Accounting for Software Investments Evaluating Financial Value, Quality and Risk
# Table of Contents

Executive summary .................................................................................................................. 3

1. Introduction .......................................................................................................................... 5

2. The importance of valuing IT assets ...................................................................................... 6
   2.1. The fair value principle ...................................................................................................... 7
   2.2. How to calculate the estimated value added by software? ................................................ 9

3. Accounting for cost of software ........................................................................................... 12
   3.1. The cost structure of software ........................................................................................ 12
   3.2. Types of software .......................................................................................................... 13
   3.3. IFRS regarding IT assets: Literature review .................................................................. 13
   3.4. Accounting for cost of software in practice .................................................................. 15

4. The importance of assessing software quality ....................................................................... 17
   4.1. Quality assessment in practice ..................................................................................... 19

5. Technical Debt ..................................................................................................................... 21

6. IT Risk Management ........................................................................................................... 22

7. Conclusion ............................................................................................................................. 25

Bibliography ............................................................................................................................. 27

Annex ........................................................................................................................................ 29

   Oral interviews ...................................................................................................................... 29

   Online interviews .................................................................................................................. 31

Literature review ....................................................................................................................... 34
Executive summary

The purpose of this study is dealing with the issue of IT management within businesses. The first section explains the importance of valuing the IT asset. In fact, an IT asset can be valued by taking into account its fair value. The purpose of fair value is to provide accurate information on the value of assets and liabilities of the company. Fair value is also very important in the process of decision making; over the years, when it comes to IT, it gained more and more weight given that the budget for IT increased in most companies. In practice, this notion is not always easy to capture.

The second chapter explains in detail how the cost of software should be accounted for in financial statements and what is done in practice. The concept of fair value exists in Belgium GAAP since 1975 and is more and more present in the international accounting standards (IAS and IFRS). Therefore, software cost could be capitalized when the entity believes that future economic benefit deriving from it will flow to the company. Besides, capitalizing costs reduces information asymmetry between management and stakeholders. However, in practice, we observe that a lot of companies tend to expense all their IT costs for the sake of simplicity or some fiscal advantages.

The third chapter explains the importance of assessing software quality. In Cost Management, many argue that increasing software quality can significantly reduce costs. A high quality application will require less time and money spent for maintenance and critical changes in response to business needs. In all, it reduces the Total Cost of Ownership. Good quality can also mean better software performance, which in turn can translate into higher employee and business process performance, and therefore better margins. However, more often than not, organizations don't have a detailed inventory of their software and they do not perform an appropriate strategic application life cycle planning. In consequence, they are not aware of the value of their IT, or the scale of IT debt, costs and risks that follow. When developing an application (in-house or outsourced), both its functional and structural quality, as well as the risks it may generate should be considered, and not only the time and budget needed.
Finally, the technical debt is "the effort required fixing violations of good architectural and coding practices that remain in the code when an application is released". Having a high technical debt can bring risks to the company. When it comes to IT risks, companies are generally more reactive rather than pro-active. If risks are not well assessed or if they are overlooked, this can mislead decision makers into choosing an IT project that would raise more costs and issues than the benefits it eventually bring. Besides, the riskier a project is, the more it can become costly for the company in the future.
1. Introduction

The main objective of IT governance is making sure that its management is well aligned with the company's global strategy. This concept allows us to view the IT department as an entire activity which can generate value for the company just as any other department.

IT can be defined as a set of technical (hardware and software), organizational and human resources which can serve to the processing, storage and transfer of information needed for the business activity. It is often considered as organization's largest cost generator, while its profitability still remains difficult to capture.

It has been demonstrated than an adequate IT management can contribute in a significant manner at creating additional business value, depending on the sector in which it operates. In addition, given the context of the current crisis, good IT management becomes fundamental in reducing IT service costs.

Therefore, we believe it is important to assess the financial value of a company's software. For the last couple of years, we have seen that IT related costs have been maintaining an upward trend in all companies. For certain types of businesses, this cost may even reach 50% of their annual expenditures. Under these conditions, what can we say about companies whose IT infrastructure is their main asset, and whose market capitalization is very important in the business world of today (such as Microsoft, Ebay and Facebook)?

However, the emergence of IT is relatively new in business. Therefore, its management is not always optimal.

The purpose of this study is dealing with the issue of IT management within businesses. The first section explains the importance of valuing the IT asset. The following treats accounting rules for software on the international GAAP. Finally, the last sections explain how companies assess quality and deal with their IT risks.
2. The importance of valuing IT assets

As we can see in the following image, IT has become an important investment for a company; it can even differentiate a firm from its competitors.

For any other asset within a company, it may be inconceivable not to value it. Therefore we should do the same for IT assets.

Consider the following: how long could a company operate if it were to completely shut down its IT infrastructure? In the case of IT based companies, only for a few seconds; for the rest, it will never exceed few hours.

Moreover, we are not looking for an accurate indicator to describe the value of a computer program; the challenge we are facing consists in finding the most relevant indicator based on all the elements we can provide.
When it comes to IT, the cost approach offers us limited use. The problem is that we need to know the value it could bring us in the future and we also need to be able to approximate its rate of depreciation.

Taking into consideration the various requirements of computer software, it quickly becomes impossible or difficult to compare IT value, based on market price. It can also be the case that a similar software is sold to different customers at different prices, because once software development is finished, its variable cost is virtually zero (we cannot use the cost of purchase as the most appropriate description of its true value).²

In conclusion, we need to find a new method to better approximate the value of IT. The one which we think would be the most appropriate is the fair value.

2.1. The fair value principle

The purpose of fair value is to provide accurate information on the value of assets and liabilities of the company; this information then becomes available to its shareholders and any other person in contact with the company. Furthermore, the continuous development of stock exchange resulted in increasingly sophisticated requirements for information production. Indeed, the rational investor of today needs access to the best and most accurate information before he can make investment decisions.²

The concept of fair value was introduced in the international accounting standards of IAS and IFRS, while it already existed in Belgium since 1975. The question remains: how can we adapt fair value to IT?

Taking into consideration the fact that it’s the first IAS standard (IAS 1), fair representation is an important part of the standards of IAS.

---

For a company, IT represents a service, a function and specific support; these could all be considered intangible assets of the company.

A computer program also meets all the requirements of an intangible asset.

- IAS 38R:8: Intangible assets are identifiable, nonmonetary items without physical substance.
- IAS38R:10: Criteria:
  o It is identifiable;
  o The entity has control over the asset;
  o It’s probable that economic benefits will flow to the entity;
  o The cost of the asset can be measured reliably (knowing the technical debt of the program could be an approximation of its potential costs).

However, we must keep in mind that computer programs can quickly become obsolete, either by technological or functional obsolescence.

Nevertheless, in the case of IT we cannot limit ourselves to this definition of fair value: "The amount for which an asset could be exchanged between knowledgeable, willing parties in an arm’s-length transaction". The reason for this is that custom solutions are very specific in scope, therefore difficult or impossible to resell.

One of the problems of fair value at market price is that the market is not transparent, liquid or perfect, especially when it comes to software which needs to meet certain criteria. If only the market price would be taken into consideration, then the information would be biased, as it would be subjective. It is therefore necessary to search for more parameters before judging the value of IT. A different way to calculate the fair value of a program is by estimating: the quality, the cost and the added value brought by the software to the company.

There is an increasing trend which pushes companies to outsource their software development. This creates the need for an indicator which evaluates the quality of a program and its fair value (or the costs of development, troubleshooting, maintenance and

---

updates compared to the generated profit). This also applies to in-house software. Only after we have this more accurate fair value, we have the possibility to reevaluate the software (depreciate or appreciate).

In regard to the “income approach”, it is sometimes difficult to find the cash flows which are specifically related to a piece of software, especially in a very complex business; a more efficient evaluation could be made with RCN (replacement cost new).

Fair value is also very important in the process of decision making; over the years, when it comes to IT, it gained more and more weight given that the budget for IT increased in most companies.

It is also important because "the use of fair value rather than other measurement is usually justified by its ability to inform on future cash flows". This means that having an exact figure for the fair value is difficult when future cash flows are uncertain. Going back to software, there are many specific problems it may encounter, unknown at the time of development. We can fix this if we know the Total Quality Index and forecast the costs thanks to the Technical Debt:

- It makes the fair value easier to calculate and more relevant;
- Best decision making based on more information.

The fair value also allows managers to analyze in a relevant way the EVA (Estimated Value Added) of the company.

### 2.2. How to calculate the estimated value added by software?

Many companies use the following two methods:

The first one is Net Present Value (NPV) and the second one is the Business Case (where all the costs and benefits of a project are estimated).

---


As we can see on this graph, it is important to know the value of a project at all times. In fact, the net present value is not always positive for every project of a company. Therefore, it is possible that, after a specific period of time, projects may stop bringing enough value to the company (e.g. its maintenance costs might become too high or the technology behind too outdated).

The same problem arises in the case of business plans. As a matter of fact, many companies that we had the chance to meet claimed that they make a business plan before starting an IT project; the purpose is to know the value that will be brought to the company and be able to compare it with the costs. As one interviewee said "We begin a project only when it can bring value to the company".
However, few companies continue to update the business plan once the project is implemented. This can distort the value of the latter such as we can see in the image below.

It becomes important to regularly update the NPV and the Business Plan in order to be able to monitor and evaluate the IT of a company.

However, as we have seen, it is sometimes very difficult to measure the benefits of an application; this is the case because there are more complicated benefits to measure than just the financial ones.

Some interviewees are trying to find different ways to consider the value of IT: "They are based on estimates. What would it mean if we wouldn’t have this application? Or if we would have to replace it with a different application? How much would it cost if we had to do it manually?".

In the majority of companies surveyed, we have seen that the CIO did not have a complete inventory of its applications. The company should have something like this. Some CIOs have admitted that they do no inventory because they consider that it’s not their job and they work for other departments of the company.
This is a major problem, because neither the CIO (who manages the applications) nor the CFO (who allocates funds for development and purchase of programs) know what is left of the money spent in previous years. In such situations, it is quite impossible to calculate the value of an IT company.

To conclude, due to the increasing importance of IT assets within a business, companies should pay more attention to their assessment. From a legal perspective, the international accounting standards principles should be respected carefully. However, managers are not used to such practices.

3. Accounting for cost of software

3.1. The cost structure of software

There are different types of costs associated with software:

- Start up costs: considered as IT expenses, they include acquisition costs, implementation costs and development costs.
- Operating costs: also regarded as IT expenses, they regroup not only maintenance costs but all the recurring operating costs. It is important to assess the quality of software and the associated risks in order to have a better estimation of these costs.
- The cost of adapting the software to business process.
- Start up costs for users: more often than not, the time and cost of adapting users to new software are not considered in the total cost of the application.

Thus, all these costs have to be taken into account when calculating the TCO. However, usually, the CEOs and CFOs do not take them all into consideration. This can lead to an inadequate decision-making.
3.2. Types of software

We can categorize software into two groups:

1. Outsourced software.

Nowadays, many companies outsource more than 50% of their software development. This allows them to incur fewer costs and more time saving. However, there are additional costs that are not always considered by these companies, such as costs of quality control. Several additional issues must be taken into account by the CEOs, like how to measure the quality of software provided by an outsourcer. This is not easy if they do not have access to the source code.

Some of the companies we interviewed that outsource their software leave the structural quality assessment in the hands of their suppliers. They care more about the functional quality.

2. In-house developed software.

This allows the company to have more control over the software development and to make sure that it will be in line with the business processes and requirements. However, given that the development requires a lot of time and that the people involved are busy with developing and managing the architecture, the security etc, they usually do not have the time to further test the quality and to analyze the various types of costs. They go deeper analyzing the system only after a problem is detected. This results in additional unexpected costs.

For these two types of software, quality assessment is important for the CEO, because he is legally responsible for what happens in the company. It is mandatory for him to have extensive knowledge about risks and to report them to the shareholders.

3.3. IFRS regarding IT assets: Literature review

The previous chapter explains briefly the fair value principle for software and how firms can assess their IT assets. The purpose of this chapter is to explain in detail how the cost of software should be accounted for in financial statements and what is done in practice.
According to the international financial reporting standards (IFRS), software costs can be capitalized and classified as intangible assets.

However, it may be difficult to classify an asset as being either tangible or intangible. For example, in the case of computer software, it can be bundled with a computer machine and be regarded as an integral part of the hardware. Therefore the asset can be classified as property, plant and equipment (PP&E). When the software is considered to be separate from the hardware, it will be treated as an intangible asset. The “identifiability” criterion means that the asset has to be separable, or that it arises from contractual or other legal right.

If the intangible asset was acquired separately (not generated internally), the price of acquisition can reflect the future benefits which would flow to the entity. On the contrary, in the case of internally generated intangible assets, it may be hard to estimate their fair value; the reason is that there is no exchange transaction and those assets may be hard to distinguish from the rest of the business. The IAS38 explains the procedure that can be followed in the case of recognizing internally generated intangible assets. In fact the generation of the asset should be divided into two different phases: research and development. In the research phase, all expenditures will be treated as expenses at the moment they occur. In the case of the development phase, costs will be capitalized.

To recognize intangible assets the following characteristics have to be demonstrated:

- *Technical feasibility of completing the intangible asset so that it will be available for use or sale;*
- *The intention of the entity to complete the intangible assets and use or sell it*
- *Its ability to use or sell the intangible asset;*
- *How future economic benefits will probably be generated (the existence of a market for the output of intangible asset or the asset itself);*
- *Availability of technical, financial and other resources to complete the development and to use or sell the intangible asset;*
- *Its ability to measure reliably the expenditure attributable to the intangible assets during its development*.

Therefore, the amount reported in the balance sheet for the initial recognition of internally generated intangible assets will be all directly attributable costs necessary to create, produce and prepare the assets in order to be capable of operating in the manner intended by management. After the initial recognition, the company has to choose between the cost model and the revaluation model depending on its accounting policy.

In the cost model, the initial cost remains the same but the firm has the right to amortize and account for impairment losses on the asset. On the contrary, in the revaluation model, the intangible asset has to be carried out at the revalued amount which represents its fair value less accumulated depreciation at the revaluation date.

### 3.4. Accounting for cost of software in practice

Mohd (2005) analyzed the impact of SFAS n°86 (which allow the capitalization of certain software development costs) on information asymmetry in software and high-tech firms in the US, from 1986 to 1995. The Standards of Financial Accounting Standards (SFAS) are used by companies listed on the American Stock Exchange. He concludes that R&D expenditure capitalization (in this case the cost of computer software developments) can reduce information asymmetry between management and investors.

Many studies have demonstrated that companies which are R&D intensive tend to have more information asymmetry (Aboody and Lev (2000); Barth and Kasznit (1999); Bart et al (2001); Boone and Roman 2001). According to the literature, when firms capitalize their R&D, information is disclosed about the success of R&D activities. Therefore the information asymmetry between investors and managers decreases. Besides, a voluntary disclosure is not as informative as the recognition in the financial statement because:

---

8 Mohd (2005) « accounting for software development cost and information asymmetries » the accounting review vol80, No4, p1211-1231
• The financial statement is audited, the voluntary disclosure is not. Thus the recognition is more credible;
• It may be more costly to extract information from voluntary disclosure than recognition.

The voluntary disclosure can increase the information asymmetry due to the fact that investors with accounting expertise will collect information easier and at a lower cost than unsophisticated investors.

Kothari and Al. (2002) argued that in case of R&D expenditures, their capitalization reduces the uncertainty of future economic benefit (by capitalizing you have to prove that future benefits will flow to the entity).

However, some managers may expense R&D costs even if they have reliable estimates of their future benefits. To support the current literature, we interviewed several CIOs and CFOs on how they account for the cost of software in the companies where they work or used to work. All in all, most of the management tends to expense the cost of software instead of capitalizing it when required. The main reason given is simplicity and fiscal advantages. Some revealed to capitalize the cost of software only when using the IFRS GAAP (usually for consolidation purposes) but not for the social financial statement. However, some declared that some costs are capitalized and usually amortized on a 3 year plan.

The situation worsens when it comes to the public sector. It appears that public companies don’t have to disclose their financial statement. Therefore, they don’t feel the need to value their IT assets and account for them correctly.

The following chart shows the evolution of public income/expenditure in EU27 from 2008 to 2011.
Public expenditure is higher than public income, which has been creating a permanent deficit since 2008. Besides, in the first chapter, the first chart showed that IT public spending is comparable to the one found in the banking sector. However, the public sector doesn’t have the same IT maturity as the banking sector. Therefore the IT spending should be much lower.

No matter the size or type of business, the cost reduction remains an important concern. In our point of view, one way to reduce public expenditure would be by paying more attention when valuing them.

To conclude, according to the law, software cost can be capitalized if the entity believes that future economic benefit deriving from it will flow to the company. Besides, capitalizing cost reduces information asymmetry between management and stakeholders. However, in practice, we observe that a lot of companies tend to expense all their IT costs for the sake of simplicity and fiscal advantages.

4. The importance of assessing software quality

Software quality is a very important notion and it has been proven to have a great influence on risks as well as costs.
In Cost Management, many argue that increasing software quality can significantly reduce costs. A high quality application will require less time and money spent for maintenance and critical changes in response to business needs. In all, it reduces the Total Cost of Ownership. Good quality can also mean better software performance, which in turn can translate into higher employee and business process performance, and therefore better margins.

It is quite obvious that quality level is strongly related to Risk Management. Many problems are born from software of poor quality. We are going to discuss Risk Management in more detail in a future chapter.

In order to define software quality in a business context, there are two notions that should be taken into consideration: functional quality and structural quality. Functional quality refers to how well a computer program performs the functions it was designed for. For an easier understanding, it can answer the simple question of "What does it do?". On the other hand, structural quality, also called non-functional quality, technical quality or internal quality, refers to how well the underlying code was written in order to deliver its functional requirements and the degree to which it was engineered correctly. It could be the answer to "How does the application do something?".

There are different types of criteria for assessing an application's quality, depending on which of the two previously defined types is being used. The definition of these factors has to comply with a company's business goals, and therefore it is very likely that they vary from one organization to another. This can also lead to the perception that assessing software quality can be quite subjective.

Some of the most common factors used in assessing functional quality are: usability, usefulness and absence of fail.

Usability refers to how easy to use and intuitive an Information System is. This includes the ease of learning, how user-friendly the user interface (UI) is and how useful the error messages are. In all, it determines a certain level of user satisfaction and acceptance.

Usefulness refers to how adequate the application's functions are in order to enhance a specific job performance. This basically represents the extent to which the software's
functions cover the business requirements, in order to attain certain business goals. This is why defining the requirements prior to developing or buying software becomes crucial.

Testing an application (during or at the end of its development process) is usually done in order to verify or to prove the absence of fail. This is more often than not only done at a functional level. Even so, it is not possible, or it is too costly to test for all possible failures, especially if the testers don’t go deeper and also verify structural elements, besides the functional ones. Therefore, if no fails are found, it doesn’t mean that there are none. That is why it is important to assess not only the functional quality, but the structural one as well.

Among structural quality factors, we can cite: Security, Performance, Robustness, Transferability and Changeability (Figure 1).

<table>
<thead>
<tr>
<th>Health Factor</th>
<th>Description</th>
<th>Example business benefits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transferability</td>
<td>Attributes that allow new teams or members to quickly understand and work with an application</td>
<td>• Reduces inefficiency in transferring application work between teams&lt;br&gt; • Reduces learning curves&lt;br&gt; • Reduces lock-in to suppliers</td>
</tr>
<tr>
<td>Changeability</td>
<td>Attributes that make an application easier and quicker to modify</td>
<td>• Improves business agility in responding to markets or customers&lt;br&gt; • Reduces cost of ownership by reducing modification effort</td>
</tr>
<tr>
<td>Robustness</td>
<td>Attributes that affect the stability of the application and the likelihood of introducing defects when modifying it</td>
<td>• Improves availability of the business function or service&lt;br&gt; • Reduces risk of loss due to operational malfunction&lt;br&gt; • Reduces cost of application ownership by reducing rework</td>
</tr>
<tr>
<td>Performance</td>
<td>Attributes that affect the performance of an application</td>
<td>• Reduces risk of losing customers from poor service or response&lt;br&gt; • Improves productivity of those who use the application&lt;br&gt; • Increases speed of making decisions and providing information&lt;br&gt; • Improves ability to scale application to support business growth</td>
</tr>
<tr>
<td>Security</td>
<td>Attributes that affect an application’s ability to prevent unauthorized intrusions</td>
<td>• Improves protection of competitive information-based assets&lt;br&gt; • Reduces risk of loss in customer confidence or financial damages&lt;br&gt; • Improves compliance with security-related standards and mandates</td>
</tr>
</tbody>
</table>

Figure 1: Structural Quality Factors and their benefits to the business

4.1. Quality assessment in practice

IT projects are usually considered a success if they fit in the allocated time, budget, and if they help attain the objectives they were implemented for. In our empirical study we found

---

out that many failures arise from business objectives/requirements related faults. All our interviewees admitted that, during their career, they have seen a rather large number of IT projects fail. Most of them fail to be completed on budget and/or on time, but this doesn't usually mean that they are considered failures in the end. A much more serious reason is having a result that does not comply with the business/client needs and objectives.

The CIOs explain that the latter can be caused by a defective definition of requirements. These requirements may often be incorrectly "translated" into computer functions, since it is usually hard to establish a good communication between business men and computer programmers. Therefore, IT professionals have to make sure that the requirements they received are clear enough, and business professionals have to make sure that the requirements they defined are well aligned with the company strategy and goals. Other situations may arise, where the definition is incomplete at the start of the project, and the requirements change and multiply during the development process. This is seen even more often in state enterprises, because their actions depend on decisions taken at government level and on changes in law.

This way we can understand the importance of factors which lead to a high functional quality.

We explained before what software quality means and why we believe it is important for business. Nevertheless, from our interviews we concluded that most CxOs don't take it enough into consideration.

For example, one CxO from an international company considers that it is enough for them to test the quality of software only at a functional level. He explains that, since most or even all the software is outsourced, it is the provider's job to test the structural quality, considering they have the resources and various international standards for doing it. And so, they don't control how the supplier works, they only decide if they are satisfied with the provided service or product. This point of view is shared by a CxO from one state enterprise, but even if they don't do the tests themselves, they at least go a bit further by asking more and more often to see the test reports done by the developers.
We do not agree with this practice of only testing for functional quality and relying mostly on absence of fail. For one, absence of fail during the life of a program from its conception and until a certain moment in time does not mean absence of fail for the rest of its remaining life, it only means looking back and not at the future. That's why it doesn't take into consideration possible future risks and problems; it is only concerned with what has already happened. Secondly, in order to better manage costs and risks, it is important to evaluate and monitor structural quality as well, as these defects are likely to cause operational problems, and they are usually not detected through regular testing. By doing so, issues can be discovered and fixed before they become actual problems.

We have seen during our empirical study that other companies pay more attention to the quality of their IT. For example, another state enterprise CxO we talked with uses an external application analyzer to assess the structural quality of its in-house and outsourced software. If the software doesn't meet a certain level of quality predefined in a contract, the company can send it back and ask for improvements. Another interviewee states that they do regular screenings for their developers, various mandatory tests (application test, acceptance test, integration test), as well as intrusion tests. They can be rather costly, but given the risks they help avoid, these tests are worth it.

5. Technical Debt

We believe that more often than not, organizations don't have a detailed inventory of their software and they do not perform an appropriate strategic application life cycle planning. In consequence, they are not aware of the value of their IT, or the scale of IT debt, costs and risks that follow. This was also confirmed by our interviews.

Companies all over the world gain more and more knowledge and concern about their IT department's technical debt. This notion is defined as "the effort required fixing violations of good architectural and coding practices that remain in the code when an application is released. Technical Debt is calculated only on violations that the organization intends to remediate". It has been estimated as an average of $3.61 of debt per line of code, which

10 The CRASH Report - 2011/12 (CAST Report on Application Software Health) - Summary of Key Findings
would amount to more than 3 million dollars for many regular applications. This number should raise concern and increase the number of CxOs willing to take it into consideration.

In order to calculate software's technical debt, the source code has to be analyzed, in order to determine the density of coding violations per thousand lines of code (KLOC). We can understand here once again the importance of assessing and monitoring the software's structural quality.

The purpose of technical debt is that of an indicator, which can aid in developing an action plan for a company's five most mission-critical applications. A decision can be made about how much of this debt can and should be eliminated before the application is released, and how much is still acceptable, based on the marginal return on business value. It can also be seen as money that didn't have to be spent. If companies spend a little more in the pre-production phase, they save money by not having to fix problems afterwards. If that were the case, they could allocate that amount to increase innovation rather than spend it on maintenance.

6. IT Risk Management

As we've discussed before, it is essential to measure software quality, in order to:

- Have better visibility on (critical) applications quality;
- Monitor the work done in-house or by an outsourcer (or even for the outsourcer to prove the quality of the delivered product);
- Carry out the most objective benchmarking possible;
- Reduce IT risks.

Good IT governance implies assessing, monitoring and reducing IT risks. These are also important steps during the creation of a business case. Various types of IT risks may arise:

- Risks related to architecture: due to inadequate implementation and/or the difficulty to connect multiple applications;
- Risks related to software: due to poor programming and/or poor software testing;
• Risks associated with lack of security: which can lead to malicious actions like changing or destroying data, disclosing confidential information, etc.;
• Risks related to process: due to IS misuse, underestimation of software complexity, difficulty to adapt the software to the company's process and way of working, and/or lack of enough time for proper testing due to "severe" deadlines.

Most IT risks don't arise from technical issues, but rather from faulty company and IT governance.

These risks can generate serious repercussions, like:

• Financial repercussions: additional costs for reconstruction of databases in case of lost or altered data, customer compensations, possible loss of business transactions during a certain time etc.
• Repercussions for the company's brand image: loss of clients' confidence.

There are various steps for managing IT risks:

• Risk assessment: assess the level of software quality and the risks it may generate; assess project, maintenance, architecture risks and the repercussions they may give rise to, take past and present bugs and issues into consideration etc. The company must perform various necessary tests to ensure the good development and operation of its information system. If it is poorly developed, negative consequences may appear anytime over the rest of the software's life.
• Select the most important risks and choose a method to manage them: disaster recovery plans, business continuity plans, making provisions for IT risks etc.

When developing an application (in-house or outsourced), both its functional and structural quality, as well as the risks it may generate should be considered, and not only the time and budget needed. As we mentioned before, the riskier a project is, the more it can become costly for the company in the future. If risks are not well assessed or if they are overlooked, this can mislead decision makers into choosing an IT project that could raise more costs and issues than the benefits it could bring. As we can see on the next image, there is a correlation between costs and risks: the more a project is risky, the higher the cost. Even if the project can bring a high value, it is less likely to be recommended than a project
that has fewer risks. On the other hand, it is more likely for the project to be rejected if the decision makers are well aware that it includes high risks.

Unfortunately, many CxOs are not concerned at all with verifying the quality of their software (especially if it is not developed in-house), or are just limited at checking the quality and the test reports provided by their suppliers. Furthermore, they don't always have good IT risk governance. Our empirical study confirms this. Nevertheless, given all we mentioned above, we believe that it would be in their company's best interest to assess and monitor the functional and the structural quality of their software, as well as their IT risks.

Currently, when it comes to IT risks, companies are generally more reactive rather than pro-active. During our interviews, some CxOs admit that they mostly repair rather than prevent various issues, and that they don't make use of proper IT risk management. Moreover, many of them are subject to frequent bugs.

One of our interviewees, who works in an state enterprise explained that, in order to limit the risk of data loss, they make copies of their data and they place them in a different data warehouse. Apart from this practice, the company doesn't really measure or quantify the IT risks; it is not very mature in terms of IT risk management. We believe that what they are doing is not enough, because issues may arise in both data warehouses. For example, last year, Amazon has experienced a failure on its platform, Elastic Compute Cloud (EC2),
due to human error. This has disrupted many customer sites such as Foursquare, Quora and Reddit for several days.\textsuperscript{11}

Another CxO says that his company outsources everything when it comes to IT and they don’t control the quality of the applications received from their suppliers. They don’t consider useful or important knowing how the software was developed or any other detail regarding its source code, as long as it has the required functionalities and it seems to work well.

However, some companies have better IT risk management. One of the CxOs we talked to states that they sign Service Level Agreements with their outsourcers, in order to define various quality criteria and levels for the developed software. They also issue disaster recovery plans and they have a unit responsible for business continuity planning.

Another CxO also affirms that they pay great attention to IT risks. They have a disaster recovery plan and they do make a "risk assessment" for each project in order to identify project risks, risks related to maintenance, software suppliers, as well as organization risks. After this, they decide what to do to avoid them and what this might cost.

7. Conclusion

During our study, we have seen that the part of IT becomes increasingly important in every type of business. We also noticed that decision makers become more and more preoccupied about the value and the risks of IT. Despite this, they are still not optimally managed and numerous improvements are still possible in this area, in order to treat these assets as any other types of assets present within a company. This could allow the firm to anticipate future IT requirements and have a better idea of the value of their business.

The evaluation of IT at its fair value also provides better information to shareholders of the company. Indeed, they are more accurately informed about the real value of the company.

\textsuperscript{11}Zdnet France (2011) « Amazon présente ses excuses pour la panne de son service cloud » http://www.zdnet.fr/actualites/amazon-presente-ses-excuses-pour-la-panne-de-son-service-cloud-39760378.htm, accessed 03/05/2012
This becomes today more necessary since market participants are increasingly seeking information and it also becomes a legal obligation. Indeed, the fair value had already appeared for many years in some national GAAP, but is now becoming increasingly present in major accounting standards such as IAS and IFRS.

For this, the IT department should also include indicators relating to the structural quality, not just the functional one, as well as business risk management driven by the IT, given that many organizations could not fulfill their mission/corporate purpose for more than a few hours without their IT systems.
Bibliography


Carpentier J. (2010) « la gouvernance du système d’information dans les PME pratiques et evolutions » ENI


Hand, M. « Discussion of “accounting for software development cost of capital: evidence from IPO underpricing in the software industry” »


Zdnet France (2011) « Amazon présente ses excuses pour la panne de son service cloud »
http://www.zdnet.fr/actualites/amazon-presente-ses-excuses-pour-la-panne-de-son-service-cloud-39760378.htm, accessed 03/05/2012

Annex

Oral interviews

Case 1

For the first case, we interviewed a CIO working in the public sector that processes rather confidential data. During this interview, we learned that most of their applications were developed externally, and that quality control of the source code was always made upon receipt. For example, a certain level of quality is required when they place an order for a new application. This quality control allows the company to correct major programming errors before putting the application in use. Other parameters are taken into account for measuring quality, like user acceptance or application performance.

The IT budget in not really a problem for this company because, in the public sector, a certain fixed budget is allocated.

Surprisingly, we also learned that the company was not very sensitive to IT risk. Most of the time, they work reactively rather than defensively. The only risk taken into account is the one linked to data loss. For this risk, they have a second database hosted in another building.

The value of IT was not really taken into account because of the nature of the business, for which the true value created is the service provided to citizens. Moreover, we also learned that the CIO did not have a complete inventory of its main applications.

Case 2

For the second case, we met a CIO from a commodities trading company. The part of IT in this activity is very important. In this company, every decision regarding IT is taken based on a business case, which provides an estimation of the costs and the benefits of the application. These estimations are often made thanks to experience and knowledge from past projects.

We were learned that the company is highly preventive when it comes to IT risks. They do a risk assessment on a regular basis for each major application. They also conduct an annual Enterprise Risk Management and they have disaster recovery plans.
Regarding quality, the company has numerous procedures for measuring it. Among them we can mention application testing, acceptance, integration, and also intrusion tests from independent companies.

When it comes to IT value, the CIO always refers to a complete and up to date business plan. The application inventory is also updated regularly.

Case 3

For the third case, we met a CIO of a state company. As in the first case, the vast majority of applications are outsourced. We learned that the CIO considers that quality tests at source code level should be made by the supplier. For the functional quality, tests are performed by the company itself, including acceptance and integration test.

Regarding the value of IT, it is now measured by building business cases, because the IT budget has decreased since the beginning of the crisis. As in case 1, the goal here is not necessarily measuring the value that IT brings to the company, but rather the service provided to the citizens.

The company is aware of its IT risks because they are very important for its activity. They monitor the applications and a special department is responsible for business continuity planning. There also have disaster recovery plans and a second database. An inventory of applications exists, but the CIO has doubts about its completeness.

Case 4

For the fourth case, we met a CIO working in a financial firm. Application quality is assessed by looking at the number of days it was available during the year. The company is aware of the risks of IT and develops defensive coding.

For the cost of developing an application, they rely on the experts estimates, which are based on the costs of other similar software. The company has an IT inventory, but the CIO admits that it is not complete.

Case 5

For the fifth case, we met a CFO in a company for new technologies. The part of IT is very important here. The company protects itself with insurance covering IT risks for the business.
They do not account for reserves and provisions in the balance sheet, because the probability of an unwanted consequence in IT is not big enough. IT software depreciation is usually accounted for in the balance sheet on the pre-established life of the application, and it generally does not exceed 5 years.

Before making a decision, the CFO always takes into consideration the relationship between costs and benefits of an IT project, even if these benefits are not quantifiable.

**Case 6**

For the sixth case, we met a CIO in a chemical company. He told us that none of their software is developed in-house, and that their IT department was only responsible for the architecture and relationships between applications. The quality of applications is assessed and managed by suppliers. The company trusts them to do a good job, since they also supply other big companies and have done so for many years. But if a problem should arise, the company would be covered by a guarantee agreement on the application that is regularly renewed with the supplier.

IT projects are launched only if they promise a certain return. In terms of IT risk, the company regularly conducts tests on its architecture in order to avoid them as much as possible.

**Online interviews**

**Case 1**

Case 1 is an IT company which has developed a logistics technology platform. This technology allows businesses to connect in a cost-efficient manner, thanks to the global logistics network. According to the interview, the company uses US GAAP for accounting for software; however, they barely estimates the total cost of ownership of their software. Besides, the company declared not carrying out a regular inventory because of lack of time, except if it is requested by auditors.

Concerning the IT risks, this firm doesn’t have a precise method to estimate them. They consider that only three types of IT risks matter:

- Risks related to infrastructure needs
- Security risks
• Scalability risks

The security risks are managed by CISO and security tools manage the various aspects of security. Besides, in order to measure the quality of outsourced software, they usually test the functional quality and not the structural one.

Case 2

This company operates in financial sector. They seem to assess the value of IT better than the previous company. In fact, in order to value an IT asset, they first build business cases which evaluate the potential return of application versus the investment and operational expenses made. More precisely, they do an estimation of ROI or payback period based on business benefits, evaluated by business lines and IT investment/operational cost (as defined in supplier's proposal). Finally, the TCO is quantified, mainly for accounting purposes.

They declared doing a regular inventory of their applications. The inventory is used as an input in the elaboration of the IT delivery plan (tactical IT plan). This methodology is used to evaluate key projects on a three years time frame and requires the knowledge of current application map (component).

They mainly outsource their software development. Therefore, to measure the outsourced software quality, they use SLAs and SLRs which are part of the contract with the outsourcing company. Actual figures are reviewed monthly by the Service Delivery Manager. In addition, they use acceptance testing and control of supplier's conformity to coding-testing requirements.

Risk sources are mainly related to e-commerce and potential intrusion and hacking. Impact depends on application. They have a system which tracks IT incidents and report them to the operational risk management and the IT department.

However, there is no follow up of the business cases to test whether a project was a success after implementation.
Case 3

This case is a telecommunication company. The software inventory is done from a financial perspective, to appreciate or write-off the assets of the company. After re-organization, they can re-allocate applications between each new department.

In order to value IT costs, the firm has various metrics and standard methodology, named TTM (Time to Market), which help evaluating the complexity, cost and duration of a project. Future benefits can be calculated by the marketing department and included in the business plan, for validation by finance department. There is no real systematic calculation of the TCO. They have a yearly budget approach; the TCO is per project;

“The value of a development is the sum of the purchase orders done towards the suppliers of the solution, plus a quota of the activity of the M* people previous experiences”.

The applications cost is calculated through the Purchase Orders done for the suppliers. Besides, the amortization is done traditionally.

To assess the quality of outsourced software, traditional acceptance tests are used, before the move-to-production. SLA/KPI/Financial penalties are used after move-to-production. Moreover, they usually measure the quality of the delivered code.

The major risks that the company faces are the obsolescence of some solutions and performance failures for others. The firm has a «risk committee» that evaluates the different risks and proposes mitigation plans per risk. Those risks are not managed on a daily base, but on mid and long term.

Case 4

This insurance firm carries out a regular inventory to remove depreciated applications and reduce cost base accordingly. Since more than 75% of software is outsourced, the quality tests are done by analyzing the outsourcer’s interface, and taking into account various quality standards.
Case 4

We already interviewed them orally, see the oral interviews section.

Literature review

The success of companies like Microsoft and Amazon are examples which allow us to understand why the quality of software is important and why their creators can earn from them. There are also many companies, even outside the field of IT, where information technology plays an important role (e.g. the steel industry); in these cases, errors in critical pieces of software would translate into big losses for the companies. It is therefore important to monitor all the software and check its quality as regularly as possible.

IT Governance and IT risk

The main objective of IT governance is making sure that its management is well aligned with the company's global strategy. This concept allows us to look at the IT department as an entire activity which can generate value for the company just as the other departments.

IT can be defined as a set of technical (hardware and software), organizational and human resources which can serve to the processing, storage and transfer of information needed for the business activity. It is often considered an organization's largest cost generator, while its profitability still remains difficult to capture.

It has been demonstrated than an adequate IT management can contribute in a significant manner in creating additional business value, depending on the sector in which it operates. In addition, given the context of the current crisis, good IT management becomes fundamental in reducing IT services costs.

Sometimes there are not enough indicators that can be used in IT management; for example, computer expenditures are often documented globally and not analytically in the balance sheet. Furthermore, these indicators do not allow comparisons between organizations. Besides, they often explain the IT governance in terms of costs and not in terms of the value created for the company.
Governance methods

IT governance uses several methods which mainly aim to reduce operational risks. The list is not exhaustive, but in the context of our work, we mention three of the most interesting methods.

**CMMI model**

With regard to information system development, the CMMI (Capability Maturity Model Integration) is a model that was initially created by the United States Department of Defense to monitor developments and the associated budgets. Afterwards, its usage was extended to any enterprise whose core business is developing IT applications. The model's main purpose is to ensure a good IT management. In fact, it provides best practices in order to be able to comply with the given deadlines, required functionalities as well as the budget. It analyses the business' level of maturity and suggests standards for continuous improvement or PDCA (Plan-Do-Check-Act).

This methodology ensures a better cost management as well as a better quality in terms of software development. In addition, it helps to better anticipate the real project costs. Finally, it carries out a comparison between the performance and the budget of project teams (internal and external), which provides a better justification of expenditures, and therefore can lead to improving the processes or applications.

**ITIL Model**

ITIL (Information Technology Infrastructure Library) offers a structured library of good practices on how to manage an IT system. This method allows the company to better understand its IT and better integrate it with the rest of the business.

The new version of ITIL has three components:

- The core, which covers the information system management, given the business' needs.
- The secondary components that deal with specific cases or situations.
- The web component, which provides practical examples such as models, process mapping and documentation, all of them useful resources for aiding in the process of IT management.
This library includes several services, like the support service, which ensures operation and technical support. It also manages incidents, problems and changes, according to business needs. When it comes to the security service, all the ITIL processes are based on ISO 27001 norms, which deal with the information security policy within a company. The IT infrastructure service takes care of the network, operations, management of local resources, computer installation and acceptance, and system management. This library in particular combines a set of financial management tools that help in analyzing the IT department's functions and processes, while referring to its accounting and its fiscal value. This service provides the company as well as the IT department with a quantified value, in financial terms, of the services provided by the information system.

**COBIT Model**

Finally, for controlling and organizing the information systems, the COBIT system (Common Objectives for Business Information Technology) audits the organizations' IT activity. This model is most commonly used in big companies as it is intended to provide assistance to corporate leaders in controlling their investments, in order to better manage risk and meet their obligations with respect to different actors. Moreover, it also concerns the IT executives and auditors.

Financial analysis of computer services

It is difficult to evaluate all the expenditures related to the IT services, as their budgets have a lack of precision and are often under evaluated, as all parameters are not necessarily taken into account.

In general, the business IT consists of an infrastructure that includes workstation networks, servers, routers, WAN equipment, software, user support, maintenance and operations. Thus, this infrastructure represents an important fraction of the IT budget. The other part goes towards software expenditures as well as human resources needed for project development, operation and maintenance.

In order to establish the IT budget, it is essential to examine its cost management process. The maintenance and human resources costs can easily represent a third of the total project
costs. But usually, the negotiation of the IT budget is mainly based on IT investments and project costs. Nevertheless, if the maintenance and operating costs are not properly anticipated, they can take up the total of the IT budget.

**Cutting down IT costs**

No matter the size or type of business, the reduction of the IT cost remains an important concern. Thus, it can be done in two areas:

- Asset management and pooling of resources
- The choice between in-house or outsourcing

For a good cost management, it is critical to know how IT assets are used and how compatible they are with the needs of the organization. For example, an indicator such as the percentage of equipment usage must be taken into account. For that, the first step is to conduct an inventory. With time, the company may remove or recycle equipment deemed unproductive or obsolete. Likewise, when a user closes an account, the company can gain some storage space. Nevertheless, a comprehensive audit is necessary in order to verify that the installed applications are correctly and efficiently used. The organization can also analyze key applications that need complex infrastructures in order to find less costly solutions. In general, cutting costs can also be done in the following manner:

- Centralizing purchases and reducing the number of suppliers
- Optional training for users on consequences of their demands on the budget
- Workplace management
- Global vision over IT budgets and accounting reports
- Extending the life of IT assets
- Establishing accounting indicators for costs management (TCO)
- Outsourcing

**IT risks and governance**
IT governance must take into account certain risks. IT risks can lead to adverse effects on IT assets or the entire business.

IT allows the consulting of databases by a huge number of users within or outside the organization. Having a large IT database makes sharing and duplicating easier and thus more vulnerable than having a physical database like in the old days. Moreover, thanks to communication networks, the various IT systems from different locations are interconnected, which enhances the risks. For example, issues such as non-authorized access, misuse, fraud can be spread quickly through the network.

IT systems components come from different sources (company, suppliers, software makers etc.). They are supposed to evolve regularly, and these changes can affect the IT system’s robustness, security and balance. The lack of security can cause other problems. For example, valuable files can be intercepted while moving within the network. Moreover, hackers can destroy files or messages by introducing malicious software or by initiating other types of attacks.

The system can also malfunction due to a crash in the equipment, inadequate configuration or damages caused by misuse. Programming errors, bad installation or unauthorized changes can cause bugs in the software.

Despite the risks related to the IT system’s architecture, other incidents can occur. In fact, Comair, a Delta Airlines subsidiary, had an IT incident on December 24 2004, when the crew scheduling application crashed. Comair has neglected to change the scheduling system because the likelihood of failure was low. As a consequence, they experienced revenue losses of about $20 million, in addition to damaging their reputation. Another example comes from Foxmeyer in 1996, a pharmaceutical company who failed to implement the SAP’s enterprise resource planning software and went bankrupt.

IT incidents have proven to have three factors in common:

- An IT system failure can harm the whole company and its entire network
- The incidents often involve public disclosure of confidential information, which can enhance the gravity of the consequences
The IT risks can arise from the failure of the general management and not only from the fault of the IT management. A company that invests a lot in IT automatically increases IT risks because it depends more on those IT processes. Nevertheless, IT risks are still regarded as technical issues in many companies, and they are still handled at the IT department level and neglected by business executives. Actually, most IT risks are not technical; rather they arise from the company's failure to properly govern the IT processes. Unhealthy decision making is the major cause of IT risks. Therefore, general management and IT management should work together for the better understanding of IT risks, which are part of the global business risks.

References:

Carpentier J. “La gouvernance du système d’information dans les PME pratiques et evolutions” ENI


Fair Value

One of the greatest challenges of our work was not in finding an accurate indicator to describe the value of a computer program, but in finding one which is the most relevant based on all the elements we can provide.

The cost approach offers us limited use in IT, because when it comes to software, we need to know the value it will bring us in the future and we also need to be able to approximate its rate of depreciation.

Comparing the value of IT using the market price is in most cases impossible or difficult to implement because of the various requirements of computer software. Moreover, it often happens that a similar computer program is sold to different customers at different prices,
because once the development of the software is finished its variable cost is virtually zero (we cannot take the cost of purchase as the most appropriate description of its true value)\textsuperscript{12}.

In conclusion, we need to find a new method to better approximate the value of IT. The one which would be the most appropriate is the fair value.

The purpose of the fair value is to provide accurate information on the value of assets and liabilities of the company to its shareholders and any other person in contact with the company. Furthermore, the continuous development of stock exchange resulted in increasingly sophisticated requirements for information production. Indeed, the rational investor of today needs access to the best and most accurate information before he makes investment decisions\textsuperscript{13}.

The concept of fair value was introduced in the international accounting standards of IAS and IFRS, while it already existed in Belgium since 1975. The question remains: how can we adapt this to IT?

The fair representation is an important part of the IAS standards considering it’s the first IAS standard (IAS 1: Fair representation)\textsuperscript{14}.

From an IT perspective, the company possesses a service, a function and specific support; these could all be considered intangible assets of the company.

Moreover, a computer program meets all the requirements of an intangible asset.

- IAS 38R:8: Intangibles assets are identifiable, nonmonetary items without physical substance.
  IAS38R:10: Criteria:
  - It’s identifiable
  - The entity has control over the asset
  - It’s probable that economic benefits will flow to the entity

- The cost of the asset can be measured reliably (knowing the technical debt of the program could be an approximation of its potential costs).

However, we must keep in mind that computer programs can quickly become obsolete, either by technological or functional obsolescence.

Nevertheless, in the case of IT we cannot limit ourselves to this definition of fair value: "The amount for which an asset could be exchanged between knowledgeable, willing parties in an arm’s-length transaction". The reason for this is that they are very specific, therefore difficult to exchange.

One of the problems with fair value at market price is that the market is not transparent, liquid or perfect, especially when it comes to software which needs to meet certain criteria\(^\text{15}\). A different way to calculate the fair value of a program is by estimating: the quality, the cost and the added value brought by the software to the company.

If only the market price would be taken into consideration, then the information would be biased, as it would be subjective. It is therefore necessary to search for more parameters before judging the value of IT.

The software of today is usually outsourced. This creates the need for an indicator which evaluates the quality of a program and its fair value (or the costs or troubleshooting, maintenance and updates compared to the generated profit). Taking this into consideration, in-house software should be evaluated in the same way.

After estimating the quality of a program and its future costs, we could also reevaluate the asset (depreciate or appreciate).

In regard to the “income approach” it is sometimes difficult to find the cash flows which are specifically related to a piece of software especially in a very complex business; a more efficient evaluation would be made by “replacement cost new”.

Fair value is also very important in the process of decision making; over the years, when it comes to IT, it gained more and more weight given that the budget for IT increased in most companies.

It is also important because "the use of fair value rather than other measurement is usually justified by its ability to inform on future cash flows"\(^{16}\). This means that having an exact figure for the fair value is difficult when future cash flows are uncertain. Going back to software, there are many specific problems it may encounter, unknown at the time of development (though, we can fix this if we know the Total Quality Index and forecast the costs thanks to the Technical Debt):

- It makes the fair value easier to calculate and more relevant!!
- Best decision making based on more information!!

The fair value also allows managers to analyze in a relevant way the EVA of the company\(^{17}\).

**Structural quality and the technical debt**

More often than not, organizations don't have a detailed inventory of their software and they do not perform an appropriate strategic application life cycle planning. In consequence, they are not aware of the value of their IT, or the scale of IT debt, costs and risks that follow\(^{18}\).

Companies all over the world gain more and more knowledge and concern about their IT department's technical debt. This notion is defined by CAST as "the effort required fixing violations of good architectural and coding practices that remain in the code when an application is released. Technical Debt is calculated only on violations that the organization intends to remediate"\(^{19}\). They estimated an average of $3.61 of debt per line of code, which would amount to more than 3 million dollars for many regular applications. This number should raise concern and increase the number of CxOs willing to take it into consideration.


\(^{18}\) Andy Kyte, Measure and Manage Your IT Debt, Gartner Inc., RAS Core Research 9 August 2010

\(^{19}\) *The CRASH Report - 2011/12 (CAST Report on Application Software Health) - Summary of Key Findings*
In order to calculate technical debt, CAST analyses the source code of an application, using the CAST Application Intelligence Platform (AIP), in order to evaluate the structural quality of software and, in other words, to determine the density of coding violations per thousand lines of code (KLOC). According to their "CRASH Report", structural quality is "the non-functional quality of a software application that indicates how well the code is written from an engineering perspective. It is sometimes referred to as technical quality or internal quality, and represents the extent to which the application is free from violations of good architectural or coding practices." It can be better understood as "how the application does something", rather than "what it does". A violation refers to "a structure in the source code that is inconsistent with good architectural or coding practices and has proven to cause problems that affect either the cost or the risk profile of an application".

There are 5 structural quality characteristics that CAST uses to highlight coding issues: Security, Performance, Robustness, Transferability and Changeability (Figure 1). From these, a composite score is computed for each application, the Total Quality Index. These five are also called Health Factors, as they have a direct impact on business productivity, customer experience and IT costs.

<table>
<thead>
<tr>
<th>Health Factor</th>
<th>Description</th>
<th>Example business benefits</th>
</tr>
</thead>
</table>
| Transferability | Attributes that allow new teams or members to quickly understand and work with an application | • Reduces inefficiency in transferring application work between teams  
• Reduces learning curves  
• Reduces lock-in to suppliers |
| Changeability | Attributes that make an application easier and quicker to modify | • Improves business agility in responding to markets or customers  
• Reduces cost of ownership by reducing modification effort |
| Robustness | Attributes that affect the stability of the application and the likelihood of introducing defects when modifying it | • Improves availability of the business function or service  
• Reduces risk of loss due to operational malfunction  
• Reduces cost of application ownership by reducing rework |
| Performance | Attributes that affect the performance of an application | • Reduces risk of losing customers from poor service or response  
• Improves productivity of those who use the application  
• Increases speed of making decisions and providing information  
• Improves ability to scale application to support business growth |
| Security | Attributes that affect an application’s ability to prevent unauthorized intrusions | • Improves protection of competitive information-based assets  
• Reduces risk of loss in customer confidence or financial damages  
• Improves compliance with security-related standards and mandates |

Figure 1: Application Health Factors and Their Benefits to the Business

It is important to evaluate and monitor the structural quality, as these defects are likely to cause operational problems, and they are usually not detected through regular testing. As Dr. Bill Curtis explains in his guide "The Business Value of Application Internal Quality", several types of business risks and business performance problems are born from a poor software quality. By measuring the Application Health Factors, issues can be discovered and fixed before they become actual problems.

In order to go a bit more into detail towards understanding the influence of IT structural quality on business outcomes and value, we studied the idea behind CAST solutions. We understood that the Health Factors are critical in order to achieve certain IT objectives, which in turn translate into various Business Outcomes, which lead to the completion of a series of Business Objectives. You can observe these relationships in Figure 2.

![Figure 2: Relationships between health factors and business objectives](image)

Moreover, the evaluation of structural quality can help us form an idea about the technical debt enclosed in software, which is compared to a ticking bomb by Tim Johnson.

The purpose of technical debt is (being used as an) that of an indicator, which can aid in developing an action plan for a company's five most mission-critical applications. A decision

---

can be made about how much of this debt can and should be eliminated before the
application is released, and how much is still acceptable, based on the marginal return on
business value. Jonathan Bloom states that it is not different from managing personal debt\textsuperscript{23},
since "it incurs interest in the form of the extra effort it takes to maintain and enhance an
application due to the structural quality flaws in the code"\textsuperscript{24}. He also sees it as money that
didn't have to be spent. If companies spend a little more in the pre-production phase, they
save money by not having to fix problems afterwards. If that were the case, they could
allocate that amount on "increased innovation rather than on maintenance"\textsuperscript{25}.

\textsuperscript{23}Bloom, J. (2012, March 8). \textit{CRASHing Into Technical Debt}. Retrieved April 2012, from CAST On Quality Blog -
Because Good Software is Good Business: http://blog.castsoftware.com/crashing-into-technical-debt/


\textsuperscript{25}Bloom, J. (2011, December 14). \textit{A Crash Course on CAST's New CRASH Report}. Retrieved from:
http://blog.castsoftware.com/crashing-into-technical-debt/