A TOOL FOR SOURCING DECISIONS

by

Miguel Manuel Miciano

B.S., Electrical Engineering, The University of Texas at Austin, 1986
M.S., Electrical Engineering, Purdue University, 1988

Submitted to the Sloan School of Management and the
Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degrees of

Master of Science in Management
and
Master of Science in Electrical Engineering and Computer Science

in conjunction with the
Leaders For Manufacturing Program
at the
Massachusetts Institute of Technology
May 1999

Signature of Author____
Sloan School of Management
Department of Electrical Engineering and Computer Science
May 13, 1999

Certified by ____
Dr. Donald B. Rosenfield
Sloan School of Management
Thesis Supervisor

Certified by ____
Professor Alvin W. Drake
Department of Electrical Engineering and Computer Science
Thesis Supervisor

Accepted by____
Professor Arthur C. Smith
Chairman, Committee on Graduate Students
Department of Electrical Engineering and Computer Science

Accepted by____
Lawrence S. Abeln
Director of Master's Program
Sloan School of Management

A TOOL FOR SOURCING DECISIONS

by

Miguel Manuel Miciano

Submitted to the Sloan School of Management and the
Department of Electrical Engineering and Computer Science
on May 13, 1999 in partial fulfillment of the requirements for the degrees of

Master of Science in Management
and
Master of Science in Electrical Engineering and Computer Science

ABSTRACT

A standard software tool is developed to assist analysts with sourcing decisions in a finite three-stage supply chain network. The purpose of this tool is to determine the minimum cost sourcing or make-versus-buy strategy. The tool can consider up to 13 major cost categories across the supply chain network. These costs can be viewed on an aggregated or disaggregated basis. The tool is also capable of performing analyses using discounted cash flow techniques. The tool consists of five worksheets within Microsoft Excel. The last worksheet contains a one period mixed-integer linear program that uses Solver to find a minimum cost sourcing arrangement. The tool is flexible, easily disseminated, transparent, and easy to use. Use of this tool should lead to more expeditious analyses while yielding reasonable estimates of the costs associated with a finite set of sourcing or make-versus-buy alternatives.

Thesis Supervisor: Donald B. Rosenfield
Title: Senior Lecturer of Management

Thesis Supervisor: Alvin W. Drake
Title: Professor of Electrical Engineering and Computer Science

Acknowledgments

I wish to acknowledge the Leaders For Manufacturing (LFM) program for its support of this work. LFM is a partnership between MIT and over a dozen American and international industrial companies. LFM's mission is to groom future manufacturing leaders through its Fellows program and advance the field of manufacturing through its research activities.

I wish to express my sincere gratitude to the program administrators for granting me the privilege of enrolling at MIT. My two-year tenure at this prestigious institution was a time of great intellectual and personal growth. I will always remember my "MIT years" with great fondness.

I wish to thank the Compaq Computer Corporation for being a gracious host during my internship. This thesis is the end product of that experience. I am impressed with the caliber of Compaq's employees and have learned much from them. I met so many people whom I wish to acknowledge that I run the risk of not naming all of them. I apologize to those whom I do not explicitly cite, and hope that my sincere gratitude for your help, guidance, and friendship during my stay at Compaq will suffice. Eugene McCabe, thank you for hosting my internship and for steering my project "from behind the scenes" to a successful conclusion. Paul Toomey, thank you for being a supportive supervisor. Doug Kellogg, thank you for your assistance and for playing a key role in getting my project launched. Stuart Sharpe, thank you for sharing your time and wisdom with me. Frank Fallon, I can't begin to express how indebted I am to you for everything you did for me: for those many hours of overtime you logged helping me develop the software tool, for being patient with me, and for sharing your knowledge with me. It was a joy working with all of you! My best wishes to all!

I wish to thank Don Rosenfield and Alvin Drake, my thesis advisors, for their support and patience throughout the internship and during the writing of this thesis.

I wish to thank my parents for their encouragement and financial support. I also wish to thank my sister Margie and my brothers Danny and Ben for their moral support. Your prayers and words of support were always a great source of comfort.

Finally, I wish to thank the two young ladies in my life. Andrea, thank you for being a constant source of support and for keeping your good cheer despite the many hardships that come with being married to a student. I also wish to thank my new daughter Maria Therese, born on February 26 of this year. You have given me a new perspective on life, and I look forward to raising you and being your father.

Deo omnis gloria!

CONTENTS

FIGURES

TABLES

1 Introduction

1.1 Problem Statement

The management of Compaq Computer Corporation's High Performance Systems Product Lifecycle Group (or Lifecycle Group) needs a standard software tool that internal financial analysts can use to assist them with sourcing and make-versus-buy decisions for Compaq's mid-range and high-end server product offerings. The purpose of this tool is to determine the minimum cost sourcing or make-versus-buy strategy. The requirements for this tool are that it be flexible, easily disseminated, transparent, and easy to use. Use of this tool should lead to more expeditious analyses while yielding reasonable estimates of the costs associated with a finite set of sourcing or make-versus-buy alternatives.

1.2 Definition of Key Terms

From a broad perspective, *sourcing decisions* are equivalent to decisions involving process positioning; that is, in making a sourcing decision, a company must choose from various positioning alternatives for a part or product.[1] In effect, a company must decide where the "source" will be for a particular part or product that it needs. Limiting component purchases to internal plants (company owned) or external plants (supplier plants) that meet certain cost, quality, and delivery metrics is an example of a sourcing decision.

A *make-versus-buy* decision is a specific kind of sourcing decision. A make-versus-buy decision involves choosing between two alternative sourcing strategies: making a part or product internally or purchasing a part or product from an outside supplier.

The *supply chain* or value chain for a product or service refers to *entities* or *processes* that are linked to one another because they add value to a final product or service. A microprocessor manufacturer such as Intel is a good example of an *entity* that is part of the supply chain for computers. The assembly process for printed circuit boards is an example of a *process* that is part of the supply chain for computers.

1.3 Justification for this Initiative

The primary goal of this initiative is to identify and, if needed, create a software tool that allows financial analysts to make better sourcing decisions and reduces the amount of time it takes financial analysts to perform sourcing analyses. The Lifecycle Group chose to make the sourcing decision more efficient for two principal reasons. First, sourcing analysis is an activity that is performed frequently, and any efficiencies that can be gained in this area will noticeably improve the productivity of financial analysts involved in sourcing analyses. Second, in the technology industry, the time between successive product generations is shortening; it stands to reason that the time required to arrive at sourcing decisions must shorten as well, at least proportionately.

In the Lifecycle Group, it is estimated that the average sourcing analysis project requires 13 man-days to complete. A recent sourcing analysis project for printed circuit boards took 45 man-days to complete and involved nine individuals from four cities. This was a complex project, and therefore, required a significant amount of time and resources to complete.

---

[1] Robert Hayes and Steven Wheelwright, Competing Through Manufacturing, (New York: John Wiley & Sons,

In a typical sourcing project, time is spent on the following activities: data collection, model development, and model reconciliation. Data collection involves gathering data from corporate databases and from plants that could be affected by the sourcing decision. The collected data is transformed into meaningful recommendations through the use of spreadsheet models. Each analyst on a sourcing analysis team typically creates his or her own unique model to process the data. The results obtained from each of these models are reconciled, and one overall recommended sourcing strategy is generated.

Within each of these activities, there are opportunities to cut the time requirement with the use of a pre-built standard model. Such a model, if accepted by the financial analyst community involved in sourcing analyses, can dramatically reduce or eliminate the amount of time spent on model development and model reconciliation. In terms of model development, a pre-built standard model will save time by limiting model development activities to specific project customization requirements. In terms of model reconciliation, a standard model will do away with most if not all activity related to model reconciliation because it will naturally limit discussions to the financial results and not the analysis technique. When these time savings are totaled, it is expected that the use of this tool will cut the time requirement associated with sourcing decisions by *as much as 50%.*

1.4 Background of Project Sponsor

Compaq Computer Corporation, headquartered in Houston, Texas, is one of the largest computer manufacturers in the world. It offers a complete line of computer products covering the entire price spectrum: from digital assistants and personal computers to large enterprise

---

1984), p. 275.

servers. "Compaq derives most of its revenues from business customers but also has product offerings for the home user, government, and schools."[2]

The work associated with this project was conducted at Compaq's Maynard, Massachusetts facility and was sponsored by the High Performance Systems Product Life Cycle Group, which has life cycle management responsibility for Compaq's mid-range servers and enterprise servers. This group became part of Compaq after the 1997 acquisition of Digital Computer Corporation by Compaq.

1.5 Thesis Overview

Chapter two provides some background information regarding mathematical programming. Also included in this chapter is a review of relevant published works concerning analysis methods and tools used in sourcing decisions. This sets the stage for chapter three, where the solution technique for the original problem statement is described. The main body of this chapter describes the "solution" – a sourcing analysis tool built on top of Microsoft Excel. The fourth chapter concludes this document by reviewing the tool's benefits and limitations.

The appendices are a rich source of detailed information about the sourcing tool. Appendix 1 lists the equations used in the tool, and Appendix 2 provides the mathematical description of the mixed-integer linear program that is included with the tool. A numerical example that uses this mixed-integer linear program is also provided in Appendix 2.

---

[2] Hoover's Online, "Compaq Computer Corporation: Company Capsule." www.hoovers.com, (March 10, 1999).

2 Background Research

2.1 Description of Terms Used in Mathematical Programming

A *constrained optimization* problem is a mathematical model composed of an *objective function* and *constraints* that bound the feasible solution space of the problem. The objective function is to be maximized or minimized within the feasible solution space. The variables in the objective function are called the *decision variables.* The goal of a constrained optimization problem is to find a set of decision variables that satisfy the objective and are within the problem's feasible solution space.

Let us examine, for a moment, a simple example of a constrained optimization problem. Through this example, we will put to use the terms introduced earlier. Assume we have a set of factories and a set of warehouses. Each factory has a fixed capacity, and each warehouse has a fixed demand. In addition, assume that we have been assigned the task of finding the least cost method of transporting finished goods from the factories to the warehouses. In this *constrained optimization problem*, the *objective* is to minimize the cost associated with transporting the finished goods from the factories to the warehouses. The *decision variables* are the number of units of finished goods that are to be shipped from each factory to each warehouse. The values selected for the decision variables (units of finished goods) must achieve the lowest possible transportation cost. These decision variables are *constrained* to be less than or equal to the capacity of each factory and greater than or equal to the demand of each warehouse.

A *linear program* is "a special case of a constrained optimization model [or problem]."[3] It is called linear because "the objective [function] and constraints are linear functions of the decision variables."[4] The use of the word "program" does not imply having to write code, and the fact that it may have this connotation is accidental.[5] The most common method for solving linear programs is with the *simplex method*, which is an algorithm that systematically searches for a solution to a linear program that optimizes a given objective function.[6]

A *mixed-integer linear program* is a linear program with one or more decision variables that are constrained to be integers. *Binary variables* are integer decision variables that are constrained to equal 0 or 1.[7] Mixed-integer linear programs typically use the simplex method together with algorithms such as the branch and bound algorithm to arrive at a solution. The branch and bound algorithm starts by relaxing the constraints for the integer variables. Next, it partially enumerates a solution to the constrained optimization problem and solves the resulting linear program.[8] Following this, the algorithm adds or tightens integer constraints to reduce the size of the feasible solution space and repeats the process of partially enumerating a solution and solving the resulting linear program. This entire process repeats until an optimal solution is found that satisfies all of the constraints in the constrained optimization problem.

---

[3] Wallace Hopp and Mark Spearman, Factory Physics, Foundations of Manufacturing Management, (Boston: Irwin McGraw Hill, 1996), p. 339.
[4] Hopp and Spearman, p. 339.
[5] Hopp and Spearman, p. 541.
[6] Wayne Winston and Christian Albright, Practical Management Science, Spreadsheet Modeling and Applications, (Belmont, CA: Wadsworth Publishing Company, 1997), p. 26.
[7] Winston and Albright, p. 214.
[8] Stephen Graves, Notes from a class entitled "System Optimization and Analysis for Manufacturing," Massachusetts Institute of Technology, (Summer 1997), p. 74.

A *non-linear program* is a constrained optimization model whose "objective function and/or constraints are not linear functions of the decision variables."[9] Non-linear programs are not guaranteed to find a solution that optimizes the objective function.

2.2 Literature Review

A literature survey was conducted to help guide the design of the sourcing tool. Three works were influential in the development of the tool. These works are cited in the References section and are listed below for the reader's convenience.

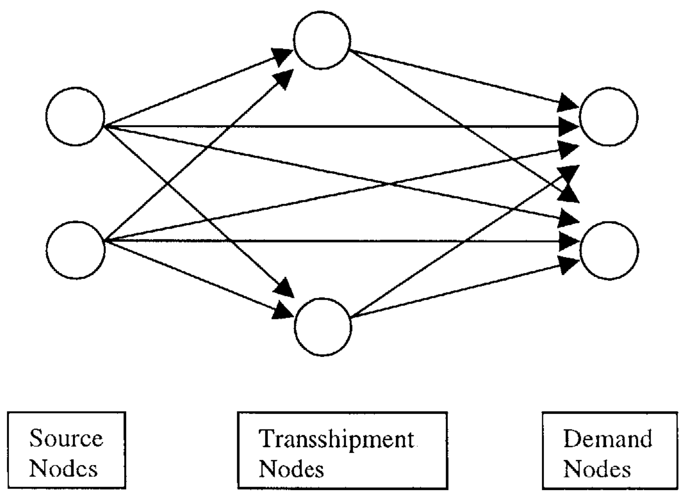
1. Notes by Stephen Graves for the class entitled "System Optimization and Analysis for Manufacturing" at the Massachusetts Institute of Technology.
2. <u>Practical Management Science, Spreadsheet Modeling and Applications</u> by Wayne Winston and Christian Albright.
3. "Global Supply Chain Management at Digital Equipment Corporation" by Bruce Arntzen, et al.

Graves uses the term *network problems* to describe the use of a network of nodes as a way of modeling some type of flow through a set of linked nodes. In the network structure, there are three kinds of nodes: source or supply nodes, transshipment nodes, and sink or demand nodes.[10] What links each pair of nodes are arcs that have an associated flow capacity and per unit cost. Graves also indicates that in the class of network problems called the *minimum cost network flow problem*, the objective is to find the network flow that simultaneously satisfies the demand at the sink (demand) nodes and achieves minimum cost.[11] Figure 1 shows the network structure of a minimum cost network flow problem.

---

[9] Winston and Albright, p. 268.
[10] Graves, Notes from class entitled "System Optimization and Analysis for Manufacturing," p. 55.

Figure 1. Example of the Network Structure for a Minimum Cost Network Flow Problem



Winston and Albright describe a specific kind of network problem called the "fixed charge plant and warehouse location problem."[12] In this problem, variable and fixed costs are modeled separately through the use of a mixed-integer linear program. The variable costs associated with each node pair in the network are computed on a per unit basis and assigned to the corresponding arc cost in the network representation. Fixed plant and warehouse costs are modeled with binary variables. If a plant or warehouse is used, the binary variable associated with that plant or warehouse equals one, which has the effect of adding the fixed cost associated with that plant or warehouse to the objective function. A binary variable that is zero will not add any fixed charges to the objective function. Winston and Albright then go on to create a model of the network in Excel using mixed-integer linear programming techniques. They use a versatile tool embedded within Excel called Solver – which finds solutions to linear, nonlinear, and integer programs – to find the minimum cost arrangement of plants and

---

[11] Graves, Notes from class entitled "System Optimization and Analysis for Manufacturing," p. 56.
[12] Winston and Albright, p. 241.

warehouses that simultaneously satisfies plant and warehouse capacity constraints as well as demand constraints.

Arntzen describes a sophisticated supply chain model that uses mixed-integer linear programming techniques to "simultaneously balance the multiple conflicting attributes of manufacturing and distribution: time, cost, and capacity."[13] This model is called the Global Supply Chain Model or GSCM and was developed internally at Digital Equipment Corporation between 1989 and 1993. GSCM is a mixed-integer linear program that can accommodate multiple "products, facilities, production stages, technologies, time periods, and transportation modes. It can also balance cost with time while considering the global issues of duty and duty relief, local content, and offset trade."[14] "The total benefit to date from all of the restructuring in manufacturing and logistics influenced by the use of the GSCM has been a $500 million cost reduction in manufacturing and a $300 million cost reduction in logistics as well as a reduction in required assets of over $400 million."[15]

---

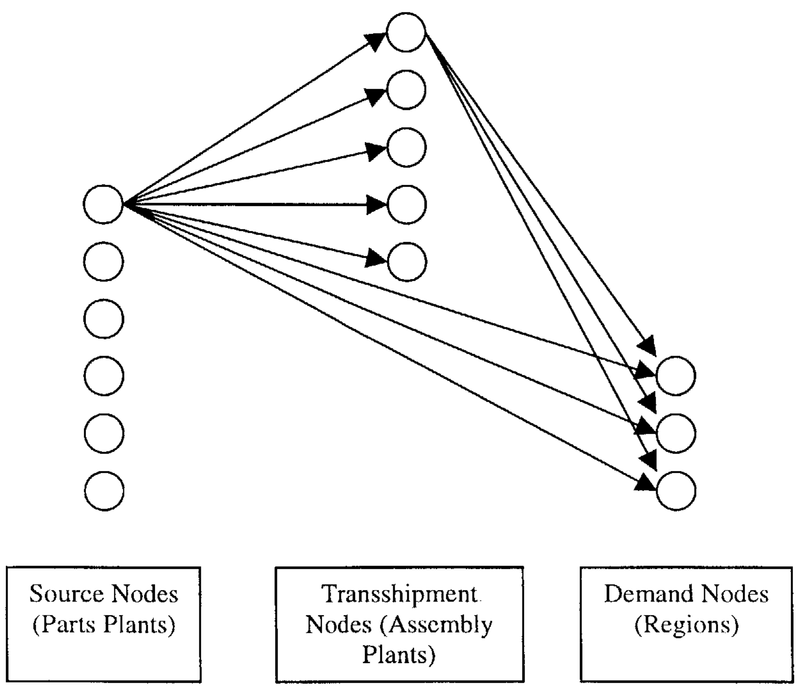
[13] Bruce Arntzen, et al, "Global Supply Chain Management at Digital Equipment Corporation," Interfaces, Vol. 25, No. 1, January-February 1995, p. 71, col. 1.
[14] Arntzen et al, p. 75.
[15] Arntzen et al, p. 82-83.

3 Solution Technique

3.1 Overview of the Supply Chain

A high level view of the supply chain for mid-range and high-end servers is shown Figure 2. The network shown in this figure consists of six source or computer parts plants, five transshipment or computer assembly plants, and three demand regions. The network was sized to accommodate Compaq's existing network of plants in the server business and a handful of extra nodes to accommodate existing external (contract) facilities as well as future external and internal facilities.

Figure 2. Supply Chain for Mid-range and High-end Servers



In this supply chain, the source nodes are connected to both the transshipment nodes and the demand nodes, and the transshipment nodes are connected to the demand nodes. For the sake of clarity, only the flows from one source node and one transshipment node are shown in

Figure 2. Also, keep in mind that each arc in the supply chain has an associated per unit cost. The term *arc cost* will be used to describe this cost. Therefore, in this representation, the cost of moving two items across an arc is equivalent to two-times the corresponding arc cost.

The source nodes (parts plants) manufacture or assemble parts such as printed circuit boards, power supplies, computer enclosures, computer frames, disk drives, cables, and peripheral devices. Since the network as sized in Figure 2 cannot accommodate the vast number of suppliers of parts that are required to manufacture a server, the source nodes should be viewed as the supply points for a *specific part.* For example, the source nodes could be viewed as supplying a specific part like printed circuit boards and nothing else. Note that the source nodes can also ship directly to the demand nodes. Given this understanding, the sourcing tool that was created can only analyze the supply chain for a specific part.

Items shipped from a source node to a transshipment node are expressed in terms of *parts-sets.* One parts-set is equivalent to one or more identical parts that are shipped together as one unit. At a transshipment node, a parts-set is converted into a server. In other words, each server gets exactly one parts-set which can come from any of the source nodes. This relationship is expressed mathematically in Appendix 2 section 2.1 under the heading *Transshipment Node Balance Constraints.*

Items shipped from a source node to a demand node are expressed in terms of individual parts as opposed to parts-sets. In Compaq's parlance, the individual parts are called *options.*

The total capacity of a source node is expressed in terms of the total number of individual parts that are shipped – options plus parts-sets multiplied by the number of parts per parts-set. The mathematical equation that describes the capacity constraints at the source nodes is provided in Appendix 2 section 2.1 under the heading *Capacity Constraints*.

The transshipment nodes or assembly plants are responsible for assembling, testing, and shipping server computers. Note that the term *system* can be used interchangeably with the term *server*. These nodes accept parts-sets from the supply nodes and ship servers to the demand nodes. For the purposes of this discussion, consider all other parts and components required for the proper assembly of a server to be exogenous. In other words, these "other" parts and components are assumed to be pre-positioned at the transshipment nodes awaiting the arrival of the parts-sets from the source nodes.

The demand nodes represent geographic areas where there is demand for server computers and options. These nodes accept servers from the transshipment nodes and/or options (individual parts) from the source nodes.

3.2 Customer Needs

As was mentioned in the section entitled *Justification for this Initiative*, the goal of this project is to create a tool for sourcing decisions that can be standardized across Compaq or at the minimum across the domain of influence of the Lifecycle Group. Several lead users were consulted to determine a set of customer needs. The primary needs are summarized below.

- **Easy to Use.** The tool is easy to learn. Users are able to learn how to use the tool without requiring formal training classes.

- **Flexible.** The tool uses disaggregated input data so that results can be grossed up in multiple ways.
- **Transparent.** The formulas in the tool are plainly visible to the user. In other words, the tool is a "clear box" rather than a "black box."
- **Easily Disseminated.** The tool is fully contained within a commercially available financial analysis software tool and is compatible with the Windows operating system. The commercially available software tool is the only piece of software that needs to be installed by the user. The files associated with this tool are easily disseminated via email.

A secondary need identified by the lead users is the ability to perform discounted cash flow analysis.

3.3 Evaluation of Alternatives

The tool called GSCM was introduced in the section titled *Literature Review*. This tool was developed at Digital Equipment Corporation and is now owned by Compaq through its acquisition of Digital. GSCM was evaluated to determine whether it satisfied the customer needs identified earlier. Table 1 is a summary of how GSCM fared in this evaluation.

Table 1. GSCM's Positive and Negative Attributes

Positive Attributes	Negative Attributes
GSCM can simultaneously model multiple products, types of facilities, time periods, manufacturing styles, and transportation modes.	GSCM does not meet the primary and secondary needs identified above. Particularly worrisome is the perception that GSCM takes a long time to learn.
GSCM includes cycle time (processing time + transit time) in its objective function.	GSCM is an "orphaned" piece of software developed for the Windows 3.1 environment. There are no plans to support this software in the near future.
	GSCM is perceived to be a tool for "specialists."

Despite its very impressive qualities, GSCM was disqualified primarily because it did not meet the needs established by the lead users. Since there were no other pre-built tools to evaluate, the decision was made to create a new tool.

3.4 Description of the Tool

Microsoft Excel's ubiquity within Compaq made it the natural platform from which to build the tool. From a technical standpoint, it makes sense to use Excel as a development platform because it offers a rich set of programming tools and mathematical functions as well as an optimization function called *Solver* that computes solutions to linear, integer, and non-linear programs. Also, given Excel's broad use and appeal, building this tool on top of Excel increases its chances of gaining wide acceptance among the financial analyst community at Compaq.

The tool models the supply chain network shown in Figure 2. The tool can only model the supply chain costs for a specific part and not a range of different parts; that is, it cannot simultaneously model the costs for printed circuit boards and power supplies, for example. The supply chain in Figure 2 consists of six source nodes (parts plants), five transshipment nodes (assembly plants), and three demand nodes (regions). The source and transshipment nodes in the network can be internal or external facilities. The tool looks at costs by year (this is the default time period but any time period can be used) and is capable of performing multi-year discounted cash flow (DCF) analyses.

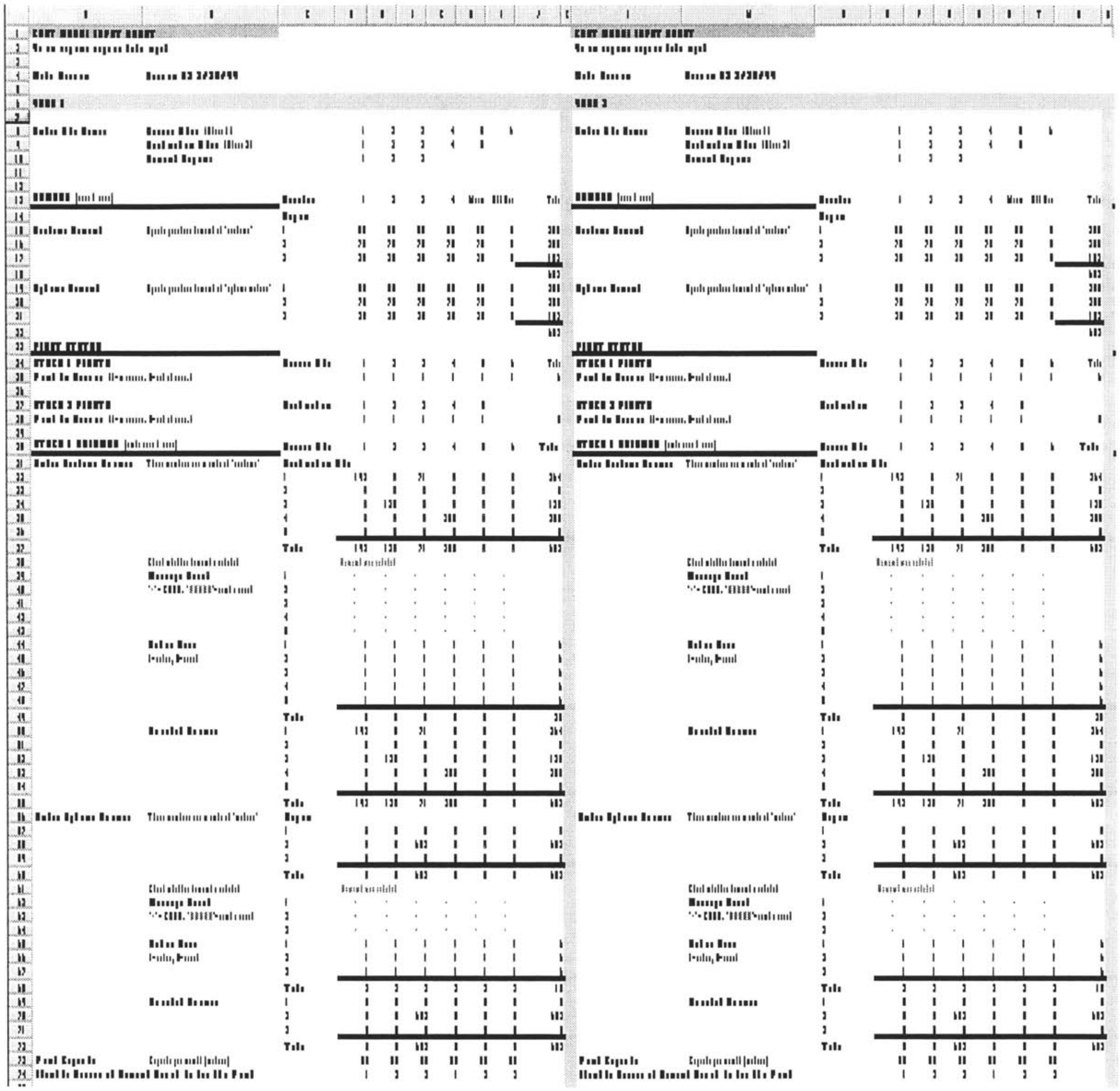
The tool is composed of five Excel worksheets. The names of these worksheets are: Cost Model Input, Cost Model, Cost Model Summary, Optimizer Input, and Optimizer Results. With the exception of the Optimizer Results worksheet, each worksheet is organized to show the costs in the supply chain along the vertical dimension of the worksheet, beginning with the costs at the source nodes and ending with the costs at the demand nodes. The Optimizer Results worksheet contains the Optimizer sub-tool. This sub-tool is a mixed-integer linear program that uses Solver to find the minimum cost sourcing arrangement.

Users have the flexibility of changing the number of years (time periods) in an analysis by simply copying the columns associated with one year and replicating those columns according to the number of years required by the analysis.

Figure 3 shows two years of input data as seen on the Cost Model Input worksheet. Due to space limitations, this figure only shows a fraction of the entire worksheet. The actual data on this worksheet is not what is important; rather, note the structure of the worksheet and, in particular, the fact that one can see two columnar sections that appear identical. Each of these columnar sections corresponds to one year in an analysis; year one is on the left and year two is on the right. The columns associated with year two were created by copying year one's 11 columns (A-K) and replicating them beginning in column I (the 12th column).

The Cost Model worksheet and Cost Model Summary worksheet have a similar format, and increasing the number of years in these worksheets is accomplished in a similar fashion.

Figure 3. Macro-View of the Cost Model Input Worksheet



3.4.1 Analysis Methods

The tool gives the user three ways of performing supply chain analysis.

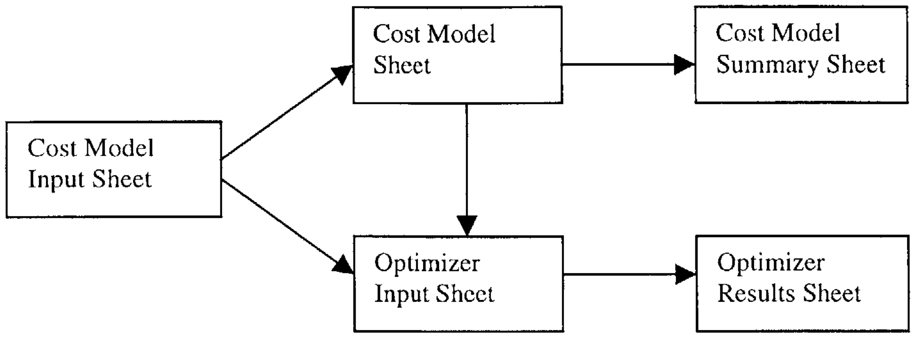
- **Perform Scenario Analysis**. The user can analyze various supply chain configurations by creating a separate file for each unique configuration. Once these files are created and fully populated with the necessary data, the user can print each file's Cost Model Summary worksheet and compare the supply chain costs for each configuration. Users must keep in mind that this analysis method does not necessarily result in an optimal solution.

- **Enumerate Solutions**. When the network representation of the supply chain is rather simple, the optimal solution can be enumerated through a close examination of the network arc costs, plant fixed costs, and node capacity constraints. In this case, only one file needs to be created. Once all of the input data is entered into the file, the total arc costs and plant fixed costs can be seen at the bottom of the Cost Model worksheet. Section 3.4.3.1 provides an example of this analysis method.
- **Use the Optimizer Sub-tool**. When the network representation becomes complicated like the one shown in Figure 2, enumerating an optimal solution becomes quite difficult. In this network, there are 108 (6*5*3 + 6*3) unique paths, 11 plant fixed costs, and 14 node constraints that need to examined to determine an optimal solution. Clearly, problems of this magnitude are best left for a computer. The Optimizer sub-tool was created to leverage a computer's ability to quickly solve constrained optimization problems. It uses a software program called Solver to find an optimal sourcing arrangement for the network depicted in Figure 2. An example that uses this sub-tool is provided in Appendix 2 section 2.2.

3.4.2 The Cost Model Input Worksheet

Figure 4 shows how data entered through the Cost Model Input worksheet flows into the remaining worksheets. Under ordinary circumstances, the Cost Model Input worksheet is the only worksheet that requires users to enter data.

Figure 4. Information Flow Between Worksheets



In contrast, the other four worksheets are passive; that is, the user's only interaction with these worksheets is to view their output. However, there are exceptions to this rule. These exceptions are as follows.

- Enhancements to the tool may require modifications to the equations contained in the worksheets.
- Changing the number of years in the discounted cash flow analysis to a number other than the default number of years (3) requires a corresponding modification to the equations in the "PV GRAND TOTAL" column in the Cost Model Summary worksheet. More specifically, cell references pointing to deleted years will need to be removed from the equations in this column. Similarly, cell references pointing to added years will need to be included in the equations in this column.
- An error message from Solver like the one described in the section entitled *The Optimizer Input Worksheet and the Optimizer Results Worksheet* will require direct user intervention within the Optimizer Results worksheet to help Solver find a solution. Please see *The Optimizer Input Worksheet and the Optimizer Results Worksheet* section for more details.

The Cost Model Input worksheet also has 19 software switches that enable or disable specific calculations. Table 5 in Appendix 1 section 1.1 provides a description of each switch and identifies the type of network arc costs that each switch influences.

3.4.3 The Cost Model Worksheet

The Cost Model worksheet is by far the largest worksheet in the tool. It has over 1400 rows. This worksheet is large because it is filled with matrices that model the supply chain network structure shown in Figure 2.

The cost categories that are modeled are summarized in Table 2. Appendix 1 section 1.1 contains mathematical and/or written descriptions of the equations used in this worksheet.

Table 2. Cost Categories for Network Arc Costs

Source Node to Transshipment Node Arcs or Source Node to Demand Node Arcs	Transshipment Node to Demand Node Arcs
Total raw material cost	Other raw material total cost
Inbound freight	Inbound freight
Inbound duty	Inbound duty
Inbound inventory carrying cost	Inbound inventory carrying cost
Cost by placement or standard hour. This cost is composed of: • Variable cost by placement or standard hour • Fixed cost by placement or standard hour • Tooling cost by placement or standard hour • Other burden by placement or standard hour	Cost by standard hour. This cost is composed of: • Variable cost by standard hour • Fixed cost by standard hour • Tooling cost by standard hour • Other burden by standard hour
Other costs	Other costs
New product start-up cost	New product start-up cost
Depreciation	Depreciation
In-house inventory carrying cost	In-house inventory carrying cost
Interplant freight	Regional freight
Interplant duty	Regional duty
Interplant pipeline inventory carrying cost	Regional pipeline inventory carrying cost
Tax holiday / charge	Tax holiday / charge

The cost categories *cost by placement or standard hour*, *in-house inventory carrying cost*, and *tax holiday/charge* need some explanation. *Cost by placement* is the labor and overhead cost associated with running a printed circuit board assembly line. It is calculated by multiplying the number of device placements on a printed circuit board by the cost of labor and overhead per device placement. A device is a discrete electrical component such as a resistor, capacitor, semiconductor chip, etc. This cost is used only if the part being manufactured is a printed circuit board. Also, as Table 2 suggests, this method of computing labor and overhead cost is only valid for arc costs that are linked to source nodes, because source nodes are where printed circuit boards are made.

Cost by standard hour is the labor and overhead cost associated with the manufacture of a part or server and, therefore, is valid for any arc cost in the network. For arc costs that are linked to source nodes, *cost by standard hour* is used in place of *cost by placement* when the part being manufactured is something other than a printed circuit board. This cost is obtained by multiplying the total standard hours involved in the manufacture of a part or server by the corresponding cost of labor and overhead per standard hour.

In-house inventory carrying cost is composed of raw material inventory carrying cost, WIP (work in process) inventory carrying cost, and finished goods inventory carrying cost. The raw material inventory carrying cost includes the cost associated with carrying raw material safety stock. The equation used to calculate safety stock is shown below. This equation is the standard *periodic review systems* safety stock equation with additional scaling factors for forecast error and a units adjustment.[16]

$$\text{Safety stock units} = z\sigma\alpha(1+\delta)\sqrt{r+L} \quad (1)$$

Where

z is the desired stock out coverage
σ is the standard deviation of demand in units of quarters
α is the units conversion from quarters to days and equals $\sqrt{4/365}$
δ is the forecast error adjustment in percent (fudge factor)
r is the time between replenishment orders, in units of days
L is the weighted average replenishment lead time, in units of days

When z = 1, a 68% stock-out coverage is achieved. When z = 2, a 95% stock-out coverage is achieved, and when z = 3, a 99.7% stock-out coverage is achieved. As can be seen from Equation 1, higher levels of stock out coverage increase the number of safety stock units.

The effective standard deviation of demand as seen by the source nodes is shown in Equation 2. This quantity is used in Equation 1 to calculate the safety stock units associated with each arc emanating from the source nodes.

$$\sigma = \sqrt{(\#ofPartsPerSystem * StdevOfDemandForSystems)^2 + StdevOfDemandForOptions^2} \quad (2)$$

The weighted average lead-time of the incoming raw material parts is computed using Equation 3. At the plants (source nodes), the incoming raw material parts are grouped by cost into three categories. Class A parts are the most expensive and typically have the longest lead times. Class B parts are less expensive than class A parts, and class C parts are the least expensive.

$$L = LeadTimeForClassAParts * ClassAPartsAs\%OfTotalRawMaterialCost + \\ LeadTimeForClassBParts * ClassBPartsAs\%OfTotalRawMaterialCost +$$

[16] Stephen Graves, Notes from a class entitled "Operations Management: Models & Applications," Massachusetts Institute of Technology, (Summer 1997), "Periodic Review Systems: Base Stock Policy," p. 1.

$$LeadTimeForClassCParts * ClassCPartsAs\%OfTotalRawMaterialCost \quad (3)$$

The *in-house inventory carrying cost* also includes the cost associated with carrying unplanned units of WIP and finished goods inventory. This excess inventory is caused by production skew. Simply stated, production skew refers to quarterly production volume that is heavily skewed to the third month in the quarter. This skew is largely driven by customer orders that follow a similar skew pattern within each quarter. A detailed description of the algorithm used to calculate this excess inventory is provided in Table 6, which can be found in Appendix 1 section 1.2. This table also has a listing of the equations that calculate this excess inventory.

Since parts plants and/or assembly plants can be located outside the U.S., the cost category *tax holiday or charge* was created to account for differences between foreign and U.S. tax rates. The equation used to calculate *tax holiday or charge* is shown below.

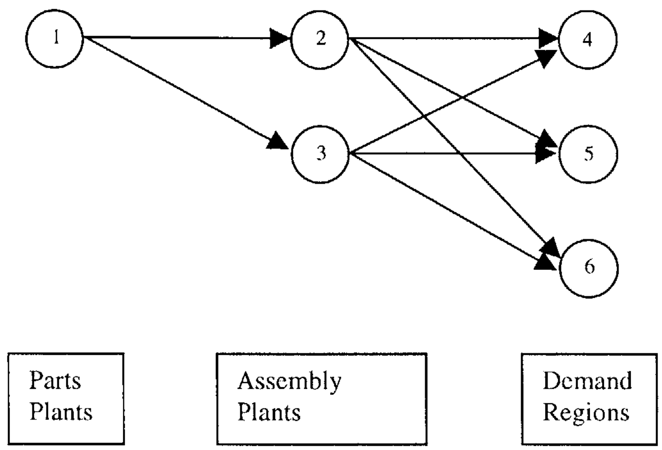
$$\textit{Tax holiday or charge} = \text{standard cost for finished goods} * \%\ \text{plant uplift} * (\text{foreign tax rate} - \text{U.S. tax rate}) \quad (4)$$

In the case of a non-U.S. plant, if the difference between the local (foreign) tax rate and the U.S. tax rate is positive, then the arc costs linked to this plant are increased by the pretax equivalent of the *tax holiday or charge*. If this difference is negative, the arc costs are decreased by the pretax equivalent of the *tax holiday or charge*. For more details, please see Table 6 in Appendix 1 section 1.2.

3.4.3.1 Using the Cost Model Worksheet: An Example

The supply chain network shown in Figure 5 depicts a scenario that was analyzed by Compaq to quantify the supply chain costs for one of its new high-end servers. Let us now use this same supply chain network to show how one can use the Cost Model worksheet to enumerate an optimal sourcing arrangement.

Figure 5. Supply Chain Network for the Cost Model Example



For the sake of brevity and simplicity, let us assume that the necessary cost data elements, plant capacities, and demand region requirements were previously entered into the Cost Model Input worksheet. The Cost Model worksheet will now have values for the cost categories shown in Table 2, and the total variable costs (total arc costs) and plant fixed costs will be visible at the bottom of this worksheet. The total variable costs and plant fixed costs are summarized in Table 3 below. Table 4 shows the plant capacities and demand region requirements.

Table 3. Total Variable Costs and Plant Fixed Costs

Total Variable Costs (Total Arc Costs)	**Plant Fixed Costs (Source Node and Transshipment Node Fixed Costs)**
$VariableCost_{12}$ = \$4	$FixedCost_1$ = \$20
$VariableCost_{13}$ = \$5	$FixedCost_2$ = \$30
$VariableCost_{24}$ = \$2	$FixedCost_3$ = \$40
$VariableCost_{25}$ = \$2	
$VariableCost_{26}$ = \$2	
$VariableCost_{34}$ = \$3	
$VariableCost_{35}$ = \$3	
$VariableCost_{36}$ = \$3	

Table 4. Plant Capacities and Demand Region Requirements

Plant Capacities (Source Node and Transshipment Node Capacities)	Demand Region Requirements (Demand Node Requirements)
$Capacity_1 = 60$	$Demand_4 = 20$
$Capacity_2 = 50$	$Demand_5 = 20$
$Capacity_3 = 50$	$Demand_6 = 20$

Given the fact that the demand in each region must be satisfied, the optimal solution boils down to one of following three scenarios: (1) source all of the systems (servers) from assembly plant #2, (2) source all of the systems from assembly plant #3, or (3) source the systems from both assembly plants. From Table 4, one can see that neither assembly plant has sufficient capacity to meet the demand from all three regions. Therefore, the optimal solution will be of the form described in scenario (3). Equation 4 is the total cost for this scenario.

$$\begin{aligned}\text{Total Supply Chain Cost} = {} & PartsSetShipment_{12} * VariableCost_{12} + \\ & PartsSetShipment_{13} * VariableCost_{13} + SystemShipment_{24} * VariableCost_{24} + \\ & SystemShipment_{25} * VariableCost_{25} + SystemShipment_{26} * VariableCost_{26} + \\ & SystemShipment_{34} * VariableCost_{34} + SystemShipment_{35} * VariableCost_{35} + \\ & SystemShipment_{36} * VariableCost_{36} + FixedCost_1 + FixedCost_2 + FixedCost_3 \quad (4)\end{aligned}$$

From this equation, one can see that the plant fixed costs do not factor into the decision process since the optimal solution requires all of the plants to be in operation. A close examination of the variable costs reveals the following fact: the sub-network composed of nodes 1, 2, 4, 5, and 6 has a lower total cost than the sub-network composed of nodes 1, 3, 4, 5, and 6. As a result, the optimal sourcing arrangement will necessarily require product to be shipped through assembly plant #2 up to its capacity, with any excess shipped through assembly plant #3. With this in mind, substituting the values shown in Table 3 and Table 4 into Equation 4 gives the total optimal cost shown in Equation 5.

$$\text{Total Supply Chain Cost} = 50 * \$4 + 10 * \$5 + 20 * \$2 + 20 * \$2 + 10 * \$2 + 10 * \$3 + \$20 + \$30 + \$40 = \$470 \quad (5)$$

3.4.4 The Cost Model Summary Worksheet

As its name suggests, the Cost Model Summary worksheet is a "presentation-ready" summary report of the calculation results contained within the Cost Model worksheet. It presents, in an orderly fashion, the costs shown in Table 2.

3.4.5 The Optimizer Input Worksheet and the Optimizer Results Worksheet

The total number of paths through the supply chain network shown in Figure 2 is quite large (6*5*3 + 6*3 = 108). Only a computer can examine all of these paths and determine a minimum cost sourcing arrangement that satisfies all of the capacity and demand constraints contained within this network.

The Optimizer sub-tool, which appears in the Optimizer Results worksheet, was developed to leverage the power of a computer. The sub-tool uses mixed-integer linear programming to identify the minimum cost sourcing arrangement. The general structure of this sub-tool closely resembles the structure of the "Warehouse Location Problem" presented in the Winston and Albright text, pages 240-245. The mathematical equations that describe the mixed-integer linear program are listed in Appendix 2, section 2.1. A full numerical example of the Optimizer sub-tool "in action" is available in Appendix 2, section 2.2.

The Optimizer sub-tool is a one period or one year model. The fixed and variable costs that it uses are calculated in the Optimizer Input worksheet. Binary variables are used to model plant fixed costs, and plant variable costs are equivalent to the network arc costs. The variable costs include a number of fixed cost elements such as fixed transformation costs and fixed material acquisition costs. These fixed costs can be considered variable or semi-variable because they change with production volume. The equations for (1) the additional units of WIP (work in process) introduced by the production skew, (2) the additional units of finished goods introduced by the production skew, and (3) raw material safety stock are not used in the calculation of network arc costs (variable costs) because they make the optimization problem non-linear. Please see Appendix 1 section 1.3 for a listing of the equations that appear in the Optimizer Input worksheet.

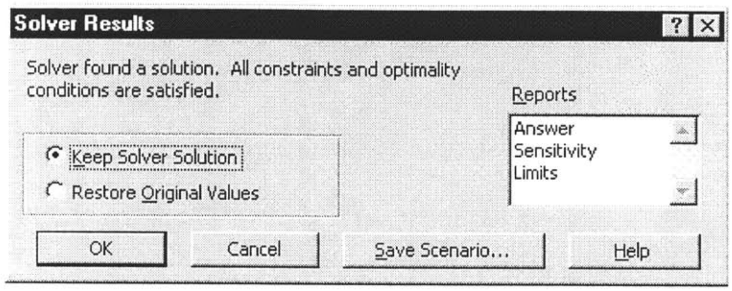
The fixed and variable costs are derived from the Cost Model worksheet year one costs and the Cost Model Input worksheet year one data. To run the Optimizer sub-tool on data from a year other than year one, the Cost Model Input worksheet data from the desired year needs to be copied to the year one position in the Cost Model Input worksheet.

The Optimizer sub-tool (mixed-integer linear program) is particularly sensitive to scaling. A poorly scaled linear program "contains some very large numbers (say 100,000 or more) and some which are very small (say 0.001 or less)."[17] In the case of the Optimizer sub-tool, poor scaling may occur when the unit costs are several orders of magnitude greater than the unit shipments or vice-versa. Solver's response to a scaling problem is to display one of the following messages in the Solver Results dialog box (Figure 6): "'the conditions for Assume

[17] Winston and Albright, p. 36.

Linear Model are not satisfied,' even though the model is linear, and the message 'Solver could not find a feasible solution' even though such a solution exists."[18]

Figure 6. Solver Results Dialog Box



Scaling problems cause round-off errors which undermine Solver's test for linearity and algorithmic methods (such as the simplex method).[19] In some cases, the Optimizer sub-tool will fail to arrive at a solution even without an apparent scaling problem. When this happens, a quick work-around is to reset the sub-tool's outputs on the Optimizer Results worksheet. These outputs are effectively the decision variables of the mixed-integer linear program. They represent the number of parts-sets, parts, or systems flowing through the supply chain network. From the perspective of the Optimizer sub-tool, resetting the decision variables is equivalent to forcing to zero the numbers that appear under the headings *source site to destination site parts-sets shipments*, *source site to region parts shipments*, and *destination site to region system shipments*. Please see Figure 7 for a visual layout of the Optimizer Results worksheet.

3.5 Tool Verification Tests

On three separate occasions, the outputs of the Cost Model worksheet and Cost Model Summary worksheet were checked against another Excel-based spreadsheet model

---

[18] Winston and Albright, p. 245.
[19] Winston and Albright, p. 36.

independently developed by a senior financial analyst in the Lifecycle Group. In each test, identical data inputs were entered into each model and the resulting outputs were compared. Any errors detected during these tests were corrected. Once this step was completed, the outputs of the Optimizer Input worksheet were then checked against similar outputs on the Cost Model worksheet. Finally, the arc flows generated by the Optimizer sub-tool in the Optimizer Results worksheet were checked to see if they "made sense;" that is, the outputs of the Optimizer sub-tool were spot checked to see if they actually appeared to generate a minimum cost solution. This process was repeated a few times beginning with simple scenarios and ending with the more complicated scenario shown in Appendix 2, section 2.2.

4 Conclusions

4.1 Benefits of the Tool

At this time, the tool is currently being rolled out within Compaq. As was mentioned earlier, the number of man-days required to complete a sourcing analysis is expected to go down by as much as 50%. In order to achieve these savings, the tool must achieve a high level of acceptance among the using community.

4.2 Limitations of the Tool

This tool should not be used in isolation; rather, it should be thought of as being part of a total analysis that includes the consideration of strategic issues. For example, if the tool indicates that sourcing a part from an external entity is the least cost alternative, the next step in the analysis process should be to consider, among other things, the impact of this decision on: (1) the economic viability of the remaining internal sites that will continue producing the part, (2) the core capabilities of the company – will they be eroded if this part is outsourced, (3) product quality, and (4) customer satisfaction, particularly with regard to product lead time.

The Optimizer sub-tool does not do multi-period analyses and does not factor cycle time (processing time + transit time) in the objective equation. These are non-trivial enhancements that if made will move the Optimizer sub-tool to a higher level of complexity. These kinds of enhancements should be done with care, since increased complexity may limit the sub-tool's attractiveness to the using community.

4.3 Extensions

Although the tool is initially focused on sourcing analyses for server computers, with a few minor adjustments, it can be applied to any kind of product that fits the network representation shown in Figure 2.

4.4 Improvements

One near term improvement that can be made to the sourcing tool is to use the Excel macro language, Visual Basic for Applications, to automate the process of adding or subtracting years in a DCF (discounted cash flow) analysis. This change is expected to increase the tool's appeal since it will eliminate any errors that may arise from changing the number of years in an analysis.

In general, users are expected to modify the tool as the nature of their analysis changes, when more functionality is needed, and when more precise calculations are required. The tool was created with this notion in mind. With time, a total redesign of the tool will probably occur as users gain more expertise in the area of simulation and modeling. If this comes to pass, the core characteristics of the present tool – easy to use, flexible, transparent, and easily disseminated – will likely become integral attributes of the new tool as well.

Appendix 1

1.1 Software Switches in the Cost Model Input Worksheet

Table 5. Summary of Software Switches

Description of Switch	Type of Arc Cost for which Switch Applies
Selects placement or standard hour costs	Source to transshipment arc costs and source to demand arc costs
Indicates whether this year is the last year for the calculation of depreciation	Source to transshipment arc costs and source to demand arc costs
Indicates whether this year is the last year for the calculation of depreciation	Transshipment to demand arc costs
Enable or disable safety stock equations used in in-house inventory calculation	Source to transshipment arc costs and source to demand arc costs
Enable or disable safety stock equations used in in-house inventory calculation	Transshipment to demand arc costs
Enable or disable base WIP calculation used in in-house inventory calculation	Source to transshipment arc costs and source to demand arc costs
Enable or disable base WIP calculation used in in-house inventory calculation	Transshipment to demand arc costs
Enable or disable base finished goods calculation used in in-house inventory calculation	Source to transshipment arc costs and source to demand arc costs
Enable or disable base finished goods calculation used in in-house inventory calculation	Transshipment to demand arc costs
Enable or disable WIP adjustment due to production skew. This adjustment is used in the in-house inventory calculation.	Source to transshipment arc costs and source to demand arc costs
Enable or disable WIP adjustment due to production skew. This adjustment is used in the in-house inventory calculation.	Transshipment to demand arc costs
Enable or disable finished goods adjustment due to production skew. This adjustment is used in the in-house inventory calculation.	Source to transshipment arc costs and source to demand arc costs
Enable or disable finished goods adjustment due to production skew. This adjustment is used in the in-house inventory calculation.	Transshipment to demand arc costs
Raw materials offset used in the in-house inventory calculation	Source to transshipment arc costs and source to demand arc costs
Raw materials offset used in the in-house inventory calculation	Transshipment to demand arc costs
WIP offset used in the in-house inventory calculation	Source to transshipment arc costs and source to demand arc costs
WIP offset used in the in-house inventory calculation	Transshipment to demand arc costs

Description of Switch	Type of Arc Cost for which Switch Applies
Finished goods offset used in the in-house inventory calculation	Source to transshipment arc costs and source to demand arc costs
Finished goods offset used in the in-house inventory calculation	Transshipment to demand arc costs

1.2 Equations Used in the Cost Model Worksheet

Table 6. Cost Equations Associated with Source Node to Transshipment Node Arcs or Source Node to Demand Node Arcs

Cost[Φ]	Equation and/or Description[#]	Units
Total raw material cost	Raw material cost + Outsourcing raw material profit + *Inbound freight*	$/part
Inbound Freight	Average cost per pound * raw material weight in pounds	$/part
Inbound duty	% duty on inbound raw materials * % regional sales * *Total raw material cost*	$/part
Inbound inventory carrying cost for cost of money	*Total raw material cost* * Inbound transportation days at cost of money / 365 days * Cost of money	$/part
Inbound inventory carrying cost for cost of revaluation & obsolescence	(Raw material cost + Outsourcing raw material profit) * Inbound transportation days for cost of revaluation and obsolescence / 365 days * Cost of revaluation and obsolescence	$/part
Inbound inventory carrying cost	*Inbound inventory carrying cost for cost of money* + *Inbound inventory carrying cost for cost of revaluation & obsolescence*	$/part
Cost by placement	Average number of placements * (Variable cost per placement + Fixed cost per placement + Tooling cost per placement + Other burden per placement)	$/part
Cost by standard hour	Average MTM assembly time * (Variable transformation + Variable material acquisition excluding freight + Other variable burden + Fixed transformation + Fixed material acquisition excluding freight + Tooling + Other fixed burden)	$/part
Other costs	Other costs	$/part
New product start-up cost	Total new product start-up cost / Volume	$/part
Depreciation	• If continuing year, *Depreciation* = Total expenditures * Depreciation rate / Volume • If last year, *Depreciation* = (Total expenditures for all years – Accumulated depreciation) / Volume	$/part
Standard cost for raw materials	Raw material cost + Raw material outsourcing profit	$/part
Standard cost for WIP	*Total raw material cost* + 0.5 * (*Cost by placement* or *Cost by standard hour*)	$/part

Cost[Φ]	Equation and/or Description[#]	Units
Standard cost for finished goods	*Total raw material cost* + (*Cost by placement* or *Cost by standard hour*)	\$/part
Safety stock units	Safety stock units = $z\sigma\alpha(1+\delta)\sqrt{r+L}$ Where • z = number of standard deviations required to achieve the desired stock out coverage. z = 1 gives 68% stock-out coverage, z = 2 gives 95% stock-out coverage, and z = 3 gives 99.7% stock-out coverage. • σ = standard deviation of demand[⊕] in units of quarters • α = units conversion from quarters to days = $\sqrt{4/365}$ • δ = forecast error adjustment (%) • r = review period or replenishment cycle, in units of days • L = weighted average replenishment lead time, in units of days = lead time for class A parts * class A parts as % of total cost + lead time for class B parts * class B parts as % of total cost + lead time for class C parts * class C parts as % of total cost	part

Cost[Φ]	Equation and/or Description[#]	Units
Additional WIP units due to skew, *Additional finished goods units due to skew*	The goal of this calculation is to get the average effect of production skew on inventory. A typical three-month skew pattern is used. • Average production in month 1 = month 1 skew % * quarterly demand • Average production in month 2 = month 2 skew % * quarterly demand • Average production in month 3 = month 3 skew % * quarterly demand • Month 1 skew + month 2 skew + month 3 skew = 100% • Capacity constrained production in month 3 = MIN(plant capacity, Average production in month 3) • Capacity constrained production in month 2 = MIN(plant capacity, Average production in month 3 + Average production in month 2 – Capacity constrained production in month 3) • IF ((Average production in month 3 + Average production in month 2 – Capacity constrained production in month 3) > plant capacity) THEN {MIN(plant capacity, Average production in month 3 + Average production in month 2 + Average production in month 1 – Capacity constrained production in month 3 – Capacity constrained production in month 2)} ELSE {MIN(plant capacity, Average production in month 1)} • *Additional WIP units due to skew* = MAX(Average production in month 2 + Average production in month 1 – Capacity constrained production in month 2 – Capacity constrained production in month 1, 0) • *Additional finished goods units due to skew* = MAX(Average production in month 3 + Average production in month 2 + Average production in month 1 – Capacity constrained production in month 3 – Capacity constrained production in month 2 – Capacity constrained production in month 1, 0) Average production in month 1, average production in month 2, and average production in month 3 are capped at the capacity of the plant. If the capacity is exceeded in month 3, the excess orders are sent to month 2. If capacity is exceeded in month 2 after including the excess orders from month 3, the excess orders are sent to month 1. If capacity is exceeded in month 1 after including the excess orders from month 2 and month 3, the excess orders show up as finished goods. The excess orders processed in months 2 and 1 show up as WIP (work in progress).	part
Inventory carrying cost	Cost of money + Cost of revaluation and obsolescence	%
Raw material inventory carrying cost	*Standard cost for raw materials* * *Safety stock units** *Inventory carrying cost* / Volume + Raw material offset in days * *Standard cost for raw materials* * *Inventory carrying cost* / 365 days	$/part

Cost[Φ]	Equation and/or Description[#]	Units
WIP inventory carrying cost	(Manufacturing cycle time in days + WIP offset in days) * *Standard cost for WIP* * *Inventory carrying cost* / 365 days + *Additional WIP units due to skew* * *Standard cost for WIP* * *Inventory carrying cost* / Volume	$/part
Finished goods inventory carrying cost	[MAX(Manufacturing cycle time in days – Lead time goal in days, 0) + Finished goods offset in days] * *Standard cost for finished goods* * *Inventory carrying cost* / 365 days + *Additional finished goods units due to skew* * *Standard cost for finished goods* * *Inventory carrying cost* / Volume	$/part
In-house inventory carrying cost	*Raw material inventory carrying cost* + *WIP inventory carrying cost* + *Finished goods inventory carrying cost*	$/part
Subtotal of variable costs	*In-house inventory carrying cost* + *Inbound inventory carrying cost* + *Cost by standard hour or Cost by placement* + *Other costs* + *Inbound duty* + *Total raw material cost*	$/part
Interplant freight	Cost per part per pound * Part weight in pounds	$/part
Interplant duty	% duty on outbound finished goods or parts * % regional sales * (*Standard cost for finished goods* + *Interplant freight*)	$/part
Interplant pipeline inventory carrying cost	*Standard cost for finished goods* * Pipeline days for cost of money * Cost of money / 365 days + *Standard cost for finished goods* * Pipeline days for revaluation and obsolescence * Cost of revaluation and obsolescence / 365 days	$/part
Tax holiday or charge	• IF U.S. tax rate = "N.A." THEN {Tax holiday or charge rate = 0} ELSE {Tax holiday or charge rate = Source country tax rate – U.S. tax rate} • *Tax holiday or charge* = *Standard cost for finished goods* * % plant uplift * Tax holiday or charge rate	$/part
Pretax equivalent of Tax holiday or charge	IF U.S. tax rate = "N.A." THEN {*Pretax equivalent of Tax holiday or charge* = *Tax holiday or charge* / (1 – Source country tax rate)} ELSE {*Pretax equivalent of Tax holiday or charge* = *Tax holiday or charge* / (1 – U.S. tax rate)}	$/part
Total fixed costs	*New product start-up cost* + *Depreciation*	$
Total variable costs (For source node to transshipment node arcs)	(*Subtotal of variable costs* + *Interplant freight* + *Interplant duty* + *Interplant pipeline inventory carrying cost* + *Pretax equivalent of Tax holiday or charge*) * Number of parts per system	$/parts-set
Total variable costs (For source node to demand node arcs)	*Subtotal of variable costs* + *Interplant freight* + *Interplant duty* + *Interplant pipeline inventory carrying cost* + *Pretax equivalent of Tax holiday or charge*	$/part

[Φ] In this column, cost categories that also appear in Table 2 are in bold italic. All other equations are in plain italic (non-bold).

In this column, names that appear in italic refer to costs whose equation is defined in this table.

⊕ $$\sigma = \sqrt{(\#ofPartsPerSystem * StdevOfDemandForSystems)^2 + StdevOfDemandForOptions^2}$$

Table 7. Cost Equations Associated with Transshipment Node to Demand Node Arcs

Cost[Φ]	Equation and/or Description[#]	Units
Other raw material total cost	Other raw material cost + Outsourcing profit for other raw material + *Inbound freight*	$/system
Inbound Freight	Average cost per pound * Other raw material weight in pounds	$/system
Inbound duty	% duty on inbound "other" raw materials * % regional sales * *Other raw material total cost*	$/system
Inbound inventory carrying cost for cost of money	*Other raw material total cost* * Inbound transportation days at cost of money / 365 days * Cost of money	$/system
Inbound inventory carrying cost for cost of revaluation & obsolescence	(Other raw material cost + Outsourcing profit for other raw material) * Inbound transportation days for cost of revaluation and obsolescence / 365 days * Cost of revaluation and obsolescence	$/system
Inbound inventory carrying cost	Same as corresponding entry in Table 6.	$/system
Cost by placement	N.A.	-
Cost by standard hour	Same as corresponding entry in Table 6.	$/system
Other costs	Same as corresponding entry in Table 6.	$/system
New product start-up cost	Same as corresponding entry in Table 6.	$/system
Depreciation	Same as corresponding entry in Table 6.	$/system
Standard cost for raw materials	Other raw material cost + Outsourcing profit for other raw material + Number of parts per system * Standard cost for finished part NOTE: "Standard cost for finished part" is equivalent to "Standard cost for finished goods" in Table 6.	$/system
Standard cost for WIP	*Other raw material total cost* + 0.5 * *Cost by standard hour* + Number of parts per system * Standard cost for finished part NOTE: "Standard cost for finished part" is equivalent to "Standard cost for finished goods" in Table 6.	$/system
Standard cost for finished goods	*Other raw material total cost* + *Cost by standard hour* + Number of parts per system * Standard cost for finished part NOTE: "Standard cost for finished part" is equivalent to "Standard cost for finished goods" in Table 6.	$/system

Cost[Φ]	Equation and/or Description[#]	Units
Safety stock units	Safety stock = $z\sigma\alpha(1+\delta)\sqrt{r+L}$ Where • z = number of standard deviations required to achieve the desired stock out coverage. z = 1 gives 68% stock-out coverage, z = 2 gives 95% stock-out coverage, and z = 3 gives 99.7% stock-out coverage. • σ = standard deviation of demand for systems in units of quarters • α = units conversion from quarters to days = $\sqrt{4/365}$ • δ = forecast error adjustment (%) • r = review period or replenishment cycle, in units of days • L = weighted average replenishment lead time, in units of days = lead time for class A parts * class A parts as % of total cost + lead time for class B parts * class B parts as % of total cost + lead time for class C parts * class C parts as % of total cost	System
Additional WIP units due to skew, *Additional finished goods units due to skew*	Same as corresponding entry in Table 6.	System
Inventory carrying cost	Same as corresponding entry in Table 6.	%
Raw material inventory carrying cost	Same as corresponding entry in Table 6.	$/system
WIP inventory carrying cost	Same as corresponding entry in Table 6.	$/system
Finished goods inventory carrying cost	Same as corresponding entry in Table 6.	$/system
In-house inventory carrying cost	Same as corresponding entry in Table 6.	$/system
Subtotal of variable costs	*In-house inventory carrying cost + Inbound inventory carrying cost + Cost by standard hour + Other costs + Inbound duty + Other raw material total cost*	$/system
Regional freight	Cost per system per pound * (Other raw material weight in pounds + part weight in pounds * number of parts per system)	$/system
Regional duty	% duty on outbound system * % regional sales * (*Standard cost for finished goods + Interplant freight)*	$/system
Regional pipeline inventory carrying cost	Same as corresponding entry in Table 6.	$/system

Cost[Φ]	Equation and/or Description[#]	Units
Tax holiday or charge	Same as corresponding entry in Table 6.	\$/system
Pretax equivalent of Tax holiday or charge	Same as corresponding entry in Table 6.	\$/system
Total fixed costs	Same as corresponding entry in Table 6.	\$
Total variable (arc) costs	*Subtotal of variable costs* + *Regional freight* + *Regional duty* + *Regional pipeline inventory carrying cost* + *Pretax equivalent of Tax holiday or charge*	\$/system

[Φ] In this column, cost categories that also appear in Table 2 are in bold italic. All other equations are in plain italic (non-bold).
[#] In this column, names that appear in italic refer to costs whose equation is defined in this table.

1.3 Equations Used in the Optimizer Input Worksheet

Table 8. Optimizer Sub-Tool Cost Equations Associated with Source Node to Transshipment Node Arcs or Source Node to Demand Node Arcs

Cost[Φ]	Equation and/or Description[#]	Units
Total raw material cost	Same as corresponding entry in Table 6.	$/part
Inbound Freight	Same as corresponding entry in Table 6.	$/part
Inbound duty	Same as corresponding entry in Table 6.	$/part
Inbound inventory carrying cost for cost of money	Same as corresponding entry in Table 6.	$/part
Inbound inventory carrying cost for cost of revaluation & obsolescence	Same as corresponding entry in Table 6.	$/part
Inbound inventory carrying cost	Same as corresponding entry in Table 6.	$/part
Cost by placement	Same as corresponding entry in Table 6.	$/part
Cost by standard hour	Same as corresponding entry in Table 6.	$/part
Other costs	Same as corresponding entry in Table 6.	$/part
New product start-up cost	Total new product start-up cost	$
Depreciation	• If continuing year, *Depreciation* = Total expenditures * Depreciation rate • If last year, *Depreciation* = (Total expenditures for all years – Accumulated depreciation)	$
Standard cost for raw materials	Same as corresponding entry in Table 6.	$/part
Standard cost for WIP	Same as corresponding entry in Table 6.	$/part
Standard cost for finished goods	Same as corresponding entry in Table 6.	$/part
Safety stock units	N.A. because calculation is nonlinear.	-

Cost[Φ]	Equation and/or Description[#]	Units
Additional WIP units due to skew, *Additional finished goods units due to skew*	N.A. because calculation is nonlinear.	-
Inventory carrying cost	Same as corresponding entry in Table 6.	%
Raw material inventory carrying cost	(Raw material offset in days + Optimizer raw material offset in days) * *Standard cost for raw materials* * *Inventory carrying cost* / 365 days	$/part
WIP inventory carrying cost	(Manufacturing cycle time in days + WIP offset in days + Optimizer WIP offset in days) * *Standard cost for WIP* * *Inventory carrying cost* / 365 days	$/part
Finished goods inventory carrying cost	[MAX(Manufacturing cycle time in days – Lead time goal in days, 0) + Finished goods offset in days + Optimizer finished goods offset in days] * *Standard cost for finished goods* * *Inventory carrying cost* / 365 days	$/part
In-house inventory carrying cost	*Raw material inventory carrying cost* + *WIP inventory carrying cost* + *Finished goods inventory carrying cost* (Same as corresponding entry in Table 6)	$/part
Subtotal of variable costs	Same as corresponding entry in Table 6.	$/part
Interplant freight	Same as corresponding entry in Table 6.	$/part
Interplant duty	Same as corresponding entry in Table 6.	$/part
Interplant pipeline inventory carrying cost	Same as corresponding entry in Table 6.	$/part
Tax holiday or charge	Same as corresponding entry in Table 6.	$/part
Pretax equivalent of Tax holiday or charge	Same as corresponding entry in Table 6.	$/part
Total fixed costs	Same as corresponding entry in Table 6.	$
Total variable costs (For source node to transshipment node arcs)	(*Subtotal of variable costs* + Optimizer cost offset + *Interplant freight* + *Interplant duty* + *Interplant pipeline inventory carrying cost* + *Pretax equivalent of Tax holiday or charge*) * Number of parts per system	$/parts-set
Total variable costs (For source node to demand node arcs)	*Subtotal of variable costs* + Optimizer cost offset + *Interplant freight* + *Interplant duty* + *Interplant pipeline inventory carrying cost* + *Pretax equivalent of Tax holiday or charge*	$/part

[Φ] In this column, cost categories that also appear in Table 2 are in bold italic. All other equations are in plain italic (non-bold).
[#] In this column, names that appear in italic refer to costs whose equation is defined in this table.

Table 9. Optimizer Sub-Tool Cost Equations Associated with Transshipment Node to Demand Node Arcs

Cost$^{\Phi}$	Equation and/or Description$^{\#}$	Units
Other raw material total cost	Same as corresponding entry in Table 7.	$/system
Inbound Freight	Same as corresponding entry in Table 7.	$/system
Inbound duty	Same as corresponding entry in Table 7.	$/system
Inbound inventory carrying cost for cost of money	Same as corresponding entry in Table 7.	$/system
Inbound inventory carrying cost for cost of revaluation & obsolescence	Same as corresponding entry in Table 7.	$/system
Inbound inventory carrying cost	Same as corresponding entry in Table 6.	$/system
Cost by placement	N.A.	-
Cost by standard hour	Same as corresponding entry in Table 6.	$/system
Other costs	Same as corresponding entry in Table 6.	$/system
New product start-up cost	Total new product start-up cost	$
Depreciation	Same as corresponding entry in Table 8.	$
Standard cost for raw materials	Same as corresponding entry in Table 7.	$/system
Standard cost for WIP	Same as corresponding entry in Table 7.	$/system
Standard cost for finished goods	Same as corresponding entry in Table 7.	$/system
Safety stock units	N.A. because calculation is nonlinear.	-

Cost[Φ]	Equation and/or Description[#]	Units
Additional WIP units due to skew, *Additional finished goods units due to skew*	N.A. because calculation is nonlinear.	-
Inventory carrying cost	Same as corresponding entry in Table 6.	%
Raw material inventory carrying cost	Same as corresponding entry in Table 8.	$/system
WIP inventory carrying cost	Same as corresponding entry in Table 8.	$/system
Finished goods inventory carrying cost	Same as corresponding entry in Table 8.	$/system
In-house inventory carrying cost	Same as corresponding entry in Table 8.	$/system
Subtotal of variable costs	Same as corresponding entry in Table 7.	$/system
Regional freight	Cost per system per pound * (Other raw material weight in pounds + AVERAGE(part weight in pounds) * number of parts per system)	$/system
Regional duty	% duty on inbound finished goods * % regional sales * (AVERAGE(*Standard cost for finished goods*) + *Interplant freight)*	$/system
Regional pipeline inventory carrying cost	AVERAGE(*Standard cost for finished goods*) * Pipeline days for cost of money * Cost of money / 365 days + AVERAGE(*Standard cost for finished goods*) * Pipeline days for revaluation and obsolescence * Cost of revaluation and obsolescence / 365 days	$/system
Tax holiday or charge	Same as corresponding entry in Table 6.	$/system
Pretax equivalent of Tax holiday or charge	Same as corresponding entry in Table 6.	$/system
Total fixed costs	Same as corresponding entry in Table 6.	$
Total variable (arc) costs	*Subtotal of variable costs* + Optimizer cost offset + *Regional freight* + *Regional duty* + *Regional pipeline inventory carrying cost* + *Pretax equivalent of Tax holiday or charge*	$/system

[Φ] In this column, cost categories that also appear in Table 2 are in bold italic. All other equations are in plain italic (non-bold).

[#] In this column, names that appear in italic refer to costs whose equation is defined in this table.

Appendix 2

2.1 Mixed-Integer Linear Program Equations Used in the Optimizer Results Worksheet

Indices

$i \in \{1, 2, 3, 4, 5, 6\}$ = source nodes
$j \in \{1, 2, 3, 4, 5\}$ = transshipment nodes
$k \in \{1, 2, 3\}$ = demand nodes

Decision Variables

- $PartsSetShipment_{ij}$. These variables represent the flow of parts-sets between source nodes and transshipment nodes.
- $PartShipment_{ik}$. These variables represent the flow of parts between source nodes and demand nodes.
- $SystemShipment_{jk}$. These variables represent the flow of systems between transshipment nodes and demand nodes.
- $SourceNode_i$. These variables are binary variables (0-1). If this variable equals 1, then the source node is in use. If this variable equals 0, then the source node is not in use.
- $TransshipmentNode_j$. These variables are binary variables (0-1). If this variable equals 1, then the transshipment node is in use. If this variable equals 0, then the transshipment node is not in use.

Miscellaneous Variables

- $PartsSetVariableCost_{ij}$. These variables represent the arc costs between source and transshipment nodes.
- $PartVariableCost_{ik}$. These variables represent the arc costs between source and demand nodes.
- $SystemVariableCost_{jk}$. These variables represent the arc costs between transshipment nodes and demand nodes.
- $SourceNodeFixedCost_i$. These variables represent the fixed costs associated with source nodes.
- $TransshipmentNodeFixedCost_j$. These variables represent the fixed costs associated with transshipment nodes.

Objective Function

$$Minimize\left[\sum_{i=1}^{6}\sum_{j=1}^{5} PartsSetShipment_{ij} * PartsSetVariableCost_{ij}\right.$$
$$+\sum_{i=1}^{6}\sum_{k=1}^{3} PartShipment_{ik} * PartVariableCost_{ik}$$
$$+\sum_{j=1}^{5}\sum_{k=1}^{3} SystemShipment_{jk} * SystemVariableCost_{jk}$$
$$+\sum_{i=1}^{6} SourceNode_{i} * SourceNodeFixedCost_{i}$$
$$\left.+\sum_{j=1}^{5} TransshipmentNode_{j} * TransshipmentNodeFixedCost_{j}\right]$$

Non-negativity Constraints

$$PartsSetShipment_{ij} \geq 0 \quad \forall\ i, j$$
$$PartShipment_{ik} \geq 0 \quad \forall\ i, k$$
$$SystemShipment_{jk} \geq 0 \quad \forall\ j, k$$

Integer (0-1) Constraints

$$SourceNode_{i} \in \{0, 1\}$$
$$TransshipmentNode_{j} \in \{0, 1\}$$

Transshipment Node Balance Constraints

$$\sum_{i=1}^{6} PartsSetShipment_{ij} = \sum_{k=1}^{3} SystemShipment_{jk} \quad \forall\ j$$

Demand Constraints

$$\sum_{i=1}^{6} PartShipment_{ik} \geq PartDemand_{k} \quad \forall\ k$$
$$\sum_{j=1}^{5} SystemShipment_{jk} \geq SystemDemand_{k} \quad \forall\ k$$

Capacity Constraints

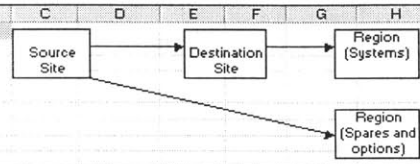
$$\sum_{j=1}^{5} PartsSetShipment_{ij} * NumberPartsInSet_{j} + \sum_{k=1}^{3} PartShipment_{ik}$$
$$\leq \quad SourceNode_{i} * SourceNodeCapacity_{i} \quad \forall\ i$$

$$\sum_{k=1}^{3} SystemShipment_{jk} \leq TransshipmentNode_{j} * TransshipmentNodeCapacity_{j} \quad \forall\ j$$

2.2 Numerical Example Using the Optimizer Sub-tool

Figure 7. Optimizer Results Worksheet

COMPAQ Sourcing Model



Legend
Solver Output
Arc Costs

Notes
This integer linear program is a one period model

Source Site to Destination Site Variable Costs Per Parts-set (parts kit)

Source	Destination 1	2	3	4	5	Source Site Fixed Costs
1	$800	$1,000	$1,200	$1,000	$900	$35,000
2	$700	$500	$700	$900	$600	$45,000
3	$800	$600	$500	$500	$700	$40,000
4	$500	$600	$700	$800	$600	$42,000
5	$700	$600	$500	$500	$500	$40,000
6	$600	$500	$800	$700	$500	$42,000

Source Site to Region Variable Costs Per Part

Source	Region 1	2	3
1	$800	$1,000	$1,200
2	$700	$500	$700
3	$800	$600	$500
4	$500	$600	$700
5	$700	$600	$500
6	$600	$700	$800

Destination Site to Region Variable Costs Per System

Destination	Region 1	2	3	Destination Site Fixed Costs
1	$40	$80	$90	$40,001
2	$70	$40	$60	$20,001
3	$80	$30	$50	$60,001
4	$85	$25	$40	$50,001
5	$90	$20	$30	$30,001

Source Site to Destination Site Parts-sets (parts kits) Shipments / **Total Source Site Parts Shipments**

Source	Destination 1	2	3	4	5	Shipped	Total Source Site Shipped		Logical Upper Bound	Use Plant?	Capacity
1	0	0	0	0	0	0	0	<=	0	0	600
2	0	204	0	0	0	204	504	<=	600	1	600
3	0	0	0	0	0	0	0	<=	0	0	600
4	0	0	0	0	0	0	200	<=	600	1	600
5	0	0	0	0	448	448	600	<=	600	1	600
6	0	0	0	0	0	0	0	<=	0	0	600
Received	0	204	0	0	448						
	=	=	=	=	=						
Shipped	0	204	0	0	448						

Source Site to Region Parts Shipments

Source	Region 1	2	3	Shipped
1	0	0	0	0
2	0	300	0	300
3	0	0	0	0
4	200	0	0	200
5	0	0	152	152
6	0	0	0	0
Received	200	300	152	
	>=	>=	>=	
Required	200	300	152	

Destination Site to Region System Shipments / **Total Destination Site System Shipments**

Destination	Region 1	2	3	Shipped	Total Destination Site		Logical Upper Bound	Use Plant?	Capacity
1	0	0	0	0	0	<=	0	0	600
2	200	4	0	204	204	<=	600	1	600
3	0	0	0	0	0	<=	0	0	600
4	0	0	0	0	0	<=	0	0	600
5	0	296	152	448	448	<=	600	1	600
Received	200	300	152						
	>=	>=	>=						
Required	200	300	152						

Summary of Costs

Source Site to Destination Site Costs	$326,000
Source Site to Region Costs	$326,000
Destination Site to Region Costs	$24,640
Source Site Fixed Costs	$127,000
Destination Site Fixed Costs	$50,002
Total Cost	$853,642

In this example, we will model the supply chain for printed circuit boards (PCB). These boards become part of a computer server at the assembly plants.

The Excel spreadsheet shown in Figure 7 is an exact facsimile of the Optimizer sub-tool. Note that in this example, the word *systems* as seen in Figure 7 is equivalent to *computer servers*. Also, the term *destination site*, which is used throughout the tool, is equivalent to *transshipment node*.

Table 10, Table 11, and Table 12 show the decision variables, objective function, and constraints, respectively, that specify the mixed-integer linear program. The cell references and equations shown in these tables are what appear in the Solver Parameters dialog box shown in Figure 8.

Table 10. Decision Variables (*By Changing Cells*) as Specified within Solver

Cells	**Description**
C43:G48	Flow between parts plants and assembly plants in units of PCB-sets (kits).
C56:E61	Flow between parts plants and demand regions in units of PCB's.
C69:E73	Flow between assembly plants and demand regions in units of servers.
M43:M48	Binary variable (0-1). Equals 1 if parts plant is used. Equals 0 if parts plant is not used.
M69:M73	Binary variable (0-1). Equals 1 if assembly plant is used. Equals 0 if assembly plant is not used.

Table 11. Objective Function (*Set Target Cell*) as Specified within Solver

Cells	**Description**
D84 = SUMPRODUCT(C14:G19,C43:G48) + SUMPRODUCT(C24:E29,C56:E61) + SUMPRODUCT(C34:E38,C69:E73) + SUMPRODUCT(I14:I19,M43:M48) + SUMPRODUCT(I34:I38,M69:M73)	• The sum of the first three terms in this equation represents the total variable cost. • The sum of the last two terms in this equation represents the total fixed cost. • In the region of the supply chain network between the parts (PCB) plants and the assembly plants, the

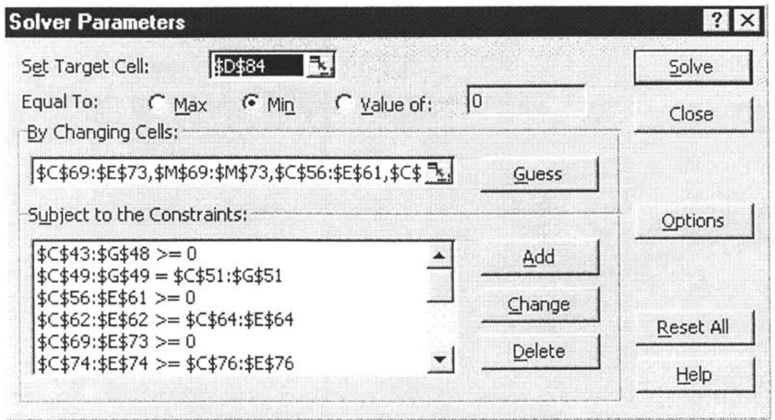
Cells	Description
	variable cost for each arc is computed by multiplying the arc's per unit cost by its associated flow (the number of units traversing the arc). The sum of these arc variable costs gives the total variable cost between the parts plants and the assembly plants. This total is the first term in the objective function. • In the region of the supply chain network between the parts (PCB) plants and the demand regions, the variable cost for each arc is computed by multiplying the arc's per unit cost by its associated flow (the number of units traversing the arc). The sum of these arc variable costs gives the total variable cost between the parts plants and the demand regions. This total is the second term in the objective function. • In the region of the supply chain network between the assembly plants and the demand regions, the variable cost for each arc is computed by multiplying the arc's per unit cost by its associated flow (the number of units traversing the arc). The sum of these arc variable costs gives the total variable cost between the assembly plants and the demand regions. This total is the third term in the objective function. • The fixed cost for each parts (PCB) plant is multiplied by a corresponding binary variable that indicates whether the plant is being used (1) or is not being used (0). The sum of these fixed costs gives the total fixed costs for the entire group of parts plants. This total is the fourth term in the objective function. • The fixed cost for each assembly plant is multiplied by a corresponding binary variable that indicates whether the plant is being used (1) or is not being used (0). The sum of these fixed costs gives the total fixed costs for the entire group of assembly plants. This total is the fifth term in the objective function.

Table 12. Constraints (*Subject to the Constraints*) as Specified within Solver

Cells	Description
C43:G48 >= 0	This is a non-negativity constraint. In the region of the supply chain network between the parts (PCB) plants and the assembly plants, the number of PCB-sets that are shipped across each arc must be greater than or equal to zero.
C56:E61 >= 0	This is a non-negativity constraint. In the region of the supply chain network between the parts (PCB) plants and demand regions, the number of individual PCB's that are shipped across each arc must be greater than or equal to zero.
C69:E73 >= 0	This is a non-negativity constraint. In the region of the supply chain network between the assembly plants and demand regions, the number of individual servers that are shipped across each arc must be greater than or equal to zero.
M43:M48 = integer M43:M48 >= 0 M43:M48 <= 1	These constraints force these variables to be either 0 or 1 (binary variables). These decision variables indicate whether a parts (PCB) plant is being used (1) or is not being used (0).
M69:M73 = integer M69:M73 >= 0 M69:M73 <= 1	These constraints force these variables to be either 0 or 1 (binary variables). These decision variables indicate whether an assembly plant is being used (1) or is not being used (0).
C49:G49 = C51:G51	Since assembly plants are transshipment nodes (material does not accumulate in these nodes), each plant must ship a server for each PCB set it receives.
C62:E62 >= C64:E64	The number of individual PCB's shipped to each demand region must be greater than or equal to the number of PCB's required by each demand region.
C74:E74 >= C76:E76	The number of individual servers shipped to each demand region must be greater than or equal to the number of servers required by each demand region.
H69:H73 <= L69:L73	The number of servers shipped by an assembly plant must be less than or equal to its capacity. This constraint is applied to each assembly plant.
J43:J48 <= L43:L48	The number of PCB's shipped by a parts (PCB) plant must be less than or equal to its capacity. This constraint is applied to each parts plant.

The demand in each region, plant variable costs, and plant fixed costs are entered via the Cost Model Input worksheet. The numbers that appear in Figure 7 are fabricated numbers. Solver can be invoked by clicking on the "Tools" menu and selecting "Solver." Figure 8 appears, and clicking the "Solve" button executes Solver.

Figure 8. Solver Parameters Dialog Box



Once Solver finishes executing, Figure 6 appears. Click on "OK" and cell D84 will show the minimum cost solution to be $853,642. The arc flows that achieve this minimum cost are shown in cells C69:E73, C56:E61, and C43:G48. For example, parts plant #2 needs to ship 204 sets of printed circuit boards (one set of boards is needed per server) to assembly plant #2. Parts plant #4 needs to ship 200 individual printed circuit boards to demand region #1. Assembly plant #5 needs to ship 296 servers to region #2.

References

Arntzen, Bruce C., et al. "Global Supply Chain Management at Digital Equipment Corporation." Interfaces, Volume 25, Number 1, January-February 1995, pp. 69-93.

Flystra, Daniel L., et al. "Design and Use of the Microsoft Excel Solver." Interfaces, Volume 28, Number 5, September-October 1998, pp. 29-55.

Fowler, H. Ramsey. The Little, Brown Handbook. Boston: Little, Brown and Company, 1980.

Glover, Fred, et al. Network Models in Optimization and Their Applications in Practice. New York: John Wiley & Sons, Inc., 1992.

Graves, Stephen. Notes from a class entitled "Operations Management: Models & Applications," Massachusetts Institute of Technology, Spring 1998.

Graves, Stephen. Notes from a class entitled "System Optimization and Analysis for Manufacturing," Massachusetts Institute of Technology, Summer 1997.

Hayes, Robert H. and Steven C. Wheelwright. Restoring Our Competitive Edge. New York: John Wiley & Sons, 1984.

Higgins, Robert C. Analysis For Financial Management. 5th ed. Boston: Irwin McGraw-Hill, 1998.

Hoover's Online, "Compaq Computer Corporation: Company Capsule." www.hoovers.com, March 10, 1999.

Hopp, Wallace J. and Mark L. Spearman. Factory Physics, Foundations of Manufacturing Management. Boston: Irwin McGraw Hill, 1996.

Nahmias, Steven. Production and Operation Analysis. 3rd ed. Chicago: Richard D. Irwin, 1997.

Rosenfield, Donald. Notes from a class entitled "Introduction to Operations Management," Massachusetts Institute of Technology, Summer 1997.

Winston, Wayne L. and Christian S. Albright. Practical Management Science, Spreadsheet Modeling and Applications. Belmont, CA: Wadsworth Publishing Company, 1997.