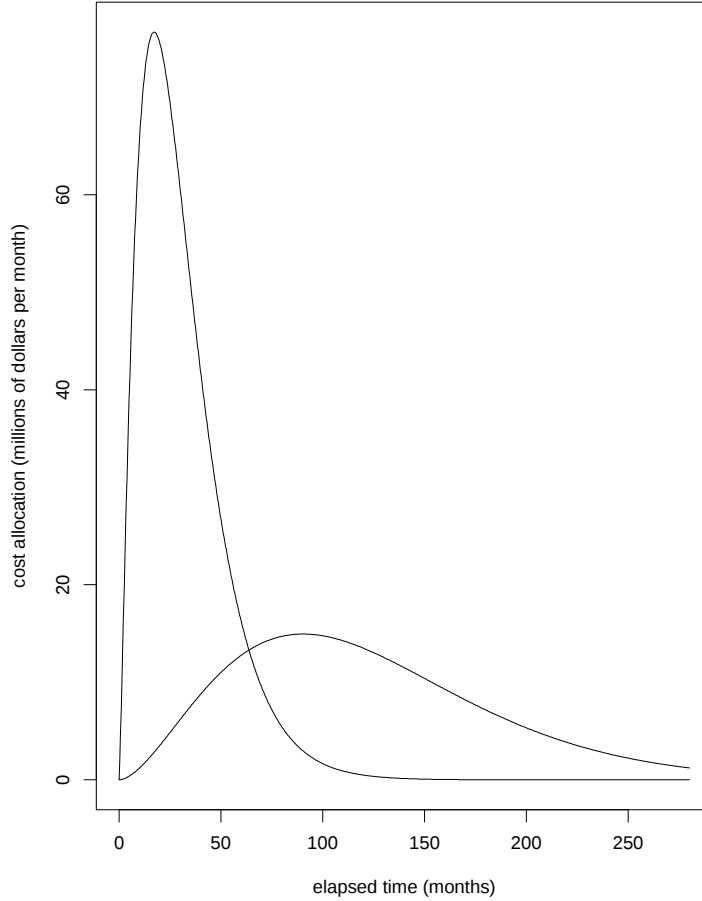


Figure 1: Seismic IT costs induce an operational cost tsunami.

We used the data points of Table 2 to infer cost allocation curves both for development and operations. A cost allocation curve shows you when to invest and how much, just like a monthly payment schedule for a mortgage to build and inhabit a house. We depicted both curves in Figure 1. These curves are typical for the type of investment and they have names. We call a sudden IT-investment a *seismic IT impulse* (it is the peaky one), the other curve with a long wavelength is called an *operational cost tsunami*. A seismic IT impulse causes an operational cost tsunami, just like submarine seismic events can cause a tsunami (a great sea wave produced by submarine earth movement or volcanic eruption). Indeed, for development costs there is a sudden rise in costs with a maximum at \$77 million when total IT development is in its 17th month. Then when development cost are decreasing, operational costs are beginning

to rise. This is a long-lasting wave, with a maximum at about \$15 million at the 90th month after the seismic IT impulse. With the cost allocation curve for development, we checked the accuracy of the model. We calculated the surface under the seismic IT impulse, a model representing the total development cost of the IT impulse, and this deviated 0.02% from the actual investment of \$3.15 billion. For the operational cost tsunami, we could not do such a check, since those costs were initially not projected (which is not an exception, but the rule). But we also calculated these costs: \$2.3 billion. So the minimal total cost of ownership for this IT-investment is 5.45 billion dollar. But it is probably more: the 5.45 billion contains an estimate for successful implementation of both development and the minimal cost of operation. It does not include functional enhancements, project failures and restarts, cost and time overruns, replacement of retired systems, etc. Therefore, it is a rather conservative estimate, a lower-bound.

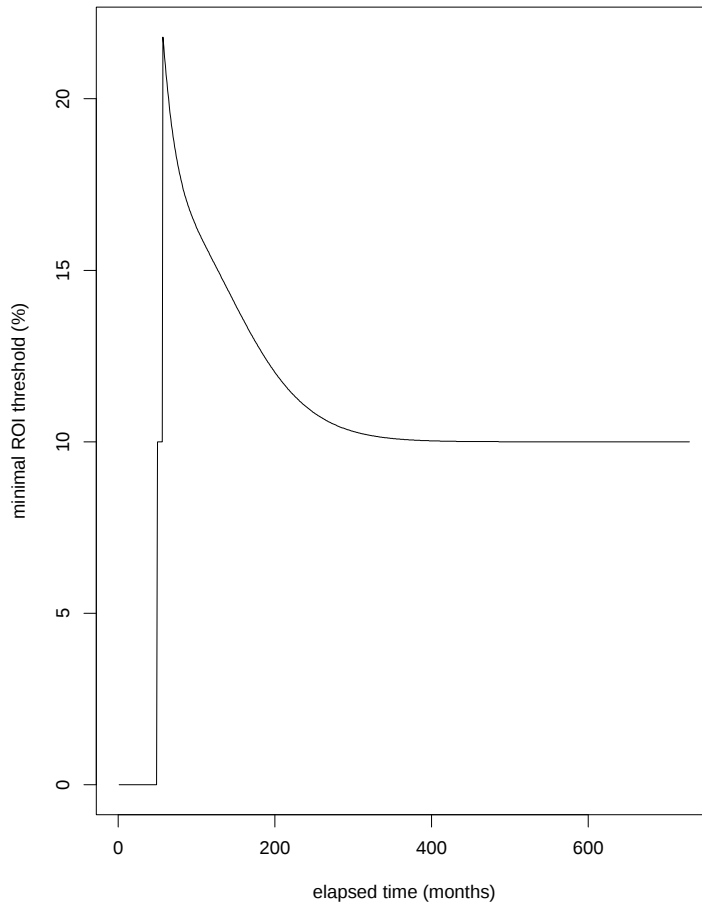


Figure 2: Minimal ROI threshold over time.

But these lower-bounds are a useful means for decision making. They give you input whether the investment makes sense, at all. Since, at least the prospective returns should make the effort worthwhile (*credIT, ergo sum*). Suppose that the company's ROI threshold for IT-investments is 10% per year. For the seismic IT-impulse, this amounts to minimally 315 million dollar return per annum to approve of the investment. So, we calculated the minimal ROI threshold over time. Such an *iso-ROI-line* also has a typical shape. We call it an *ROI quaver* after the quaver-shape of an eighth note in musical notation. In Figure 2 you can get an impression what return your business case minimally needs to get a net 10% bottom line. The first 50 months are the investment period, so we do not expect a return. Hence the horizontal line at zero in Figure 2. After 50 months, we need a 10% bottom line. We calculated that the initial investment of \$3.15 billion is consumed in the 57th month, so the 10% horizontal line is only 7 months long, and more a tremble in the vertical line up to the actually needed ROI. After 57 months, the initial investment is consumed, and to obtain a net 10% bottom line, the \$2.3 billion needs to be taken into account as well. The iso-ROI-line indicates that the minimal ROI threshold initially is over 20% and for many years to come the ROI needs to be well above 10% in order to finance the IT investment and comply with the 10% bottom line.

For the entire IT investment we calculated two exposures: the aggregated chance of failure (12.6%) and the aggregated chance on serious cost overruns (13.5%). As you can see, these percentages deviate from the ones given by Standish Group. The reason is that there are a lot of small projects (more than 50) and only a few large ones. But the large ones do have a high risk-profile: there is at least a 50% chance for both failure and serious cost overruns. for a more detailed analysis, you may need indicators like the pay back period, and other economic indicators, like net present value, risk-adjusted return on capital, and more. Also note that current, then, and future dollars must be taken into account for such long periods.

The point we like to make here is that with relatively sparse data you can get a clear impression of the cost, the risk, and the minimal return you need to have a sensible IT investment at all. Also the long-term consequences in terms of minimal cost of operation, how much return is needed to finance the IT investment, the lifetimes of the various systems, are made transparent. Moreover, risk indicators for failure, and the risk on cost overruns are given.

Don't try this at home We obtain frequent questions to provide very simplistic tools to enable quantitative IT portfolio management for all. Investing in breeding and educating your staff is crucial, as Derek Bok, a former president of Harvard University once said: if you think education is expensive, try ignorance. It is not possible to dummy down the necessary work to a level where after keying in some data, the relevant charts pop up by turning the handle. It is not possible either to encapsulate the mathematics and statistics in some tool, doing the job. For instance, the derivation of curves as in Figure 1 requires some mathematical craftsmanship: you must come up with a first guess of the exact shape and coefficients of such a curve, and only then using the input data and a sophisticated programmable statistical package, we have a chance to fit a curve through the actual data—but there is no guarantee for success. It requires some mathematical juggling, so, unless you are fluent in mathematics and statistics, don't even THINK of using the formulas that we used to create the insights that we just presented. Instead, hire someone who can, like you would also do to manage other types of assets like your real estate, financial assets, and intellectual property.

At the time of writing this paper, you can't open a can of IT portfolio specialists, so you have to breed them. Typically you need people graduated in mathematics, statistics, econometrics, physics, astronomy, financial mathematics, business mathematics, biostatistics, etc. The more theoretical their background, the better: such people master the art of turning brittle data with mathematical and statistical means into useful output. It is our experience that they become productive when they mastered the principles of quantitative IT portfolio management [8]; the underlying methodology we used for the two given examples.

* * *

In short, the key to getting on top of IT, is to understand the cost and risk of IT, so that the minimally needed returns can be projected. A quantitative IT portfolio approach supports in answering these important questions. It can help in detecting the necessary return from other IT investments, to see if a mandatory cost can be balanced by other IT-investments. To jump-start quantitative IT portfolio management, you need to breed people initiated in the underlying mathematically oriented macro-economic theory of IT. This, accompanied with your own or public IT-benchmarks will put you on a train to *credIT*, *ergo sum*. Bon voyage!

References

- [1] S. Berinato. Do the MATH. *CIO Magazine*, October 2001. Available via: www.cio.com/archive/100101/math.html.
- [2] R.L. Glass. *ComputingFailure.com – War Stories from the Electronic Revolution*. Prentice Hall, 2001.
- [3] P.X. Harder. A Conversation with Dr. Harry Markowitz. *Gantthead.com*, May 21 2002. Purchase via: www.eitforum.com/read.asp?ItemID=1157.
- [4] C. Jones. *Software Assessments, Benchmarks, and Best Practices*. Information Technology Series. Addison-Wesley, 2000.
- [5] F.W. McFarlan. Portfolio approach to information systems. *Harvard Business Review*, 59(5):142–150, September – October 1981.
- [6] The Bureau of Labor Statistics. Computer Programmers. In *Occupational Outlook Handbook, 2002-03 Edition*, pages 166–169. Bureau of Labor Statistics, Chicago, USA, 2002.
- [7] D.R. Pitts. Metrics: Problem Solved? *Crosstalk: The Journal of Defense Software Engineering*, 1997. Available via: www.stsc.hill.af.mil/crosstalk/1997/dec/metrics.asp.
- [8] C. Verhoef. Quantitative IT Portfolio Management. *Science of Computer Programming*, 45(1):1–96, October 2002. Available via: www.cs.vu.nl/~x/ipm/ipm.pdf.