A COST-ORIENTED TOOL TO SUPPORT SERVER CONSOLIDATION

Danilo Ardagna, Chiara Francalanci
Dipartimento di Elettronica e Informazione, Politecnico di Milano, Via Ponzio 34/5, 10133, Milano, Italy

Gianfranco Bazzigaluppi, Mauro Gatti, Francesco Silveri
IBM Italy, Circonvallazione Idroscalo, 20090 Segrate, Milano, Italy

Marco Trubian
Dipartimento di Sicenze dell'Informazione, Università degli Studi di Milano, Via Comelico 39, 20135 Milano, Italy

Keywords: IT Costs, Distributed systems, Server Consolidation, Optimization

Abstract: Nowadays, Companies perceive the IT infrastructure as a commodity not delivering any competitive advantage and usually, as the first candidate for budget squeezing and costs reductions. Server consolidation is a broad term which encompasses all the projects put in place in order to rationalize the IT infrastructure and reduce operating costs. This paper presents a design methodology and a software tool to support Server Consolidation projects. The aim is to identify a minimum cost solution which satisfies user requirements. The tool has been tested by considering four real test cases, taken from different geographical areas and encompassing multiple application types. Preliminary results from the empirical verification indicate that the tool identifies a realistic solution to be refined by technology experts, which reduces consolidation projects costs, time and efforts.

1 INTRODUCTION

There is increasingly clear evidence that IT contribution to productivity growth is sizeable and positively affects both firms and countries (Dedrick 2003). Yet, the benefits of IT investments exhibit a striking variance among firms and the continual growth of the portion of capital spending devoted to IT is getting to unsustainable levels. There is therefore an increasing focus on curbing IT costs, notably on those areas not delivering visible and short term business benefits. The IT infrastructure, being often perceived as a commodity not delivering any competitive advantage (Carr 2003), is thus the first candidate for budget squeezing; with ultimate goals as heterogeneous as cost savings or freeing resources for more business-related IT investments.
Server consolidation is a broad, weakly defined term which encompasses all the projects put in place in order to rationalize the IT infrastructure. Although each project is unique, we can identify in each server consolidation project five main phases: a) goals and constraints identification; b) data gathering; c) analysis; d) solution test; and e) deployment.
For most of the projects the key goal is to reduce costs without adversely affecting the key operational requirements, i.e. performance, scalability and dependability. Thus the architectural design has to be optimized against the cost variable, while the operational requirements act as constraints.
The data gathering phase aims at collecting basic information about the to-be–consolidated servers (e.g. server configuration, OS, major applications) and workload data (e.g. peak CPU utilization). Since no non-intrusive tool existed in the market able to gather workload data, IBM has developed its own tool (IBM CDAT). All the customer data analyzed in this article were obtained using this tool.
During the analysis phase the data gathered during the previous phase are sorted out and architectural decisions are taken. The project team needs at least to: a) to select the consolidation technique; b) select the server models; c) size the systems; d) assign each application to a server e) determine the optimal

Ardagna D., Francalanci C., Bazzigaluppi G., Gatti M., Silveri F. and Trubian M. (2005).
A COST-ORIENTED TOOL TO SUPPORT SERVER CONSOLIDATION.
In *Proceedings of the Seventh International Conference on Enterprise Information Systems - DISI*, pages 323-330

location (if appropriate). This article is focused on identifying an optimal solution to problems (b), (d) and (e), where optimal means the cheapest solution meeting operational requirements.
The three key operational requirements are performance, scalability and dependability. As to performance, the aforementioned IBM CDAT tool measures the average and peak CPU utilization; it is therefore possible to select the server models and configuration in such a way that performance requirements are met. As to scalability, the model will be extended to cover scalability issues in future work. As to dependability, two High Availability (HA) cluster configurations have been considered (see Marcus 2000): a) 1-to-1 (symmetric), b) load sharing. 1-to-1 HA clusters consist of two nodes that deliver different services (even when the two nodes host the same type of application the two servers deliver different services). 1-to-1 clusters can be configured in asymmetric mode (also known as Active-Passive) or symmetric mode (also known as Active-Active). In the asymmetric configuration server applications run on the two servers but only one machine delivers service to users while the second one is in standby. In the Active-Active configuration, vice versa, server applications are installed on the two machines but only one instance of a server application is executed on the cluster. The asymmetric configuration makes a suboptimal usage of the resource and therefore it is implemented only when the symmetric configuration is not supported. Load sharing HA clusters consist of two or more nodes that deliver the same service. The multiple nodes extensions of 1-to-1 HA cluster (e.g. N-to-1 in which multiple nodes can fail over one standby node), albeit considered in the analysis, are not widespread enough in the Intel-based servers market and therefore have not been described in this article.
This paper is the result of a joint project between IBM and Politecnico di Milano. In a previous work (see (Ardagna and Francalanci 2002), (Ardagna et al. 2004) and references therein), we have developed a cost oriented methodology and a software tool, ISIDE (Information System Integrated Design Environment) for the design of the IT architecture. In this paper, we apply our tool to four server consolidation projects implemented by IBM for their customers, in order to evaluate the quality of our solutions. The results we obtained show that ISIDE can identify a low cost candidate solution, which can be refined by the project team experts, which reduces the cost and time of server consolidation projects. The current version of the tool does not consider scalability issues. However, the tool can be extended in order to entirely support the server consolidation process considering additional constraints.
This paper is organized as follows. The next section reviews previous approaches provided by the literature. Section 3 discusses a model for an Enterprise-wide Information System which supports a server consolidation project. Section 4 describes the current version of ISIDE which has been adopted to investigate case studies discussed in Section 5. Conclusions are drawn in Section 6.

2 RELATED WORK

A server consolidation project is a special case of design of an IT infrastructure. Modern infrastructures are comprised of hardware and network components (Menascé and Almeida 2000). Since hardware and network components cooperatively interact with each other, the design of the IT infrastructure is a systemic problem. The main systemic objective of infrastructural design is the minimization of the costs required to satisfy the computing and communication requirements of a given group of users (Jain 1987; Blyler and Ray 1998). In most cases, multiple combinations of infrastructural components can satisfy requirements and, accordingly, overall performance requirements can be differently translated into processing and communication capabilities of individual components. These degrees of freedom generate two infrastructural design steps: a) the selection of a combination of hardware and network components; b) their individual sizing.
Cost-performance analyses are executed at both steps. Performance analyses receive a pre-defined combination of components as input and initially focus on the application of mathematical models to define the configuration of each component (Lazowska et al. 1984; Menascé and Almeida 2000). Conversely cost analyses start at a system level, to identify a combination of components that minimizes overall costs, which is initially calculated from rough estimates of individual components' configurations and corresponding costs (Blyler and Ray 1998; Zachman 1999). The evaluation of costs of individual components is subsequently refined based on more precise sizing information from performance analyses.
The literature provides various approaches to support the design process, especially in the performance evaluation field (Menascé and Gomaa 2000) or for specialized applications (Gillman et al. 2000) and often only a limited set of architectural variables or sub-problems are considered (e.g. the

configuration of an application server (Xi et al. 2004)).

A scientific approach has been rarely applied to cost minimization and a rigorous methodological support to cost issues of infrastructural design is still lacking. Our model draws from (Jain 1987) the approach to the representation of infrastructural design alternatives as a single cost-minimization problem; however design variables and steps have been significantly extended to account for the complexity of modern computer systems and to support server consolidation projects. The cost optimization problem is NP-hard and in previous works (Ardagna et al. 2004) we have proposed an heuristic solution based on the tabu search algorithm which will be briefly discussed in Section 4.

3 THE SYSTEM MODEL

The model of the Enterprise Information System we consider is depicted in Figure 1. The Internet ties multiple local networks, whose number and extension depends on the topology of organizational sites and on the location of users. The Infrastructure is associated with a “Total Cost of Ownership” (TCO), defined as the summation of network costs, investment and management costs of all infrastructural components (Faye Borthick and Roth 1994). The server consolidation problem consists of the joint problem of selecting new hardware components and localizing them while minimizing the TCO of the system, according to operational requirements. The set of servers to be consolidated is a subset of application servers adopted in the current enterprise infrastructure which execute server applications (web servers, e-mail servers, DBMSs, etc.) and thin/Hybrid Fat Client (HFC) servers, i.e. servers which support users which adopt thin or HFC client computers (Ardagna et al. 2004) and access a Metaframe or Terminal Services environment (Microsoft 2003).

The system is described by the following fundamental variables:

- Organization sites S_i, defined as sets of organizational resources (users, premises and technologies) connected by a LAN.
- User classes C_i, defined as a group of users with a common application profile located in an organization site, where an application profile is characterized by the set of applications and functionalities, computing requirements, and user think time.

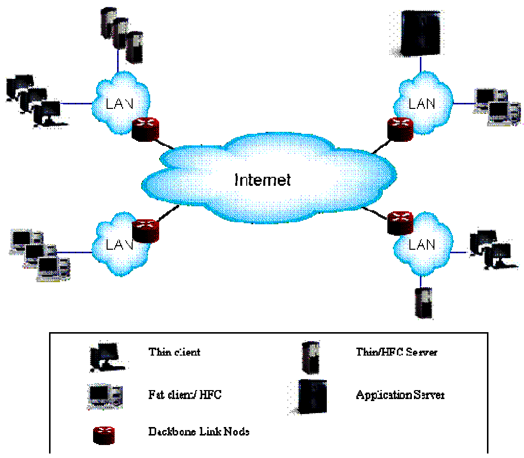

Figure 1: Enterprise Information System Model.

- Applications A_i, defined as a set of functionalities that can be accessed by activating a single computing process. Applications are characterized by computing and memory (primary and secondary) requirements.
- Databases D_i, defined as separate sets of data that can be independently stored, accessed and managed. Note that DBMSs are supposed to be specified as server applications and, accordingly, databases are simply described by the size of secondary memory that they require.

Applications and databases are the main drivers of architectural design of server farms, while the specification of sites and user classes is critical to select and size network components. All users in a user class are supposed to use the same set of applications. Application computing capacity requirements are evaluated by considering the average utilization of the server which support their execution in the original system. Note that a thin/HFC servers in a server consolidation project can be modelled as a single application. User classes and applications exchange data. The data exchange is modelled as a weighted directed graph whose nodes represent an application or a user class and the weight of each edge (i,j) represents the average bandwidth required to support data exchange between node i and j.

Different sites are constrained to be connected through an IP-based Virtual Private Network (VPN). VPNs have been selected due to their flexibility in realizing point-to-point connections. In this way, network design is performed by sizing link capacity and taking into account the associated costs. This provides a necessary input for the evaluation of overall infrastructural costs and allows the analysis of the impact of Internet costs on infrastructural design choices.

The optimization domain can be limited by the project team by specifying one or multiple constraints on the association between a) hardware components which will constitute the new consolidated system; b) existing applications. Constraints are defined by the project team before optimization and represent an input to the software tool presented in the next section. The set of all possible optimization constraints is defined as follows:

- Client Typology: each user class can be associated to a client typology (thin, fat or hybrid). The target operating system and, possibly, remote protocol are also specified.
- Allocation of applications and thin/HFC users: server application can be supported by the same cluster (i.e., a set of nodes which work collectively as a single system). This is specified by consolidation island Γ_k, i.e., sets of server applications A_i that can be possibly allocated to the same cluster. The set of consolidation islands is denoted as G_1.
- In the same way, thin and HFC servers can be shared among different user classes associated with thin/HFC clients and consolidation island Φ_k are defined accordingly. The set of consolidation islands for thin/HFC servers is denoted as G_2. Note that G_2 is a partition since usually user classes of an Information System are partitioned for security reason or privileges. If the cardinality of a consolidation island is n, 2^n-1 different allocations of server applications or user classes can be selected (that is, the consolidation island's power set, excluding the empty set). For example, if there are two instances A_1 and A_2 of web servers the consolidation island $\Gamma_1=\{A_1, A_2\}$ is introduced, then the set of candidate clusters is $\{A_1, A_2\}$, $\{A_1\}$ and $\{A_2\}$.

For each consolidation island Γ_k the following technology constraints are specified:

- The family of servers that will be adopted for the server consolidation (e.g., IBM xSeries 345 and xSeries 440).
- The virtualization of the servers in the consolidation island by means of a Virtual Machine Monitor (VMM), e.g. VMware ESX Server. VMMs allow multiple operating systems to run on the same server. Since most applications do not scale up, VMMs increase thereby server utilization.
- The value of availability *AV* required for the hardware platform.
- The fault tolerant schema implemented in the consolidated system (1-to-1 or load sharing).
- Thin/HFC servers and application servers, and corresponding user classes/server applications, can be constrained to be located in a specific organizational site.

As discussed above G_2 is a partition, while consolidation islands in G_1 can overlap. In this way multiple tiers allocation for server application can be defined. As an example, a servlet engine can be executed with a web server or an application server or as an independent tier, vice versa application and DBMS servers are usually allocated to individual machines (possibly supporting multiple application instances) for management and security reasons. Such a situation can be characterized by defining three consolidation islands: the first consolidation island contains web servers and servlet engines; the second consolidation island contains the servlet engines and the application servers and the third consolidation island contains all of DBMS applications.
Technology constraints are satisfied as follows:

Computing Capacity

The computing capacity of thin/HFC servers is evaluated as the maximum value of MIPS required by applications that are executed remotely multiplied by the number of concurrent users of the corresponding user class (Ardagna and Francalanci 2002). Servers are selected to provide a computing and storage capacity that guarantee a utilization of CPU and disk lower than 60% (Menascé and Almeida 2000; Ardagna et al. 2004) under the hypothesis of load balancing in the cluster and a single faulty server of the consolidated system. With values of utilization greater than 60%, small variations of throughput would cause a substantial growth of response time and, overall, performance would become unreliable. This empirical rule of thumb, which is commonly applied in practice (Menascé and Almeida 2000; Microsoft 2003), has been provided a formal validation. It has been formally demonstrated that a consolidation island of a-periodic tasks will always meet their deadlines as long as CPU and disk utilization of the bottleneck resource are lower than 58% (Abdelzaher et al. 2002). Note that performance analyses should follow cost analyses to refine sizing according to a formal queuing model. The aim of the proposed tool is to evaluate a large number of alternative solutions and find a candidate minimum-cost infrastructure that can be analyzed subsequently by the project team by applying fine-tuning performance evaluation techniques. The prediction of computing requirements for applications on the consolidated system is evaluated by benchmarking data (e.g. SpecInt, TPC-C). If a Virtual Machine Monitor is adopted, then corrective factors, which take into account the system overhead for the virtualization both for CPUs and disks, are considered.

Servers Availability

The availability required by a single server of the consolidated system depends on the fault tolerance schema required for each consolidation island. If the load sharing is implemented, then the availability AV_{Server} of each of the physical servers which compose a cluster of N machines is evaluated as follows:

$$AV_{Server} = 1 - \sqrt[N]{1 - AV}$$

this equation is derived by the relation that evaluates the availability of parallel systems (Trivedi 2002):

$$AV = 1 - (1 - AV_{Server})^N$$

The cluster which can support a given set of application is evaluated by an exhaustive search; anyway the enumeration can be stopped before reaching the maximum number of servers which can be supported by server applications (Ardagna et al. 2004).

Vice versa, if the 1-to-1 schema is selected, then a cluster of two machines is identified as a solution and the availability of each server is simply given by:

$$AV_{Server} = 1 - \sqrt{1 - AV}$$

Note that, in this latter case, the system has to be sized in order to guarantee that a single server can sustain the overall application load independent of the configuration Active-Active or Active-Passive adopted.

Network bandwidth requirements

Once physical hardware resources have been selected and servers have been localized to organization sites, the VPN is designed and sized. Each site is associated with its total input and output bandwidth requirements, calculated as the summation of input and output bandwidth requirements of all client/server requests exiting or entering the site and of thin clients and HFCs accessing remote servers. The bandwidth requirements of an active user of class C_i accessing a remote thin/HFC server are evaluated according to professional benchmarks as a function of think-time and of the remote protocol (Microsoft 2003). The capacity of the physical VPN, referred to as *physical bandwidth*, is then calculated according to Kleinrock's model:

Physical-Bandwidth=Total-Bandwidth+l/T,

where T is the time latency and l is the average packet size, which for IP-based VPNs can be empirically set to 200 ms and 550 bytes, respectively (Yuan and Strayer 2001).

4 SOFTWARE TOOL

The selection of a cost-minimizing combination of hardware and network components that satisfy organizational requirements is a complex design problem with multiple degrees of freedom. The corresponding optimization problem not only embeds the structure of NP-hard problems, but also represents a challenge for well-structured heuristic approaches. To reduce complexity we have applied a problem decomposition technique. We have identified three sub-problems which are sequentially solved. The given solution is improved by a final re-optimization step. The following decomposition is performed:

- *Thin/HFC servers optimization*: Thin clients and HFCs are assigned a minimum-cost clusters to support the remote execution of client applications. Clusters are assigned to their clients'site to minimize network communication costs. The solution of the thin/HFC server optimization sub-problem is modelled as a set partitioning problem (Papadimitriou and Steiglitz 1982) and is solved by a state of the art integer linear programming tool.
- *Server optimization*: Server applications are assigned to minimum-cost clusters that satisfy computing requirements and constrains. This optimization sub-problem is formalized as a set partitioning problem and is solved as the previous problem.
- *Server localization*: Server machines identified by solving the previous sub-problems are allocated to sites by minimizing overall network and management costs. This optimization sub-problem is formalized as an extension of a min k-cut problem (Lengauer 1990) and is solved by implementing a tabu search heuristic (Glover and Laguna 1997).

The problem decomposition is discussed in (Ardagna et al. 2004). The decomposition of the overall optimization problem into three sub-problems does not guarantee that the final solution is a global optimum. Hence, an overall re-optimization process based on a tabu-search approach has been implemented to improve the (possibly) local optimum obtained by separately solving the three sub-problems.

5 EMPIRICAL ANALYSES

This section provides empirical evidence of the quality of the solution which can be obtained by the software tool. Analyses focus on four case studies, which have substantially different requirements. In the first case study, a single site system is considered and the server consolidation includes only a limited number of servers of the customer infrastructure. The second test case considers a more complex IT infrastructure extended over three sites. The third server consolidation project is based on new technologies, i.e., server virtualization and blade servers; finally, fault tolerance issues are addressed in the last case study. The solutions provided by the software tool have been compared with those of the project team. The project team solution considers always peak CPU utilization load provided by IBM CDAT for the sizing of the hardware platform. Vice versa, the solution provided by the software tool considers two different scenarios. The first scenario considers the peak CPU utilization load, while the second scenario, in a more conservative way, assumes that the CPU utilization is 100%. This second solution will provide as a result an hardware infrastructure whose total computing capacity is greater or equal to that of the initial system configuration. Such solution can be used as benchmark even if it provides an over-sized estimate of system configuration and costs.

The TCO is evaluated by benchmarking data (Ardagna and Francalanci 2002, Ardagna et al. 2004) over a three year period. Management costs are estimated as a percentage of hardware investment costs as in (Blyler and Ray 1998). In the following tables TCO is expressed in Euros.

Server Consolidation Project A

The customer infrastructure includes 134 servers in a single site but the server consolidation project considers only 20 of them: 8 e-mail servers, 8 file servers and 4 print servers. The peak utilization of the set of servers target of the consolidation varies between 5% and 77%. The project team has considered two different alternatives:

- *Alternative a)*: file and print servers are consolidated on separate machines.
- *Alternative b)*: file and print servers are consolidated on the same set of machines.

From the methodological point of view, in the first alternative three different consolidation islands Γ_1, Γ_2 and Γ_3 are defined in G_1, one for each application type. Vice versa, in alternative *b*, only two islands are specified. Γ_1 includes file and print servers, Γ_2 includes e-mail application servers. The server families considered in the server consolidation project are xSeries 345 and xSeries 360. No availability constraints (in terms of availability required and fault tolerance schema) are introduced. Table 1 reports the solution provided by the IBM project team, for the two alternatives, Table 2 reports the solution identified by the software tool.

Table 1: Project A IBM Project Team Solution.

Altern.	**Initial numb. of servers**	**Numb. of servers of the cons. Solution**	**TCO**
A	20	6	160.544
B		4	166.112

Table 2: Project A Software Tool Solution.

Altern.		**Initial numb. of servers**	**Numb. of servers of the cons. Solution**	**TCO**
a	*peak CPU*	20	12	111.620
	100% CPU		20	170.042
b	*peak CPU*		12	106.422
	100% CPU		20	168.658

The software tool solution is cheaper in the peak utilization scenario of about 30%, while in the 100% scenario is more expensive than the IBM solution of about 5%. Both solutions are based on xSeries 345 and xSeries 360 servers. In the peak utilization scenario, overall the same computing capacity is implemented for both the software tool and the project team solution, but with different configurations. The software tool solution has lower costs, but the number of servers identified is twice as high as those of the IBM solution.

Server Consolidation Project B

The customer infrastructure comprises 62 servers in three remote sites. The server consolidation project considers 27 servers which are classified and located as reported in Table 3. The peak utilization of the set of servers target of the consolidation varies between 30% and 90%.

Table 3: Server applications in project B.

	Site		
Application	**S_1**	**S_2**	**S_3**
Application Server (A)	0	1	0
Application Server+DBMS (AD)	8	0	4
DBMS (DB)	0	2	0
E-mail (E)	1	4	2
Web (W)	0	4	1
Initial numb. of servers	**9**	**11**	**7**

The project team has identified three consolidation islands:

- Γ_1 includes applications of type A, AD and DB;
- Γ_2 includes e-mail servers;
- Γ_3 includes web servers.

The server families considered in the server consolidation project are xSeries 360 and xSeries 440. No availability constraints are introduced. Table 4 shows results in terms of total number of servers and costs for both the IBM project team, and the software tool solution.

Table 4: Project B Solutions.

Solution	Initial numb. Of servers	Numb. Of servers of the cons. Solution	TCO
IBM	27	5	149.148
Softw. Tool peak. CPU		4	61.512
Softw. Tool 100% CPU		6	97.480

Here, the difference between the peak and 100% utilization solution identified by the software tool are lower than in project A, since there is a lower variance in server CPU utilizations. Even the 100% CPU analysis has a lower cost than the solution provided by the IBM project team. The main reason for this difference of about 30%, origins in the scalability required by the customer which is not currently supported by the software tool. Indeed, the minimum cost solution identified by the software tool introduces only xSeries 360 servers, while the IBM solution introduces a few xSeries 440 servers which, with an additional cost of about 20-30%, provide higher scalability.

Server Consolidation Project C

The customer infrastructure includes 67 servers in two remote sites. The project addresses a large portion of the IT infrastructure since 56 servers, are considered for the consolidation. The peak servers' CPU utilization varies between 1% and 89%. The server families target for the server consolidation are xSeries 445 and BladeCenter. A single consolidation island is defined, which is supported by a virtual machine monitor and severs are constrained to be located in a single site. 30 applications with peak utilization lower than 5% are modelled as a single application (this reduce the number of alternatives explored by the software tool and it is reasonable to centralize under-utilized applications on a single server). No availability constraints are introduced. Results are reported in Table 5. A BladeCenter is considered as a single server independent of the number of blades installed. As results show, the software tool solution is 35% cheaper than the solution provided by the project team in the peak utilization scenario. Vice versa the 100% scenario is more expensive but the system is over-sized (30 severs in the original system have peak utilization lower than 5%). Both project team and software tool solutions employ a large number of blades. This is very attractive since the IT architecture is centralized, while it could be less interesting from a cost perspective in a multi-site scenario.

Table 5: Project C Solutions.

Solution	Initial numb. of servers	Numb. of servers of the cons. solution	TCO
IBM	56	5	245.972
Softw. Tool peak CPU		3	157.916
Softw. Tool 100% CPU		4	280.032

Server Consolidation Project D

The customer infrastructure comprises 32 servers in two remote sites and the server consolidation project considers the overall infrastructure. The peak utilization of the set of servers target of the consolidation varies between 2% and 90%. The project team has identified four consolidation islands:

- Γ_1 includes 8 DBMS servers, the target system is xSeries 440;
- Γ_2 includes 2 e-mail servers, the target system is xSeries 440;
- Γ_3 includes heterogeneous servers (mainly web and application servers), the target system is BladeCenter;
- Γ_4 includes heterogeneous servers (network and file servers), the target system is BladeCenter with a Virtual Machine Monitor (VMware ESX Server).

The first two consolidation islands support mission critical applications and the availability is fixed to 0.99999. The fault tolerance schema implemented is the load sharing. Results are reported in Table 6. Table 7 shows the solution identified by the software tool in the peak utilization scenario for consolidation islands Γ_1 and Γ_2 as a function of hardware availability.

Table 6: Project D Solutions.

Solution	Initial numb. of servers	Numb. of servers of the cons. Solution	TCO (€)
IBM	32	7	354.572
Softw. Tool peak CPU		6	276.566
Softw. Tool 100% CPU		5	429.202

It is interesting to note that, with current technologies, low availability requirements can be satisfied even by introducing a single server. The 100% utilization scenario is very expensive since the system is oversized, the cost difference between the peak utilization scenario and the solution identified by the project team is about 22%.

6 CONCLUSIONS AND FUTURE WORK

We have developed a software tool which supports server consolidation projects and identifies the IT infrastructure of possible minimum cost. The tool is based on the decomposition of the overall problem into sub-problems which are sequentially solved by optimization techniques and whose solution is fine-tuned by a local search based heuristic (tabu-search). The tool has been tested by considering four real projects implemented by IBM system designers. Results show that our minimum cost solution is realistic and the cost reduction with respect to an expert ranges in 20-30% of the total infrastructural cost. The minimum cost solution can support technology experts on further analysis and allows to reduce the time, costs and efforts required in server consolidation projects.

Availability constraints are included in the design of the IT infrastructure. The current version of the software tool identifies a possible minimum cost architecture, however it does not guarantee that the solution can be easily upgraded at low costs in order to support new customer requirements. Future work will consider scalability issues in the server consolidation process and analyses will be based on management costs provided by customers.

ACKNOWLEDGEMENTS

Particular thanks are expressed to Thomas Vezina (IBM US) who provided most of the data on which our analysis has been based. Thanks are expressed also to Mattia Poretti, Andrea Molteni and Simone Amati for their assistance in data collection and development activities.

REFERENCES

Abdelzaher, et al. 2002. *Performance Guarantees for Web Server End-Systems: A Control-Theoretical Approach*. IEEE Trans. on Parallel and Distr. Systems. 13(1), 80-96.

Ardagna, D., Francalanci, C. 2002. *A Cost-Oriented methodology for the design of Web based IT Architectures*. ACM SAC2002 Proc.

Ardagna, D. et al. 2004, *A Cost-Oriented Approach for Architectural Design*, ACM SAC2004 Proc.

Blyler, J. E., Ray, G. A. 1998. *What's size got to do with it? Understanding computer rightsizing*. IEEE Press, Understanding Science & Technology Series

Carr, 2003, *IT doesn't matter* by Nicholas G. Carr, Harvard Business Review

Dedrick, J., et al., 2003, *Information Technology and Empirical Evidence*, ACM Computing Surveys (CSUR)

Faye Borthick, A., Roth, H. P. 1994. Understanding Client/Server Computing. *Management Accounting*, 36-41.

Gillmann, M., et al. 2000. *Performance and Availability Assessment for the configuration of Distributed WFMS*. Proc. of the 7th International Conf. on Extending Database Tech.

Glover, F., W., Laguna, M. 1997. *Tabu Search*. Kluwer Ac. Publ.

Jain, H. K. 1987. *A comprehensive model for the design of distributed computer systems*. IEEE Trans. on software engineering. 13(10), 1092-1104

Lazowska, E. D., et al. 1984. *Quantitative System Performance Computer system analysis using queueing network models*. Prentice-Hall

Lengauer, T. 1990. *Combinatorial Algorithms for Integrated Circuit Layout*. J. Wiley & Sons

Marcus, E., et al., 2000, *Blueprints for High Availability. Designing Resilient Distributed Systems*, John Wiley & Sons

Menascé, D. A., Almeida, V. A. F. 2000. *Scaling for E-business. Technologies, models, performance and capacity planning*. Prentice-Hall

Menascé, D. A., Gomaa, H. 2000. *A method for design and performance modeling of client/server systems*. IEEE Trans. on software engineering. 26(11), 1066-1085

Microsoft. 2003. *Windows Server 2003 Terminal Server Capacity and Scaling*. www.microsoft.com/windowsserver2003/

Papadimitriou, C., Steiglitz K. 1982. *Combinatorial Optimization*. Prentice Hall

Trivedi, K. 2001. *Probability and Statistics with Reliability, Queuing, and Computer Science Appl.*, J.Wiley & Sons

Xi, B. et al. 2004. *Server performance and scalability: A smart hill-climbing algorithm for application server configuration*. WWW2004 Proc.

Yuan, R., Strayer, W. T. 2001. *Virtual Private Networks: Technologies and Solutions*. Addison Wesley

Zachman, J. A. 1999. *A framework for information system architecture*. IBM System Journal. 38(2), 454-470

Table 7: Solutions for DBMS and e-mail servers as a function of availability requirements.

		AV=0.9		AV=0.99		AV=0.999		AV=0.9999	
Cons. island	**Initial numb. of servers**	**Servers cons. solution**	**TCO (€)**	**Servers cons. solution**	**TCO (€)**	**Servers cons. solution**	**TCO (€)**	**Servers cons. solution**	**TCO (€)**
Γ_1	8	1	60.712	1	60.712	1	97.860	2	121.446
Γ_2	2	2	16.954	2	19.752	3	25.434	3	29.630