SYSPOINT: Unit of Measure for IT Infrastructure Project Sizing

Srinivasa Raghavan[1] and Veeraswamy Achanta[2]

[1]President, Krea Corporation, IL, USA
[2]President, Rangasoft, IL, USA

What can be done to improve the success rate of IT infrastructure projects? The Standish group considers an IT project successful when it is completed on time and on budget, with all the features and functions originally specified. Thus, time, cost, and scope – the triple constraint of project management is measured for success. The initial project estimation directly results in constructing the baseline for the two success parameters – time and cost. The traditional software estimation models uses lines of code and function points as sizing unit of measure. The IT infrastructure projects are significantly different from software development projects to use the software sizing techniques. This paper defines the concept of size for IT infrastructure projects. Specifically, the project size for IT infrastructure projects is measured in terms of the eight factors (server, workstation, printer, LAN, WAN, handheld, server applications, and client applications), infrastructure related software components, and the complexities defined based on the physical and functional categories of those factors.

Keywords: information technology, project sizing, infrastructure, infrastructure project sizing, syspoint.

1. Introduction

In this competitive market environment, companies try to differentiate their offerings through their IT infrastructure (Liu, 2002). For the purpose of this study, IT infrastructure is defined as the hardware used to interconnect computers and users and the software used to manage the infrastructure. The advancement in the technology and the availability of cost effective hardware and networking options have encouraged companies to expand and enrich their IT infrastructure. Liu (2002) suggest that the IT infrastructure even though important, remains one of the poorly defined areas. The importance is given more to IT software development projects compared to that of the IT infrastructure projects. IT infrastructure projects are generally considered subsets of large software development or package implementation initiatives. While this was true 15 years ago, the reduction in the costs of hardware, innovation of new devices, fast and affordable telecommunication medium has led to the increase in IT infrastructure projects. The need for a reliable, consistent, and quantitative model to size and estimate effort for infrastructure projects is evident. It is important to perform project sizing prior to estimating the effort of a project (Peters, 1999). The project estimation for these IT infrastructure projects is performed primarily from a cost perspective. The size and effort of the IT infrastructure projects are primarily estimated using qualitative techniques like expert judgment or analogy approach.

Project sizing can be performed by quantifying the physical characteristics of a project or by the logical characteristics of a project. In traditional software development sizing models, project size is measured either in terms of lines of codes or functionalities that need to be performed, while the project complexity is qualitative based on the each functionality provided by the project. The logical characteristics of the project are determined by the project manager by looking at what the project does in terms of requirements, queries, inputs, outputs and interfaces. The project sizing function applies several characteristics of a project for its input to deduce the size of the project. The project size can then be used as an input to the estimation equation to deduce the effort of the project.

In the remainder of this paper, the literature of project sizing is discussed first. Next, the paper presents the methodology used to define a project size for IT infrastructure projects. The subsequent sections discuss the method to calculate IT infrastructure project size and the advantages of such measure.

2. Concept of Project Sizing

It is important to perform project sizing prior to estimating the effort of a project (Peters, 1999). The Standish group CHOAS report identifies inaccurate estimates as one of the key reasons of project failure (Johnson, Boucher, Connors, and Robinson, 2001). Sizing is the key input for all effort and costing models. Hence, determining the accurate size of a project is the requirement for precise cost and schedule estimation. The calculation of the project size can be extremely challenging at the early stages of the project. Project sizing determines how big a project is, based on some unit of measure. Prior to the mid 1970's, qualitative techniques like expert judgment were used in determining the size of a project (Ferens, 1999). There was no commonly accepted unit of measure to determine project size. The primary focus was given to software development projects. The programming languages used at that period of time influenced the development of sizing models. The prevalent use of third generation languages like COBOL during the 1970's resulted in use of a sizing technique called "Lines of Code". As the name indicates, the "lines of code" is based on the number of lines of code that will be written for a software project. This technique is still used as a sizing measure for various software costing models. The key advantage of this method is its ease of use and understanding.

Even though this sizing technique is popular, it has a lot of drawbacks too. There is not a single standard definition on how to count the lines of code. As long as the counting technique is consistent within an organization, it can be used to compare between projects and for other benchmarking purposes. However, comparing two projects between organizations using "Lines of Code" sizing technique requires several assumptions to be made. The major disadvantage of this technique is its dependence on the lines of code. Coding is just a phase of the software development life cycle. It does not account for the program complexity, vague requirements, bad design, and the choice of the life cycle model. It is also difficult to estimate the lines of code at the early stages of the project life cycle. If the requirement of the project is not detailed enough, sizing that project using this technique would be difficult and is as good as a speculation.

There are several variations of the technique like counting only the executable lines, counting executable lines plus data definitions, counting executable lines plus data definitions and comments (Jones, 1986). More variations are possible based on the type of the project. If a project is an enhancement of an existing project, the calculation of the lines of code can take variations like, counting only new lines, counting new lines and changed lines, counting new lines, changed lines and reused lines. The "lines of code" technique depends on the programming language. If a project uses multiple programming languages, then the counting gets complicated. In spite of these drawbacks, "lines of code" is a widely used sizing technique even today!

The drawbacks of "lines of code" technique and its dependence on the programming language gave rise to another sizing technique called the "function point" technique. The concept of "function point" technique is to focus on the function requirements of the project to determine the project size as against the lines of code. This idea was developed by Albrecht, an engineer at IBM (Dreger, 1989). The current counting practices are published in a manual maintained by the organization called International Function Point Users Group (IFPUG). This manual defines "function point" technique as "measurement of software by quantifying the functionality the software provides to the user based primarily on the logical design" (Function point counting practices manual, 2000, p. 3). The basis of function point analysis is to count function points that can be classified into five broad categories – External Input, External Output, Internal Logical File, External Interface File, and External Inquiry (Dreger, 1989).

The advantage of "function point" technique is its independence of the programming language. This technique can be applied during the requirements stage of the project even with vague requirements. The "function point" technique also has disadvantages. The rationale for the assigned weights in calculating unadjusted function points is not documented. Hence some organizations assign their own weights or sometimes ignore the weights while calculating the unadjusted function points. It has been proved that not all of the five functional types are required to be counted in calculating the unadjusted function points (Function point counting practices manual, 2000). This technique is also oriented towards the traditional data processing applications and cannot be applied to real time systems or scientific software (Abran, Maya, and Desharnais, 1997).

Several extensions to the function point measure were proposed in the past two decades. The most popular of these extensions is the feature point analysis. It enables the function point measure to be applied to systems software and real-time applications. "Feature point" analysis includes a function type called algorithms in addition to the five function types identified by function point (Jones, 1986). Thus applications that have high algorithmic complexity such as real-time systems, process control applications, embedded software applications etc, can be estimated using feature point analysis.

The object-oriented development methodology is significantly different such that traditional sizing measurements appear meaningless when applied. The popularity of the object-oriented design and development tools have resulted in a significant amount of research on estimating object-oriented projects.

One of the significant researches on estimating object-oriented projects was performed by Chidamber and Kemerer. They have identified six different sizing metrics for object-oriented projects – Weighted methods per class, depth of inheritance tree, number of children, coupling between object classes, response for a class, and lack of cohesion in methods (Chidamber and Kemerer, 1994).

The review of the software development sizing models highlights the fact that sizing is performed by counting the deliverables that can be quantified by the end users or the developers. In the case of "lines of code" technique, it was the size recognized by the developers. Function point and feature point analyses focused primarily on the deliverables to the end user. The object point analysis focused on the number of objects that are recognized by the developers.

Extending this approach, the sizing of IT infrastructure projects could be performed by counting the tangible hardware and network components of the projects that are recognized both by the end users and infrastructure technicians. The sizing technique for IT infrastructure projects can be constructed similar to that of the function point analysis to include both the function and the complexity of the project.

In this section, the concept of project sizing and the software development sizing models were discussed. The next section discusses the research methodology used to construct a sizing model for IT infrastructure projects.

3. Research Methodology

Qualitative research methodology was used to develop a sizing model for IT infrastructure projects. As a pilot, an expert interview was used to identify the factors that have an impact on the sizing of an infrastructure project. Further, a focus group of IT infrastructure experts were gathered to verify the sizing factors, develop a complexity matrix and the sizing model for IT infrastructure projects. The key research questions were:

- What are the factors that affect the sizing of IT infrastructure projects?
- What are the functional and physical characteristics of the factors identified?
- What are the quantitative weights categorized by complexity for the characteristics of the factors identified?

The selection criteria for the expert interview was 10 years of IT infrastructure project management (including project effort and cost estimation) experience and a minimum of 25 IT infrastructure project management experience. The selection criterion was relaxed for the focus group experts – 3 years of IT infrastructure project management experience and a minimum of 5 IT infrastructure project management experience.

The expert who participated in the interview was also member of the focus group research.

The need for participants was advertised using professional contacts, and the network of professional organizations. Each participant was required to submit their resume and participants were screened based on their willingness to participate, commitment to provide timely and active feedback and experience in the research topic. A total of eight experts provided their willingness to participate in the survey and six of them were selected to participate in the focus group. Two of them did not meet the time commitments required to participate in the focus group.

Expert A is a Senior Principal for an IT consulting firm with twelve years of IT infrastructure project management experience and has managed $30 million worth of IT infrastructure projects. This expert also is Cisco, Microsoft, Novell, Comptia, and Citrix certified systems engineer. Expert B is a Principal for an IT consulting firm with seven years of IT infrastructure project management experience and has managed about $20 million worth of IT infrastructure projects. This expert is Cisco, UNIX, and Apple certified systems engineer. Expert C is a President of a consulting firm with 13 years of IT infrastructure project management experience. This expert has managed more than 75 network installation projects utilizing Microsoft, Apple and Citrix technologies. Expert D is an IT Director for a Federal Government Agency. This expert has managed more than 10 government IT infrastructure projects utilizing mainframes, Microsoft and UNIX technologies. Expert E is a network architect in an IT consulting firm with ten years of IT infrastructure project management experience. This expert is Cisco, IBM, and Microsoft certified systems engineer. Expert F is an Engagement Manager with 15 years of IT infrastructure project management experience. This expert has extensive experience in qualitative IT infrastructure project estimating.

Clear and open ended questions were presented to promote brain storming and also to challenge the arguments in order to draw out the differences between the opinions and experience. Proper analysis was performed to check if the data gathered are relevant to the topic of discussion. As mentioned earlier, one the big challenges of this technique was to maintain the focus of the discussion. Conversations easily tend to go in tangents and it was important to shift the focus to the objective. Good interpersonal skills, good analytical skills, non-judgmental approach, and flexibility of the researcher were key leadership traits that came in handy during the focus group research. These qualities resulted in the participants' trust and increased the likelihood of open and interactive brain storming discussions.

The objective of the focus group was to find the answers to the three key research questions. An initial conference call was held to provide orientation to the experts about the research topic, the research questions, the concept of sizing, and the sizing methods used in the software development area, the results of the pilot interview, and the timeframe. A PowerPoint presentation was developed for the purposes of this call, the research questions, and the results of the expert interview were sent to the focus group experts in advance. This helped to keep the discussion focused and work towards the common goal.

The focus group was conducted using an online method. An online discussion group system called "intranets.com" was used as the vehicle to launch this study. The focus group was conducted for a period of four weeks. The results of the research were validated by the experts during the fifth week.

The findings of the focus group research are presented in the following section. The experts identified the factors that impact the sizing of IT infrastructure projects, the functional and physical characteristics of those factors, categorized them according to their complexity, and assigned weights by complexity. This structure was synonymous to that of the function point model. This research identified a new sizing unit of measure called *syspoint*.

4. Research Findings

The focus group identified the factors that impact the sizing of IT infrastructure projects and categorized them by respective complexities. In addition, quantitative weights were assigned to each of the complexity. To categorize the factors by complexity, other sizing models were studied. COCOMO model uses Very Low, Low, Nominal, High, and Very High as the qualitative complexity grouping. All other sizing models use Low, Average, and High as the qualitative complexity grouping. Initially, the experts decided to keep both options open and try to categorize the factors by very low, low, nominal, high, and very high categories. However, after preliminary categorizations, it was evident that the differences between the categories are negligible and it made sense to use low, average, and high as the categories. The experts identified eight key factors – server, server application, workstation, workstation application, printer, local area network, wide area network, and handheld application as factors that have an impact on IT infrastructure project sizing. There certainly can be several other general project-related factors that may have an impact on project sizing. Since this is one of the first researches carried out in this area of knowledge, it was decided to keep the scope simple and focused.

4.1. Sizing Process

The sizing process used by the focus group research is depicted in Figure 1. Each factor is defined and categorized by complexities using a matrix structure. The rows of the complexity matrix represent the physical categorization and the columns of the complexity matrix represent logical or functional categorization.

After the complexity matrix was established for each factor, the experts then defined the sizing unit for each of the complexity within each factor. The sizing unit for IT infrastructure projects was aptly named *SYSPOINT*. The experts decided to use the installation of a hub (unmanaged switch) as one syspoint. Using the installation of one hub (or one syspoint) as the base, the other complexities within and between the sizing factors were identified. Thus, the syspoint weights were carefully assigned by making sure that the weights assigned within and between the categories were proper and agreed to by all the experts.

The number of factors under each complexity is counted and multiplied by their corresponding syspoint weight to arrive at the total unadjusted syspoint for that factor. The total unadjusted syspoint for each factor is then summed to arrive at the total unadjusted syspoint for the project. An adjustment for the operating system is performed finally to arrive at an adjusted syspoint for the project. The server factor is explained in detail and to keep the paper focused and simple, the sizing for other factors are grouped into tables.

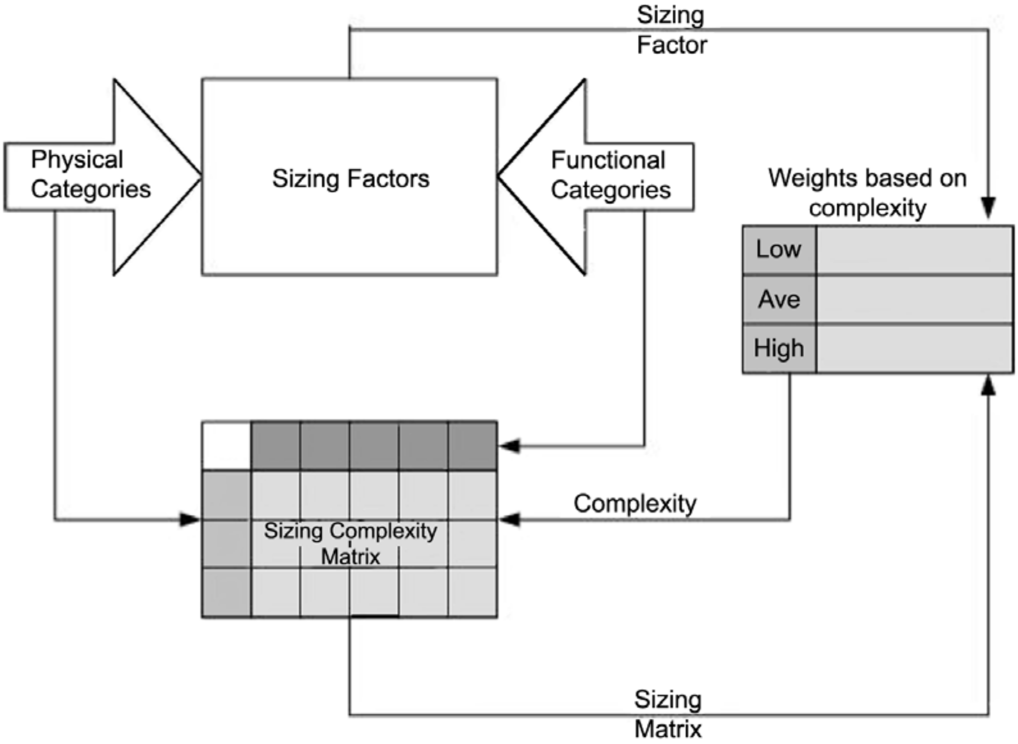

Fig. 1. Project Sizing Process.

4.2. Server Factors

For the purpose of this study, the experts defined servers as “A computer system in a network that is shared by multiple users. Servers come in all sizes from x86-based PCs to IBM mainframes” (Techweb, 2003). The physical and functional categories of the server factor are presented as rows and columns respectively in the following complexity matrix.

The single server represents categories where only one server is installed. The network load balancing servers implement a software scaling technology that spreads the client requests among servers that are connected to support a particular application (Techweb, 2003). In defining the Cluster, one of the experts, referred to the following definition from Techweb (2003) “Using two or more computer systems that work together. It generally refers to multiple servers that are linked together in order to handle variable workloads or to provide continued operation in the event one fails. Each computer may be a multiprocessor system itself. For example, a cluster of four computers, each with four CPUs, would provide a total of 16 CPUs processing simultaneously”.

The functional categories include new installation, upgrade, and migration. New installation covers those projects when the installation of the server is performed from the scratch. Server upgrade refers to the upgrade of different versions of the same operating system. For example, an upgrade of Windows 2000 from Windows NT is considered a server upgrade. Patch upgrades like Windows Service Packs are not covered by this sizing. They are considered normal maintenance features and not installation projects. Server migration refers to the migration from one operating system to another operating system. For example, a migration from an UNIX Server to a LINUX Server or a migration from a Novell Server to a Windows 2000 Server is covered by this classification.

Table 1 represents the complexity matrix for the server factors. Using the physical categorization as the row and the functional categorization as the column, the cells of the matrix are filled with Low, Average, or High values. For example, a single server migration is considered an average complex factor and a cluster server upgrade is considered a high complex factor.

	New Installation	Upgrade	Migration
Single Server	Low	Low	Average
Network Load Module	Low	Average	High
Cluster	Average	High	High

Table 1. Server Complexity Matrix.

Table 2 represents the weights (syspoint) assigned to each complexity of the server factors. The weights are determined by the experts, based on their past experience. The weights are carefully assigned by taking into consideration their relationship within each sizing factor and between sizing factors. For example, the average complexity server is considered about 1.7 times bigger than the low complexity server. Similarly, a low complexity server is considered 3 times bigger than that of the average complexity workstation.

	Unadjusted Syspoint
Low complexity servers	15
Average complexity servers	25
High complexity servers	40

Table 2. Server unadjusted Syspoint.

Based on the above definition of the low, average, and high complexity servers, it is important that the total count of low, average, and high complexity servers is determined. Using the unadjusted syspoint from the table above, a total server unadjusted syspoint (λ_S) is then calculated by multiplying the count of low, average, and high complexity servers with the corresponding unadjusted syspoint multiplier.

4.3. Workstation Factors

For the purpose of this study, the experts defined workstation as “A computer system in a network that could be shared by multiple users or stand alone to be used by a single user. Workstations come in all sizes from dumb terminals to thin clients, to personal desktop computers, to Midrange and Mainframe Workstations" (Techweb, 2003). The physical categories of the workstation component were identified as dumb terminals, thin client-based terminals, and desktops or laptops. The functional categories were identified as new installation, cloned image, upgrade, and migration.

A dumb terminal was defined by an expert as a "display terminal without any processing ability". Dumb terminals are entirely dependent on the main server for processing functions. Even though the mainframe and mini computer terminals have some independent capabilities, they are still categorized as dumb terminals.

Thin clients are similar to dumb terminals, but with some capabilities. Thin clients do not perform any application processing. They process only keyboard input and screen output and function like an input/output terminal. The application and functional processing are performed at the server level. However, the presentation and input mechanisms are performed at the terminal level (Techweb, 2003). Windows terminals and the X-Window system are examples of this type of thin client. The focus group experts referred desktops as a personal computer or a Macintosh or a UNIX workstation that has individual processing capabilities and is able support multiple devices that could be shared or used as local devices.

The functional categories include new installation, cloned image, upgrade, and migration. New installation covers those projects when the installation of the workstation is performed from the scratch. These categories are very similar to those of the server factors.

Table 3 represents the complexity matrix for the workstation factors. Using the physical categorization as the row and the functional categorization as the column, the cells of the matrix are filled with Low, Average, or High values. The table below (Table 4) represents the weights (syspoint) assigned to each complexity of the workstation factors.

	Unadjusted Syspoint
Low complexity workstation	15
Average complexity workstation	25
High complexity workstation	40

Table 4. Workstation unadjusted Syspoint.

Based on the above definition of the low, average, and high complexity workstations, it is important that the total count of low, average, and high complexity workstations are determined. Using the unadjusted syspoint from the above table, a total workstation unadjusted syspoint (λ_W) is then calculated by multiplying the count of low, average, and high complexity workstations with the corresponding unadjusted syspoint multiplier.

4.4. Printer Factors

The installation and configuration of the printer hardware is discussed in this section. The physical categories of the printer component were identified as local printer, and network printer. The functional categories were identified as standard, duplex, and multi-function.

The local printer represents categories where the printers are connected directly to a port of a workstation. A local printer can also be shared with other computers through the workstation's share capabilities. A network printer was defined by a focus group expert as follows: "When a printer is installed to be a network printer that can be shared with multiple workstations, it is

	New installation	Cloned Image	Upgrade	Migration
Dumb Terminal	Low	NA	NA	NA
Thin client	Low	Low	Average	Average
Desktops/Laptops	Average	Avg	High	High

Table 3. Workstation complexity matrix.

called a network printer". If a local printer is set up to be shared with other users, it should not be counted as network printer; it is still a local printer.

The functional categories include standard installation, duplex, and multi-function. Standard installation covers those projects with the installation of the printer with the default basic features that arrive with the printer. The duplex function refers to the back to back printing capability of a printer. This requires additional configuration and installation of additional units. The multi-function capability refers to the installation and configuration of devices that provide features like copier, fax, and/or scanner in the printer unit.

	Standard	Duplex	Multi function
Local	Low	Low	Average
Network	Low	Average	High

Table 5. Printer Complexity Matrix.

Table 5 represents the complexity matrix for the printer factors. Using the physical categorization as the row and the functional categorization as the column, the cells of the matrix are filled with Low, Average, or High values. The table below (Table 6) represents the weights (syspoint) assigned to each complexity of the server factors.

	Unadjusted Syspoint
Low complexity printers	2
Average complexity printers	4
High complexity printers	6

Table 6. Printer unadjusted Syspoint.

Based on the above definition of the low, average, and high complexity printers, it is important that the total count of low, average, and high complexity printers are determined. Using the unadjusted syspoint from the table above, a total printer unadjusted syspoint (λ_P) is then calculated by multiplying the count of low, average, and high complexity printers with the corresponding unadjusted syspoint multiplier.

4.5. Local Area Network Factors

Local area network (LAN) is a data communication network that serves users within a confined area – mostly a building or even a small area within the building. The network would generally encompass servers, workstations, printers, a network operating system and a communication link (Techweb, 2003). The physical categories of the local area network component were identified as "hubs" or unmanaged switches, managed switch without a VLAN, wireless VLAN, and managed switches with a VLAN. The functional categories were identified as new direct connection, routing, limited access, and firewall.

Techweb (2003) defines a hub as, "a central connecting device in a network that joins communication lines together in a star configuration". There are two types of hubs – active and passive. Passive hubs are just connecting hubs and add nothing to the data that are passed through them. Active hubs are multi-port regenerators that can regenerate the data bits to maintain the strong signal and can include some intelligence. An unmanaged switch is a network switch that does not support any statistics or management-related queries.

Managed Switch is a network switch that has the intelligence built into the firmware. These switches can respond to statistical and management queries and typically would require additional configuration for management and monitoring purposes.

Wireless local area networks transmit the data through unlicensed frequency such as 2.4 GHz band. The wireless local area network does not require lining up of devices like the infra red transmission. This typically requires installation of an access point (or a base station) that is physically connected to the network. This base unit can transmit signals to multiple wireless network cards using the radio frequency over a limited area, penetrating walls and other non-metal barriers.

Virtual local area network (VLAN) is a logical subgroup within a LAN that is established by software configuration instead of physical cables. One of the focus group experts defined its function as – “It combines user stations and network devices into a single unit regardless of the physical LAN segment they are attached to and allows traffic to flow more efficiently within populations of mutual interest” (Techweb, 2003).

The functional categories include direct connection, router, limited access and firewall. The direct connection refers to the simple plug-in connectivity to the router or the modem. The router refers to the presence of a device in the network to forward data packets from one local area network to another. The routing functional category represents simple routing entries without any customization.

The limited access function represents the customization of the routing tables and routing protocols that are read by the routers to translate the network address in each transmitted frame and make a decision on how to send it based on the most expedient route – based on parameters like traffic load, line costs, speed, and bad lines (Techweb, 2003).

The firewall is a mechanism for implementing security policies designed to keep a network secure from intruders. Firewalls installed to protect entire networks are typically implemented in hardware; however, software firewalls are also available to protect networks from attack (Techweb, 2003).

Table 7 represents the complexity matrix for the LAN factors. Using the physical categorization as the row and the functional categorization as the column, the cells of the matrix are filled with Low, Average, or High values. Table 8 represents the weights (syspoint) assigned to each complexity of the local area network factors.

Based on the above definition of the low, average, and high complexity LANs, it is important that the total count of low, average, and high complexity servers is determined. Using the unadjusted syspoint from the table above, a total LAN unadjusted syspoint (λ_{LAN}) is then calculated by multiplying the count of low, average, and high complexity printers with the corresponding unadjusted syspoint multiplier.

	Unadjusted Syspoint
Low complexity LANs	1
Average complexity LANs	2
High complexity LANs	5

Table 8. LAN unadjusted Syspoint.

4.6. Wide Area Network Factors

For the purpose of this study, the experts defined Wide area networks as “a communications network that covers a wide geographic area, such as state or country. A wide area network (WAN) generally covers a wide geographic area” (Techweb, 2003). Physical categories of the WAN component were identified as cable modems, ISDN/Frame relay, DS3/OC and wireless wide area networks. The functional categories were identified as direct connectivity, router, limited access and firewall.

A cable modem is a hardware used to connect a computer to a cable television service that provides internet access. Cable modems have become a popular choice of wide area network at small companies and for home network setups. They provide high bandwidth network connectivity between the user’s computer and the internet service provider. Cable modems link to the computer via Ethernet, which makes the service online all the time.

Integrated services digital network (ISDN) is an international telecommunication standard for providing a digital service from the customer’s premises to the dial-up network (Techweb, 2003). Frame relay is a high speed packet

	Direct	Routing	Limited Access	Fire-wall
Hubs/Unmanaged Switch	Low	N/A	N/A	N/A
Managed Switch without VLAN	Low	Average	High	High
Wireless LAN	Avg	Average	High	High

Table 7. LAN complexity matrix.

switching protocol and it has become a popular choice for LAN to LAN connections across remote distances.

A focus group expert defined DS3/OC as follows: "Digital Signal (DS3) is a classification of digital circuits. The DS technically refers to the rate and format of the signal, while the T designation refers to the equipment providing the signals. In practice, "DS" and "T" are used synonymously; for example, DS1 and T1, DS3 and T3. Optical carrier (OC) refers to the transmission speeds defined in the SONET specification. OC defines transmission by optical devices, and STS is the electrical equivalent" (Techweb, 2003).

Wireless wide area networks are very similar the wireless local area networks. In the case of wireless wide area network, the communication is carried out between buildings without any wiring. This technology is expected to explode and provide more options in the future.

The functional categories include direct connection, router, limited access and firewall. The descriptions of these functional categories are similar to those of the local area network factors.

Table 9 represents the complexity matrix for the wide area network factors. Using the physical categorization as the row and the functional categorization as the column, the cells of the matrix are filled with Low, Average, or High values. Table 10 represents the weights (syspoint) assigned to each complexity of the wide area network factors.

Based on the definition above of the low, average, and high complexity wide area networks, it is important that the total count of low, average, and high complexity wide area networks is determined. Using the unadjusted syspoint from the above table, a total wide area network unadjusted syspoint (λ_{WAN}) is then calculated by multiplying the count of low, average, and high complexity printers with the corresponding unadjusted syspoint multiplier.

	Unadjusted Syspoint
Low complexity WANs	2
Average complexity WANs	5
High complexity WANs	10

Table 10. WAN unadjusted Syspoint.

4.7. Server Application Factors

For the purpose of this study, the experts defined server applications as "the applications that are installed at the server to support tasks like Web, database, email and security". The physical categories of the server component were identified as file and print, web, database, email, and security. The functional categories were identified as new installation, upgrade, and migration.

The file and print server was defined by an expert as follows: "A file server is a high-speed computer in a network that stores the programs and data files shared by users. It acts like a remote disk drive. A print server is a computer in a network that controls one or more printers. It is either part of the network operating system or an add-on utility that stores the print-image output from users' machines and feeds it to the printer, one job at a time. The computer and its printers are known as a print server or a file server with print services" (Techweb, 2003).

The web server is a computer that provides World Wide Web services on the internet. A database server is a computer in a LAN dedicated to database storage and retrieval. The database server is a key component in a client/server environment. It holds the database management

	Direct	Routing	Limited Access	Fire-wall
Cable Modems	Low	Low	Average	High
ISDN/Frame relay	Average	Average	High	High
DS3/OC	Average	High	High	High
Wireless WAN	Average	High	High	High

Table 9. WAN complexity matrix.

system (DBMS) and the databases. Upon requests from the client machines, it searches the database for selected records and passes them back over the network. For example – installation and configuration of Oracle database on a LINUX Operating System.

An email server is a computer in a network that provides post office facilities. It stores incoming mail for distribution to users and forwards outgoing mail through the appropriate channel. The term may refer to just the software that performs this service, which can reside on a machine with other services. Example includes installation of Microsoft Exchange Server.

A security application that performs protection of the data against unauthorized access is generally termed as security program. The installation and configuration of such security applications like the Firewall, and Access Control applications fall into this category.

The functional categories include new installation, upgrade, and migration. New installation covers those projects where the installation and configuration of the application is performed from the scratch. Server application upgrade involves upgrade of different versions of the same application System. For example, an upgrade of Microsoft SQL Server 6 to SQL Server 7 is an upgrade installation. However, installations of patches or service packs are not covered by this sizing. They are considered normal maintenance features and not installation projects.

The migration function involves migration from one application system to another application system. For example, a migration from a Microsoft Exchange Server to a Lotus Notes Groupware is covered by this classification.

Table 11 represents the complexity matrix for the server application factors. Using the physical categorization as the row and the functional categorization as the column, the cells of the matrix are filled with Low, Average, or High values. Table 12 represents the weights (syspoint) assigned to each complexity of the server application factors.

	Unadjusted Syspoint
Low complexity Server apps	15
Average complexity Server apps	40
High complexity Server apps	60

Table 12. Server Application unadjusted Syspoint.

Based on the above definition of the low, average, and high complexity server applications, it is important that the total count of low, average, and high complexity server applications is determined. Using the unadjusted syspoint from the table above, a total server application unadjusted syspoint (λ_{Sa}) is then calculated by multiplying the count of low, average, and high complexity printers with the corresponding unadjusted syspoint multiplier.

4.8. Client Application Factors

For the purpose of this study, the experts defined client application factor as “the applications that are installed on the workstation to support client tasks like word processing, remote support, print and device software, and security”. The physical categories of the client application component were identified as a word processing/spreadsheet, remote support tools, database client, printer software, fax/scan/imaging software, video conferencing, security, email client,

	New Installation	Upgrade	Migration
File and Print	Low	Low	Average
Web	Low	Low	Average
Database	Average	Average	High
Email	Average	High	High
Security	High	High	High

Table 11. Server Application Complexity Matrix.

	New Installation	Upgrade	Migration
Word Processing and Spreadsheet	Low	Low	Low
Remote Support Tools	Low	Low	Average
Database Client	Low	Low	Average
Printer Software	Low	Low	Average
Fax/Scan/Imaging Software	Average	Average	High
Video Conferencing Software	Average	Average	High
Anti-Virus	Average	Average	High
Email Client	Average	High	High
Custom Application	High	High	High

Table 13. Client Application Complexity Matrix.

and custom application". The functional categories were identified as new installation, upgrade, and migration.

One of the experts defined the Word processing and spreadsheet category as follows: "Word Processing involves the creation of text documents. Example: Microsoft Word, and Word Perfect. Spreadsheet is the software that simulates a paper spreadsheet (worksheet), in which columns of numbers are summed for budgets and plans. It appears on screen as a matrix of rows and columns, the intersections of which are identified as cells. Example: Microsoft Excel, and Lotus 1–2–3" (Techweb, 2003).

Remote support tool is a software installed in both machines, that allows a user at a local computer to have control of a remote computer via modem. Both users run on the remote computer and see the same screen. Database client is a software that is provided by the database vendor and is required to be installed at the workstation for database connectivity purposes. Printer software is a software that is provided by the printer vendor and is required to be installed at the workstation for printer connectivity purposes. Fax or scan or imaging software is a software that is provided by the respective vendor and is required to be installed at the workstation for the connectivity purposes. Video conferencing is a software that is provided by the respective vendor and is required to be installed at the workstation for video conferencing purposes. Anti virus is a software that is provided by the security vendor and is required to be installed at the workstation for detecting and blocking computer viruses. Example: Installation of Norton antivirus utility, and MacAfee antivirus utility. Email client is a software that is provided by the respective vendor and is required to be installed at the workstation, so that the users can access the email servers from local or remote networks. Example: Installation and configuration of Microsoft Outlook client. Custom applications refer to the application that is specifically designed and programmed for an individual customer.

The functional categories include new installation, upgrade, and migration. The descriptions of these functional categories are similar to those of the server application factors.

Table 13 represents the complexity matrix for the client application factors. Using the physical categorization as the row and the functional categorization as the column, the cells of the matrix are filled with Low, Average, or High values. Table 14 represents the weights (syspoint) assigned to each complexity of the client application factors.

	Unadjusted Syspoint
Low complexity Client apps	4
Average complexity Client apps	7
High complexity Client apps	11

Table 14. Client Application unadjusted Syspoint.

Based on the above definition of low, average, and high complexity client applications, it is important that the total count of low, average, and high complexity client applications is determined. Using the unadjusted syspoint from the above table, a total client application unadjusted syspoint (λ_{Ca}) is then calculated.

4.9. Handheld Application Factor

For the purpose of this study, the experts defined handheld application as follows: “The custom applications that are installed in the handheld devices on top of the OEM installed operating system are referred to as handheld applications”. The physical categories of the handheld component were identified as a local network, database, and web based. The functional categories were identified as basic installation, phone, and wireless. The local network handheld application is stand-alone and requires simple local USB or Serial “Sync” based communication configuration. A database-related handheld application refers to those applications that involve storage and retrieval of information, both from a local database and from a database stored at a remote server. A web-based handheld application represents those applications that are web-based and uses wireless access protocols and architecture to display the information in the handheld. The customization of such applications may require additional steps and intensive testing procedures.

The functional categories include basic installation, phone, and wireless. Basic installation covers those projects that involve simple information input, and query application. The phone installation represents those applications that use the phone capabilities of the handheld. The wireless installation represents those applications that use the wireless capabilities of the handheld.

	Basic	Phone	Wireless
Local Network	Low	Low	Average
Database	Low	Average	High
Web-based	Average	High	High

Table 15. Handheld Application Complexity Matrix.

Table 15 represents the complexity matrix for the handheld application factors. Using the physical categorization as the row and the functional categorization as the column, the cells of the matrix are filled with Low, Average, or High values. Table 16 represents the weights (syspoint) assigned to each complexity of the handheld application factors.

	Unadjusted Syspoint
Low complexity servers	2
Average complexity servers	4
High complexity servers	7

Table 16. Handheld Application unadjusted Syspoint.

Based on the above definition of the low, average, and high complexity handheld application, it is important that the total count of low, average, and high complexity handheld application is determined. Using the unadjusted syspoint from the above table, a total handheld application unadjusted syspoint (λ_{Ha}) is then calculated.

4.10. Unadjusted Project Syspoint

A total unadjusted syspoint is calculated for each of the factors using the syspoint worksheet as the guideline. The sum of all those total unadjusted syspoints will result in the unadjusted project syspoint (Λ_{PROJ}). The formula for computing the unadjusted project syspoint is given below.

$$\begin{aligned}\Lambda_{\text{PROJ}} = \lambda_{\text{S}} + \lambda_{\text{W}} + \lambda_{\text{P}} + \lambda_{\text{LAN}} + \lambda_{\text{WAN}} \\ + \lambda_{\text{Sa}} + \lambda_{\text{Ca}} + \lambda_{\text{Ha}}.\end{aligned}$$

Where Λ_{PROJ} is the project unadjusted syspoint, λ_{S} is the total unadjusted server syspoint, λ_{W} is the total unadjusted workstation syspoint, λ_{P} is the total unadjusted printer syspoint, λ_{LAN} is the total unadjusted LAN syspoint, λ_{WAN} is the total unadjusted WAN syspoint, λ_{Sa} is the total unadjusted server application syspoint, λ_{Ca} is the total unadjusted client application syspoint, and λ_{Ha} is the total unadjusted handheld syspoint. The unadjusted syspoint allows comparison of projects without adjustment to the operating system into consideration.

4.11. Adjusted Project Syspoint

The operating systems play a large role in determining the size of the project. The unadjusted project syspoint provides the size to of the project without taking this factor into consideration. The purpose of the unadjusted project syspoint is to analyze and compare projects without any additional dimension added by the operating system.

Operating systems bring an additional dimension to the sizing equation. The operating system impacts the server, workstation, and handheld factors. For example, if an IT infrastructure project involves installation of 5 servers, and 2 out of 5 are low complex servers, 2 average complex servers and 1 high complex server. Assume that, of the 2 low complex servers, 1 is a Microsoft Windows operating system and the other is a UNIX operating system. Hence it will be difficult to use one categorization and apply it to the other. The sizing of IT infrastructure projects is really not a straight forward two dimensional matrix.

Server Operating System Adjustment Multiplier	
Microsoft Windows	1
Macintosh	1
Novell	1.25
UNIX	2.5
IBM OS/2	1.25
LINUX	1.5
IBM OS/400	3.5
VAX/VMS	3.5
IMB MVS	5
Client Operating System Adjustment Multiplier	
Microsoft Windows	1
Macintosh	1
OS/2	1
LINUX	1.25
UNIX X-Windows	2
Handheld Operating System Adjustment Multiplier	
Microsoft Windows	1
Palm OS	1
Telephone-based	1.25

Table 17. Operating System Adjustment Multiplier.

To simplify the calculation, the experts suggested using the sizing worksheet for each operating system separately. In the above example, the unadjusted server syspoint would be calculated twice – once for 1 low complexity server and adjusted to the Microsoft Windows operating system, and again for 1 low complexity server adjusted to the UNIX operating system.

Operating system adjustment multipliers are provided in the table below. These weights are multipliers and are identified based on Microsoft Windows operating system as the baseline. In other words, the unadjusted project syspoint will not change if the project only uses Microsoft Windows as the operating system in the project. The multipliers are determined by the experts based on their past experience.

The adjusted syspoint for the server ($\lambda_{\mathrm{S(adj)}}$) is calculated by multiplying the unadjusted server syspoint (λ_{S}) with the adjustment multiplier. The adjusted syspoint for the workstation ($\lambda_{\mathrm{W(adj)}}$) is calculated by multiplying the unadjusted workstation syspoint (λ_{W}) with the adjustment multiplier. The adjusted syspoint for the handheld ($\lambda_{\mathrm{Ha(adj)}}$) is calculated by multiplying the unadjusted handheld application syspoint (λ_{Ha}) with the adjustment multiplier. The adjusted syspoint for the project is calculated by adding the three adjusted syspoints to the other five unadjusted syspoints.

$$\begin{aligned}\Lambda_{\mathrm{PROJ(ADJ)}} = {}& \lambda_{\mathrm{S(adj)}} + \lambda_{\mathrm{W(adj)}} + \lambda_{\mathrm{P}} + \lambda_{\mathrm{LAN}} \\ & + \lambda_{\mathrm{WAN}} + \lambda_{\mathrm{Sa}} + \lambda_{\mathrm{Ca}} + \lambda_{\mathrm{Ha(adj)}}\end{aligned}$$

Where $\Lambda_{\mathrm{PROJ(ADJ)}}$ is the project adjusted syspoint, $\lambda_{\mathrm{S(adj)}}$ is the total adjusted server syspoint, $\lambda_{\mathrm{W(adj)}}$ is the total adjusted workstation syspoint, λ_{P} is the total unadjusted printer syspoint, λ_{LAN} is the total unadjusted LAN syspoint, λ_{WAN} is the total unadjusted WAN syspoint, λ_{Sa} is the total unadjusted server application syspoint, λ_{Ca} is the total unadjusted client application syspoint, and $\lambda_{\mathrm{Ha(adj)}}$ is the total adjusted handheld syspoint.

4.12. Limitations of Study

The non existence of a sizing model for IT infrastructure project forced the usage of a qualitative brain storming method like a focus group to develop a model. The method of selection of the focus group experts is not based on statistical sampling technique. However, this method is valid when the objective of the study is not to perform any statistical tests. This method is described as theoretical sampling until saturation (Glaser and Strauss, 1967). The study stopped with developing the model. The testing of the model will be conducted by another study which will be quantitative in nature.

This study focused on the IT infrastructure project-related sizing factors. The generic project factors were not included into the scope of the study. The generic project factors such as the expertise of resources, risk factors, critical nature of the project, and government project can influence the sizing of any project. Since this is the first study involved in developing an IT infrastructure sizing model, these generic factors were purposefully placed outside the scope of the study.

5. Conclusion

The focus group of experts discussed the advantages of having a sizing model and its application in their professional environment. In answering to the discussion question about the current sizing process, one of the experts commented, "...It allows me to relate to what is missing in our area. Even with the results we have identified so far, I am sure, I will be able to manage my projects better".

This comment allowed expanding the thought process of how the sizing model will allow the IT infrastructure project managers to manage their projects better. Hence a discussion question, "That is an important point that I would like to capture in detail. Why do you feel that you will be able to manage your projects better?" was raised. This discussion question had a lot of comments and they are discussed in detail below as they pertain to the application of syspoint in the project environment.

The first reply to this question was, "In the area of infrastructure projects, as you very well know, we don't have tools similar to that of software development projects. That does not mean that we don't carry out large projects. Without proper tools to do sizing, estimation, and comparison, we have to rely on our expert judgment. It helps to have an objective measure, document the procedure and allow the team members do the sizing, and estimation of projects using bottom-up technique. Most of my projects have been estimated just by me or with my manager. Top-down approach – even though it is faster, could mean lack of team members' buy-in. With a quantitative measure, as a manager, I will be able to make the team participate in this process, groom other members in estimation, improve team motivation etc. This will have a positive effect on the team members in various areas. This will make a difference for infrastructure project managers".

Another expert added, "Absolutely, this is an important research in the infrastructure area. Currently, grooming junior project managers in the infrastructure area is a difficult task. Since the key project management activities are performed using subjective approach, it becomes difficult to train junior project managers. Most training happens on-the-job. We have to learn through mistakes. If techniques and methods similar to software development area exist, we will be able to conduct a training class and teach them how to perform these activities instead of relying on them picking my brain".

In replying to this comment, another expert said, "Also, don't forget about the power of using an objective approach when trying to justify the estimates. Currently, it is subjective and requires a lot of justification on how we come up with the numbers we come up with. It is sometimes difficult to make the senior management understand expert judgment has some 'judgment' factor involved".

Based on this discussion, the following advantages of the syspoint measure were identified

- Objective estimation technique
- Consistency in estimation within the organization
- Ability to compare IT infrastructure projects within or between organizations
- Ease of training

- Ability to implement bottom-up estimating technique
- Ability to justify the estimates to the senior management
- Ability to calibrate based on the organizational experience
- Improve team buy-in and increase motivation among team members

The study offers a lot of scope for future research. Testing of this model is the first step. The next step would be to include the generic project sizing factors into the equation. Furthermore, extending this model to develop effort estimation and cost estimation can be performed using quantitative research techniques. Schedule estimation can also be performed based on a high level work break down structure for IT infrastructure projects.

Acknowledgment

The authors would like to thank the editors and reviewers of CIT for their positive feedback about the SYSPOINT research and for recognizing SYSPOINT as a valuable addition to the body of knowledge.

References

[1] A. ABRAN, M. MAYA & D. DESHARNAIS, Adapting function points to real-time software. *American Programmer*, November (1997), pp. 32–42.

[2] S. R. CHIDAMBER & C. F. KEMERER, A Metrics suite for object-oriented design. *IEEE Transactions on software engineering*, June (1994), pp. 476–493.

[3] J. B. DREGER, Function point analysis. *Englewood Cliffs*, NJ, Prentice Hall, 1989.

[4] D. FERENS, Software size estimation: Quo Vadis? *National Estimator*, Winter (1999), pp. 43–54.

[5] Function point counting practices manual, 2000, IFPUG

[6] B. G. GLASER & A. L. STRAUSS, *The Discovery of Grounded Theory*, Chicago, Aldine Publishing Co., 1967.

[7] J. JOHNSON, K. BOUCHER, K. CONNORS, K., & J. ROBINSON, Collaborating on Project Success, *Software Magazine*, Mar (2001).

[8] T. JONES, *Programming productivity*, New York, McGraw Hill, 1986.

[9] S. LIU, practical framework for discussing IT infrastructure, *IT Pro*, Aug (2002), pp. 14–21.

[10] K. PETERS, Software Project Estimation, Vancouver, CA, Software Productivity Center, 1999.

[11] Techweb 2003. The Business technology network: TechEncyclopedia (Online), Available, `http://www.techweb.com/encyclopedia/`

Received: August, 2003
Accepted: March, 2004

Contact address:
Srinivasa Raghavan
President
Krea Corporation
3471 Wilkes Drive
Naperville, IL-60564
USA
e-mail: sri.raghavan@kreacorp.com

Veeraswamy Achanta
President
Rangasoft Corporation
178 Michigan Court
Bloomingdale, IL-60108
USA
e-mail: swamy@swamy.com

SRINIVASA RAGHAVAN works as a President of Krea Corporation, an IT and management consulting firm located in Illinois, USA. He also functions as a mentor at the School of Business, NorthCentral University and as a faculty at the School of Technology, Capella University. He has a BS in Mathematics from the University of Madras, a MS in Computer Applications from the University of Madras, an MBA in International Business from Keller Graduate School of Management, a MS in Project Management from Keller Graduate School of Management and a PhD in Organization and Management with specialization in IT, from Capella University. Raghavan's research interests include IT project estimation models, decision support systems, operation research, project management, qualitative research techniques – in particular online methods, and IT life cycle models.

VEERASWAMY ACHANTA works as a President of Rangasoft Corporation, an IT consulting company located in Illinois. He has a B.Tech in Industrial Engineering from Nagarjuna University, Master's Degree in Engineering with Specialization in Industrial metallurgy from Bharathiar University, Master's Degree in Industrial Technology from Eastern Michigan University and an MBA in Finance from Keller Graduate School of Management. Veeraswamy's key interests include IT infrastructure and project management.