# IS YOUR DATA WAREHOUSE SUCCESSFUL?
## Developing a Data Warehouse Process that responds to the needs of the Enterprise.

**Peter R. Welbrock**
**Smith-Hanley Consulting Group**
**Philadelphia, PA**

## ABSTRACT

Developing a Data Warehouse can be very costly to any enterprise in terms of both resources and money. There is a certain amount of fixed cost in every data warehousing project, with an additional set of variable costs depending upon the scope of project. The spending of the fixed cost requires a leap of faith based on the assumption that the potential benefits of developing the Data Warehouse will be accrued. The variable costs can be more closely monitored, with additional resources only being incrementally used based upon the return (benefits).

This paper will discuss the importance of developing a data warehousing process that inherently allows for the measurement of success. Each Data Warehouse will require a different process, based upon the particular needs and culture of the enterprise. The measurement of *success* will also differ from Data Warehouse implementation to implementation. The paper will also purport an algorithm that in principle is very simple that can be used to measure the success of a Data Warehouse.

## INTRODUCTION

Many Data Warehouse implementations are never monitored for their success. Others are only monitored based upon their technological facets (e.g. speed with which data can be queried) as opposed to the business facets (what enterprise needs are actually being addressed). The major misconception is to treat the success of the Data Warehouse in the same way as the success of an Operational System. Much has been written on the differences between Data Warehouses and OLTP (On-line Transactional Processing) Systems[i], but little on any corresponding differences on measurement of success.

The measurement of the success of the Data Warehouse is outside the experience of most Information Technology (I.T.) specialists. The major problem is that the data warehousing process is largely a business, as opposed to a technological exercise. It does not start or end in the actual implementation of the Data Warehouse. This means that the success cannot be fully ascertained by only looking at its apparent technological attributes but in terms of its impact to the entire enterprise, as defined by the explicit *vision* that is set.

## THE DATA WAREHOUSING PROCESS

There is still no definitive definition of a Data Warehouse. Usually, a Data Warehouse is defined based upon what it is, rather than what it does. This invariably leads to an incomplete definition since it leads to a fundamental misunderstanding: that a Data Warehouse is essentially a store of data that has been extracted from operational systems to be used for decision support. Although this definition is correct, it is incomplete. It leads to an implementation centric approach to data warehousing that can potentially result in a storage of data that is out of touch with the decision support needs of the enterprise. This definition does not have any intrinsic factors by which the Data Warehouse can be considered a success or a failure. In terms of success measurement, this definition does not give any real help.

To fully appreciate the entire scope of a Data Warehouse, it is more accurate to look at it in terms of a process. The Data Warehouse can therefore be defined as a *process by which the decision support enterprise needs are defined and satisfied*. Note that this definition does not include any mention of the extraction, transformation or storage of data, the classic triumvirate of Data Warehousing. Using this definition it is far easier to determine the success factors of the Data Warehouse. This is because there are two explicit factors contained within the

definition that lead directly to the ability to test the worth to the enterprise that the Data Warehouse has:

- An explicit list of defined enterprise needs.
- An explicit list of those needs that are satisfied.

There are as many different potential data warehouse processes as there are enterprises that could benefit from them. However, each of these potential processes share three common elements. These three elements are different stages that make up the data warehouse process. Each of these stages has its own internal process that will vary from enterprise to enterprise, but they are all necessary to help ensure that the Data Warehouse will be responsive to the needs of the enterprise. The three stages are:

- The Conceptual Data Warehouse: a business perspective.
- The Transition: moving to a technological lexicon.
- The Implemented Data Warehouse.

Each of these stages are outlined briefly below.

***Stage 1: The Conceptual Data Warehouse***

The conceptual data warehouse is purely a business view of what the data warehouse should contain. It contains only business metadata that pertains to the enterprise needs and the constituent business elements that make up each of those needs. There is no actual data (as distinct from metadata) within the conceptual data warehouse.

***Defining the vision for the data warehouse***

The driver behind the contents of the conceptual data warehouse is the *vision* of the project. The *vision* is the oracle that should be consulted whenever decisions have to be made that will influence the data warehouse. This *vision* is the seminal influence on the data warehouse and should be set at the beginning of the project. This is not to imply that the *vision* cannot change as the development of the data warehouse progresses, but without it there will be no cohesive decision making.

It is the *vision* that defines the scope of the data warehouse. The *vision* might also contain any enterprise level rules that will be adhered to by the data warehouse. For example, the *vision* might contain a clause that states that no users of the data warehouse should need to understand the structure of any of the tables. This has far reaching implications for the design of the data warehouse and is therefore very important in the way it will develop.

The *vision* contains the guidelines upon which the development of the contents of the data warehouse will be based. It is more than a mission statement. It is a set of rules by which the data warehouse will be governed.

***Defining the enterprise needs***

Based upon the *vision* it will be possible for a list of *enterprise needs* to be defined. These *needs* could well be very broad. Two examples of *enterprise needs* from two different industries are:

- To improve the purchasing of raw materials based upon analyses of sales (*manufacturing)*.
- To better target existing customers with mailings (*retail*).

These *enterprise needs* are completely independent of any knowledge of existing data, but are completely based upon the business itself. It is not necessary to have any input from a data warehouse implementation prospective to design the conceptual data warehouse.

The list of *enterprise needs* is pure business metadata. It should be stored electronically, possibly in a structure as simple as a SAS data file, but in of itself, all it does is to outline the potential scope of the data warehouse. Just because an item is in the conceptual data warehouse does not mean that it will be implemented. This decision is made in the second stage in the data warehousing process, described below.

The final stage in the defining of the enterprise needs is to assign a ranking of the relative importance of each of the needs. This could be a straight ranking from most to least important, or

more likely based upon scaling techniques (e.g. 5=very important, 4=important, etc...).

***Defining the business elements***

Each of these *enterprise needs* will need to be decomposed into more specific business elements. For example, looking at the second of the two *needs* listed above, that of better targeting customers, the following elements could be deduced:

- Unique customer identification.
- Previous products the customers have bought.
- Date the products were bought.
- Location where the products were bought.
- Payment type.
- Amount of each sale.

Each of these business elements will also be stored electronically, being linked to the enterprise need. It is possible (and in fact desirable) that an individual business element could be part of two or more different *enterprise needs*. Each of the different elements will need to be thoroughly defined from a business perspective, and the owner of the definition should be associated with the element. This will be invaluable as the data warehouse is developed so that there is no confusion as to what is actually being used.

***Stage 1: Synopsis***

The construction of the conceptual data warehouse is a structured wish list of needs the enterprise would like satisfied based upon the vision of the project. The list is based upon optimistic business requirements rather than pessimistic technological limitations. Just because a need is included in the conceptual data warehouse does not mean that it will be implemented. Because it is not implemented does not make it any less important to the business. It is possible, although undesirable, that the most important enterprise needs will not be implemented due to technological constraints. This will obviously lead to a data warehouse that has limited success, but is grounded in the reality of the situation.

***Stage 2: The Transition***

This stage of the data warehousing process is the most important to ensure a successful project. The following activities take place during this stage:

- Each of the business elements is decomposed into a technical lexicon.
- The decisions are made as to which enterprise needs will be addressed in the implemented data warehouse.
- User expectations are managed by including the business units in the decision making process.

***Decomposing the business elements***

Each of the business elements will have a specific business definition associated with it. Based upon this definition, the corresponding operational table(s) and column(s) that make up the element will be documented.

In the same way that business elements can span enterprise needs, individual columns of data can span business elements. Each of these columns (and therefore the tables that they come from) will be linked to the business elements (and therefore to the enterprise needs). This process can be quite complicated, especially in situations where more than one column of data will make up a particular business element. Many enterprises are in the unfortunate situation where multiple operational systems perform the same function. This could happen, for example, where an enterprise has bought another, but not converted the operational systems. In this case, it will be necessary to document not only the table(s) and the column(s), but also the system from where the data will be extracted.

It is very probable that not all of the business elements can be decomposed into a technical lexicon. This is because the data is not available to satisfy the enterprise need. There is nothing wrong if this happens. In fact, it is a strength of the process, since it allows for the business units to fully understand why their enterprise needs cannot be satisfied. Because they can understand this, then managing expectations is made far easier.

***Selecting Enterprise Needs***

Once the columns that will be needed to satisfy the enterprise needs are documented, then it will be possible, if the process is well thought-out, to determine which enterprise needs can be most easily be satisfied. There is a balance that will be essential during this phase: that of ease of implementation to importance of enterprise need.

It will be preferable to select the most important enterprise needs first, but this will not always be possible. There might be several reasons for this, including:

- The data to satisfy the need might not be available.
- The extraction and transformation of the data might be very expensive.
- The data that will be required will only be used to satisfy the one enterprise need. This will restrict any economies of scale that are desirable for an efficient implementation.

The selection of enterprise needs to be satisfied should be made based upon a combination of business needs to technological realities. The business units should be fully involved in the decisions. Only through a partnership between the business and technological communities can the data warehousing process be successfully designed.

Although there are three major stages proposed in the data warehousing process, they do not necessarily follow each other. As the data warehouse develops, new enterprise needs will be defined and therefore there will be a continual process of decomposing business elements and selecting new enterprise needs to be addressed within the data warehouse.

***Managing User Expectations***

One of the major problems with data warehouse implementations occurs when the user expectations are out of line with the realistic implementation expectancies. This occurs due to two major reasons:

- The business community that will make up the vast majority of the data warehouse usage is not part of the design process.
- The expectations of the data warehouse users are not managed correctly and therefore there is the presumption that more will be achieved than is reasonably possible.

Both of these can be avoided with very little effort and lot of pre-planning. In the first stage of the data warehouse process, it is the business units (the eventual users of the data warehouse) that design the conceptual data warehouse. This means that they are already the driving force behind the contents of the implemented data warehouse.

The business units are also involved in the selection of enterprise needs that are to be addressed within the implemented data warehouse. Any decisions that are made can be logically discussed between the business and technological units due to the documentation in the first stage of the process, alongside the decomposition of business elements into a technological lexicon (i.e. systems, tables and columns). This means that every decision can be logically decided upon based on real data that everyone understands. This alone will ensure that the business units fully understand the gradual evolution of the data warehouse and will have control of that evolution.

***Stage 2: Synopsis***

The second stage in the data warehouse process takes the enterprise needs that have been decomposed in the first stage and translates them into a technological lexicon. It is therefore possible to rationally select those enterprise needs that will be addressed within the implemented data warehouse based upon a full understanding of what will be needed, from a data perspective, to fulfil those needs.

This stage, as with the first, only involves the development of metadata. No ‘actual’ data is ever extracted or transformed, except as needed to fully understand its meaning. If this stage is completed, then it will be possible to investigate any column of data from an operational system and cross reference it to any of the enterprise needs defined in the first stage.

It is not necessary to fully decompose every single enterprise need before moving to the data warehouse implementation. It is sometimes necessary, based upon the defined *vision* of the data warehouse to select enterprise needs to address before any decomposition is undertaken. Business necessity should always override technological restrictions.

***Stage 3: The Implemented Data Warehouse***

This stage of the data warehousing process is the one most often written about. It involves the following steps:

- Extraction
- Transformation
  - Data Validation
  - Data Scrubbing
  - Data Integration
  - Data Derivation
  - Data Denormalization
  - Data Summarization
  - Metadata Creation
- Loading

Each of these topics is thoroughly covered in most literature about data warehousing and is largely beyond the scope of this paper. This stage is where it is possible to use tools such as the SAS/Warehouse Administrator™ to manage the process. Indeed, due to the steps taken in the first two stages of the process, it makes the SAS/Warehouse Administrator™ an even more powerful tool. It will be possible through the extraction of metadata (possibly using the available API) to analyze exactly which items of data defined in any of the Operational Data Definitions (ODDs) are used to satisfy which enterprise needs.

## HOW THE PROCESS HELPS DETERMINE THE SUCCESS OF THE DATA WAREHOUSE

To fully determine the success of a data-warehousing project it is necessary to consider the following:

***How the Vision is related to the Success***

It is not possible to determine the success of the data warehouse without fully understanding the *vision* (and therefore the scope) of the project. How can one determine the success of a project without knowing what it is meant to achieve?

***Success as a Dynamic Measure***

The degree of success of the data warehouse will be very dynamic based upon any changes in the business, or on any changes to the *vision*. A data warehouse that is working will result in the definition of new enterprise needs. The three-stage process outlined above fortunately allows for the continual incorporation of new enterprise needs into the data warehouse and will therefore allow for continual updates to the success factor.

Given that the success factor is based both on the *vision* of the data warehouse and that it is a dynamic measure, there are two different components to calculating a success factor: internal and external influences.

***Internal influences on the success factor***

The internal influence on the success factor is the piece that determines how well the data warehouse is performing given its current expectations. In other words, the internal influence determines how quickly users can get to data that already exists with the data warehouse.

In some ways, this would be similar to more classic ways that are popular with Information Technology units that would determine the success based upon the time it takes to access data. This is one part of the internal measure, but does not make up the entire picture.

The data warehouse is a tool for the users (the business community) and therefore any success must be measured on their terms. This means that it is not enough to just measure, for example, the amount of time it takes for a query to run. Once that query has run, there is also additional time that must be spent in the manipulation/summarization of the data so that the pertinent analyses can be performed.

From the user's perspective, if the data that is queried is not in exactly the correct form for the analyses, then additional manipulation will need to be performed. The more customized the data is to the user's needs, then the more time the user can spend on the actual analysis rather than the

extraction. Ideally, it should be possible for the user to directly run the analysis against the data in the warehouse, but this is often not the case. One of the major complaints from users is the amount of time it takes them to get the data. A good data warehouse design will reduce this amount of time. This will especially be the case if the process described above is used, since it inherently encourages a data model that will suit the solution of specific enterprise needs.

It is possible, therefore, to design a ratio as follows:

$$\text{Internal Success} = \frac{\text{Analysis Time}}{\text{Access Time} + \text{Analysis Time}}$$

This essentially sees the internal success of the data warehouse as a ratio of the amount of time the users spend actually performing their analyses versus the amount of time they spend actually getting to and manipulating the data. It assumes that it is bad for an analyst (user) to be spending valuable time accessing data and good to be spending time analyzing that data.

This ratio although easy to understand in principle is harder to actually calculate. This is especially the case when the data warehouse has a large number of users, each with different types of needs. The key to this ratio is to get a feeling from the users as to how easy (or difficult) it is for them to obtain the data they need to fulfil the any specific enterprise need.

Rather than calculating this need based upon each individual user, it should be possible to calculate it based upon each enterprise need addressed. Because each enterprise need has an associated importance factor (assigned in the first stage of the data warehousing process described above) it will then be possible to use this as a weight to obtain an overall data warehouse internal success factor.

***Example: Calculating an Internal Success Factor***

Suppose that in the design of a conceptual data warehouse, there were five different enterprise needs as follows:

1. Monitor retail sales by region to give managers control over their areas. (5)
2. Improve purchasing of inventory based on anticipated sales. (5)
3. Analyze returned merchandise to determine any key patterns with an aim to reducing the amount returned. (4)
4. Assess the success of promotions based on increase in sales versus cost of promotion. (1)
5. Assess the impact of making sales associates commission based. (2)

The figures in parentheses represent the importance of each of the enterprise needs with (5) being most important. Assume that although all five needs were included in the conceptual warehouse, that only needs 1,2 and 4 were actually implemented.

In talking with the users that are responsible for completing the analyses, it was determined that the following ratios for each of the three needs respectively were applicable:

| Enterprise Need | Time spent accessing data | Time spent analyzing data |
|---|---|---|
| 1 | 10% | 90% |
| 2 | 40% | 60% |
| 4 | 50% | 50% |

These figures, although not arbitrary, are based upon the user's perception as much as reality. The key is to get an idea as to the amount of time is spent on the analysis of the data, rather than to create a scientific analysis. Most users are more than willing to discuss such a ratio, especially when they are involved in the process.

The actual calculation of the internal success factor is therefore:

$$\frac{(.9 * 5) + (.6 * 5) + (.5 * 1)}{11} = 0.727$$

This applies a weighting factor to each of the three enterprise needs that are addressed within the implemented data warehouse. The weighting factor is the importance level, and the denominator (11) represents the sum of all the importance factors.

What this factor, 0.727 actually tells us, is that given the current implemented data warehouse, that we are 72.7% successful in our goal. Our goal is for our users to spend all of their time analyzing data and none of it in access and manipulation. Of course, it is highly unlikely that the goal of 100% will ever be reached, but that should be the target.

This example is just one way the internal success factor could be calculated. Different enterprises will determine the internal success factor in a different way. For instance, the manner in which each of the enterprise needs are weighted might well be different. The key to the calculating the factor is to have a good working relationship with the users to ascertain an accurate idea of the relative time they are spending on the access versus the analysis of data.

In many data warehouse implementations, the success is determined through such determinants as the speed with which a query runs. Such measurements are inherently included in the calculation above, since they determine the amount of time it takes to access the data and are therefore accounted for.

There are many factors that can affect the internal success factor. As has been mentioned above, the speed with which a query will run (which is dependent upon hardware, software and data modeling) is one such factor. Another, which is very much undervalued, is that of training. Very often, the data warehouse is either underutilized or not efficiently utilized due to the lack of training the users are given. With additional training the time it takes to access the data could be reduced, giving the users more time for analysis.

***External influences on the success factor***

The external influences on the success factor of the data warehouse is not based on what is included within the data warehouse, but what is not included. This measure is essentially a ratio that illustrates what the data warehouse is doing, versus what it should be doing. Such a measurement is only possible if a data warehousing process is implemented that documents fully the enterprise needs that make up the *vision*.

The external influence can be defined as follows:

$$\text{External Success} = \frac{\sum \text{Enterprise Needs Addressed}}{\sum \text{Enterprise All Needs}}$$

This equation assumes that there is a good attempt at documenting the universe of enterprise needs, based upon the specified *vision*. This does not mean that every single enterprise need has to be decomposed either in terms of business elements (in the first stage of the process) or into a technological lexicon (in the second stage of the process). Very often, due to limited resources, this decomposition does not always happen, but this should not stop the enterprise need from being documented and assigned an importance rating.

The external success factor is very dynamic. A successful data warehouse will lead to as many new questions (needs) as it does to answers. This means that each component of the above equation might well change on a monthly basis, depending upon how the data warehousing process has been designed.

As with the internal success factor, each of the enterprise needs will most probably be weighted for importance, and this should be reflected in the overall external success factor.

***Example: Calculating an External Success Factor***

Using the same example as in the section dealing with the internal success factors, there are five different enterprise needs actually defined. Of these, only three are actually implemented.

If the relative importance weighting were ignored, then the external success factor will be:

$$\frac{3}{5} = 0.6$$

If the relative importance weighting of each of the enterprise needs is taken into consideration, the external success factor will be:

$$\frac{\sum \begin{matrix}\text{weights of} \\ \text{implemented needs}\end{matrix}}{\sum \text{weights of all needs}} = \frac{11}{17} = 0.647$$

This means that we would be 64.7% successful in our attempt to meet our goal of addressing all of the enterprise needs that are documented based on the *vision*.

It is only possible to use such a measure in a situation where there is an honest attempt to collect and document all of the enterprise needs. This measure will not work in a situation where the scope of the data warehouse has not been fully defined.

If the scope of the data warehouse has not been defined through a *vision* document then the data warehouse will essentially be 'rudderless'. In this situation, the development of the data warehouse will be arbitrary and will therefore most likely fall prey to a situation where no one ever really knows what the data is used for. This is similar to the 'build and they will come' approach to data warehouse development, which assumes that if enough data is extracted someone will find a need for it.

***Calculating the overall success factor***

The overall aim of the data warehouse should be to have the users performing analysis one hundred percent of the time and for every single defined enterprise need to be addressed. This means that there are two components to the overall success factor, both of which have been discussed: the internal and the external success factors. The overall success factor can therefore be defined as:

$$\text{Overall Success Factor} = \text{Internal S.F.} * \text{External S.F.}$$

If the internal success factor were 1.0 (the users would be spending all of their time on the analysis of information) and the external success factor were also 1.0 (all known enterprise needs were addressed), then the overall success factor would be 100%, a flawless data warehouse.

***Example: Calculating the Overall Success Factor***

Using our examples above, calculating the internal and external success factors, then the overall success factor would be as follows:

$$\text{Overall Success Factor} = 0.727 * 0.647 = 0.47$$

The obvious question to then ask is: "What is a good overall success factor?" This is very much dependent upon the individual enterprise that has designed the data warehouse. It might well be that for certain enterprises, it is more important to address all enterprise needs than to have the analysis versus extract time very high. In this situation, there would be weighting associated to the internal and external success factors to reflect this bias.

This success factor is nothing more than an indicator as to how the data warehouse is performing. Its use should be as a way to assess the performance of the data warehousing process. Because it is a very dynamic measurement, it is also a statistic that can be used to compare data warehouse performance between different time periods.

**RETURN ON INVESTMENT (ROI)**

The overall success factor described above does not take into account any return on investment. Return on investment for a data warehouse is very difficult to calculate. It requires in depth knowledge of both the monetary costs and the monetary benefits accrued.
The monetary cost of a data warehouse should not only include the hardware infrastructure (computers, networks etc,), software and personnel costs (both in the I.T. and business units), but should also include the continuing cost of maintenance and usage. This means that if an analyst spends five hours using the data warehouse, this is a cost of the warehouse.

The data warehousing process as described in this paper at least makes the job of assessing benefits possible. Since the data warehouse is built around addressing specific enterprise needs, then for a need to actually be included on the list there must be a business justification. This business justification should usually include some kind of monetary benefit to the need being addressed. For example, taking one of the enterprise needs used above, the analysis of returned merchandise to determine any key patterns with an aim to reducing the amount returned, if successful there will be a monetary benefit. This however, can only be known after the fact. This monetary benefit will be the difference in the amount of merchandise returned before and after any action is taken based upon analysis performed from the data warehouse.

Very often the benefit of the data warehouse is based on the difference in cost to performing a task before versus after the warehouse has been implemented. In other words, it might take an individual two weeks to create a series of reports before the data warehouse addresses that enterprise need, but then it takes only one day after the implementation. The monetary benefit to the enterprise is seen as the savings in time spent to obtain the reports early.

Because in many situations the monetary amount of benefit cannot be known before the data warehouse addresses the enterprise need, then it is necessary to 'speculate to accumulate'. This returns to the point made during the Abstract: there has to be a belief that the data warehouse can monetarily benefit the enterprise and that it takes a leap of faith in terms of the investment in resources before fully knowing what the impact will be. Each incremental part of the data warehouse is an experiment with a hypothesis. The hypothesis is that there will be an enterprise wide benefit to the incremental investment, but there is always a cost to the experiment. The key is not a technological, but a business one. If the designers of the data warehouse really understand the business and the information that it will take to improve it, then the 'leap of faith' becomes more a 'leap of judgement'. A data warehouse can only support business judgement, not replace it.

The complex nature of calculating the ROI of a data warehouse means that other techniques should be used. The careful selection of enterprise needs to be addressed is one of these. The ROI discussion should informally come about during the second stage of the data warehousing process. The resources that it will take to address the enterprise need should be formulated along with the expected benefit to addressing that need. Neither exact costs or benefits can necessarily be created, but at least a discussion as to the implications can be had.

## SUMMARY

How good is your data warehouse? This is an often-avoided question, largely because there are limited tools available to answer it. The key to answering the question is to design the data warehousing process so that metadata exists that can be used to make a judgment.

To know how good a data warehouse is, it is necessary to know two pieces of information:

- What enterprise needs the data warehouse addresses versus those it ignores.
- How well the data warehouse solves the enterprise needs it addresses.

The former can be determined using an *external success factor* and the latter using an *internal success factor*. It is the combination of these two that will lead to the calculation of an *overall success factor*.

Because it is necessary to know not only what the data warehouse addresses, but what it does not address, it is absolutely essential to have a specified *vision* (scope) to the project. Once the scope has been determined, not only will it be the driver that will lead to an efficient data warehouse implementation, but it will also give the designers the ability to understand the limitations of the implementation.

The success of a data warehouse is dynamic. A good data warehouse implementation will naturally lead to new enterprise needs. These will either be implemented or not. In either case, the ratio of addressed to not addressed needs will change, thus changing the external success factor.

The internal success factor will also be continually changing either as the users become more expert in the efficient extraction of data from the warehouse, or as the warehouse develops so that the data model more efficiently reflects the needs of the users.

To know how good the data warehouse is will also show where more resources should be used. If the internal success factor is weak, then the likelihood is that there is an inefficient data model, not enough summarized data or the users are not well trained. If the external success factor is weak, then more resources should be put into addressing more of the needs. The success factor is therefore more than just an isolated statistic that can be reported to management, but is in itself a tool that can be used to make the data warehouse more effective for the enterprise.

[i] Kimball, R: *The Data Warehousing Toolkit: Practical Techniques for Building Dimensional Data Warehouses.* John Wiley and Sons, 1996. ISBN 0-471-15337-0, Chapter 1.

**Address:**

Any questions regarding this paper will be willingly answered. Please contact:

Peter Welbrock
Smith-Hanley Consulting Group
937 East Haverford Road
Suite 203
Bryn Mawr, PA 19010

Tel: 610 520-2900
e-mail: welbrock@erols.com