



**A QUANTITATIVE
AND ANALYTICAL
APPROACH TO SERVER
CONSOLIDATION**

JANUARY 2006

White Paper

**ANDREW HILLIER
CIRBA INC.**

COPYRIGHT © 2004-2006, CIRBA INC. ALL RIGHTS RESERVED

www.cirba.com

CONTENTS

Introduction	3
IT Asset Optimization Defined	3
Traditional Approaches to Asset Optimization	4
A Quantitative and Statistical Approach to Asset Optimization	6
Server Consolidation Analysis Overview	6
How Consolidation Analysis is Used	6
How Consolidation Analysis Works	7
Factors Affecting Rule Sets	11
Factors Affecting Workload Analysis	12
Other Technical Analysis Factors	13
Non-Technical Analysis Factors	14
Benefits of a Quantitative and Statistical Approach	14
Realization of Cost Reductions	15
Estimating Target Reductions	15
Validating Targets with Contracts Management	16
Determining True ROI	16
Conclusion	17

INTRODUCTION

Over the past 20 years the IT infrastructures of most organizations have moved away from a reliance on centralized computing power toward distributed systems. While the benefits of a distributed approach are numerous and well understood, the associated management challenges are yet to be mastered. A significant part of the challenge is the sprawl that can occur over time as applications and servers proliferate. Decentralized control and decision making around capacity, provisioning of new applications and hardware combined with a perceived low cost of server hardware have created environments with far more processing capacity than is required. When cost is considered on a server by server basis this may not be troubling, but when you consider the multiples in a large environment, having too many servers becomes a significant burden. Simple math suggests that on license redundancy alone, taking even a modest number of servers out of an environment saves a significant amount on a yearly basis.

This dynamic has caused many organizations to begin asking the question:

“How do we consolidate some of this capacity to drive out cost?”

The heterogeneous nature of distributed configurations makes this question extremely difficult and time consuming to answer.

This paper looks at the challenge of IT asset optimization, specifically server consolidation, and how an approach that relies on quantitative and statistical analysis can greatly assist in meeting the challenge.

IT ASSET OPTIMIZATION DEFINED

True asset optimization is multi-faceted. When considering the optimization of assets you must include four sub activities:

RATIONALIZATION

Rationalization is the systematic process of removing “slack” from the system and re-negotiating contracts and expenditures to match actual resource needs. This is especially important in areas such as software licensing; where being over-licensed has adverse budgetary impact and being under-licensed has adverse governance implications.

OPTIMIZATION

The complement to rationalization, optimization, is the process of altering actual resource requirements to allow gains to be realized in efficiency and economies of scale. It is typically used to free up resources so they can be utilized for other

purposes, and can be exploited in initiatives such as consolidation and virtualization.

CONSOLIDATION

Consolidation refers to the process of combining applications and/or data on systems in order to achieve economies of scale and increase aggregate resource utilization. Variants on consolidation include physical consolidation (merging similar / identical platforms into larger ones) and application consolidation ("stacking" multiple applications, or multiple instances of the same application, on a single server).

VIRTUALIZATION

Virtualization is the process of combining several OS images into a single virtualized platform, providing economies of scale in resource utilization while maintaining a partition between operational environments. This is often used as a mechanism to perform consolidation, but also complicates the process of rationalization, and is therefore best performed in a planned and controlled manner.

TRADITIONAL APPROACHES TO ASSET OPTIMIZATION

Asset sprawl and the potential of consolidating are an increasingly popular topic in IT and finance circles. There are significant savings to be had through successfully culling surplus licenses, maintenance and hardware from infrastructure. The issue is that identifying the low hanging fruit is very difficult as the number of parameters and variables that come into play when considering something like server consolidation is massive.

INFORMATION REQUIRED IN A SERVER CONSOLIDATION INITIATIVE

Server consolidation is a complex undertaking that requires a detailed knowledge of the static and dynamic attributes of an environment. In order to accurately identify consolidation candidates and assure that any planned transitions will be problem-free, a significant amount of information on the target systems is required. Information required in a typical consolidation analysis includes:

- Hardware Inventory & Configuration
 - System models
 - CPU architectures
 - Non-volatile (EEPROM) settings
 - Device settings
 - Serial numbers
- Operating System Settings & Files
 - OS versions and rev levels
 - Kernel parameters & Registry settings
 - Name service parameters
 - Locale and Time zone settings
 - Scheduled job configurations
 - Library versions
 - Local user accounts
- Installed Patches/Hot fixes
 - Security patches
 - Infrastructure software patches
 - Patch application frequencies

- Application Inventory & Configuration
 - Application versions
 - Application configuration settings
 - Application usage
- Middleware Configuration
 - Middleware versions
 - JVM versions in use
 - Heap settings
 - Class Paths
- Database Configuration
 - Database versions
 - SGA parameters
 - Data dictionary
 - Formatting and locale settings
- System Capacity and Utilization
 - CPU utilization
 - Network I/O
 - Per-process statistics
 - Device and resource statistics
 - Platform benchmarks

Although daunting to some, this list is just a starting point – in order to perform a comprehensive analysis it is necessary to scrutinize system configurations and runtime behavior to a much deeper level. Subtleties within the configuration of systems and environments can have a drastic impact on the well-being of applications and business services; as in all aspects of server management, the devil is often in the detail.

Viewed in this way, it is easy to see why server consolidation must start with a detailed understanding of the environment, a fact that must not be overlooked when undertaking a consolidation initiative.

METHODS COMMONLY USED FOR SERVER CONSOLIDATION

In most organizations the methods employed to optimize assets are often ad-hoc and/or unstructured in nature. Even in cases where projects are well structured, the absence of sufficient data in the analysis process often means that the undertaking will ultimately involve a leap of faith with no guarantee of success.

Common approaches to server consolidation include:

Incremental: During the course of business a consolidation opportunity becomes obvious, a manual analysis is performed and decisions are made.

Departmental: Small numbers of systems are scrutinized in a specific area of an organization, often with objectives that are tactical in nature with little support in the form of information, tools or methodologies.

Monumental: A high-level initiative is identified and planned, but the lack of a clear methodology makes it overly manual in its execution. Such initiatives often have a highly variable return on investment and may experience creeping timeframes, mainly due to imprecise objectives, ad-hoc analysis and the involvement of many parties (including external consulting firms)

While each approach has its own merits, each also suffers from limitations due to lack of rigor, the lack of the information or the lack of a coherent approach to solving the problem.

A QUANTITATIVE AND STATISTICAL APPROACH TO ASSET OPTIMIZATION

A quantitative and statistical approach to asset optimization is a process that considers all of the critical variables discussed earlier and results in a data driven and accurate view of optimization potential. This approach contains four main steps:

1. **Discovery:** An in-depth understanding of what is currently deployed and how it is configured and operating in an environment.
2. **Configuration comparison:** An analysis of configuration synergies across systems in iterative one to one comparisons to identify opportunities to stack applications, rationalize configurations, optimize resources or virtualize OS images. Candidates are identified due to similarity and consideration for the cost and effort required to make systems similar.
3. **Workload analysis:** Detailed workload information to enable what-if workload stacking analysis on candidate systems. This process effectively determines if synergistic workload capacity and patterns are present and can be exploited to the benefit of the organization.
4. **Scoring:** The creation of a scorecard based on the combination of configuration and workload data.

SERVER CONSOLIDATION ANALYSIS OVERVIEW

HOW CONSOLIDATION ANALYSIS IS USED

A server consolidation analysis is typically used in one of three ways:

VALIDATION OF EXISTING PLANS

Data driven analysis helps validate plans by ensuring that the source and destination systems are compliant from a configuration perspective and have sufficient capacity from a workload perspective. In other words, it can benefit initiatives that are already underway by providing independent validation and verification of the intended changes.

DIRECTED ANALYSIS

Using this method for a specific project allows the targeting of specific platform types, business groups or physical locations, in order to uncover consolidation potential and identify viable strategies. In other words, it provides the analysis mechanism when overall goals and strategies have been defined.

OPPORTUNISTIC TARGET IDENTIFICATION

This approach, sometimes referred to as “scavenging”, looks across broad sets of systems or entire environments, letting the rulesets and algorithms do the work of identifying candidates in a purely empirical manner. This replaces human resource intensive-tasks with rule-based number crunching, and the resulting scorecards often uncover “regions of compatibility” in areas that may not have been previously considered.

As is implied, the strategy employed is somewhat dependent on the scope and overall lifecycle of a consolidation initiative. In practice, however, all three are typically leveraged in order to automate the various phases of the project and provide an ongoing “dashboard” of optimization potential.

SCI %	a433-key	a51-ginger	aiw4	aiwgrind	aix	h102-fly	h11-tea	hewlett	hphphry	rh-greenhat	expression	kilkenny	matrix	overlord	sf-sugar	sf-doc
a433-key	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95
a51-ginger	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95
aiw4	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95
aiwgrind	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95
aix	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95
h102-fly	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95
h11-tea	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95
hewlett	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95
hphphry	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95
rh-greenhat	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95
expression	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95
kilkenny	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95
matrix	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95
overlord	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95
sf-sugar	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95
sf-doc	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95
tungsten	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95	95

Example of opportunistic target identification that spans platform and application boundaries.

In this visual “System Compliance Index” (SCI) representation, higher numbers (in green and yellow) indicate high compatibility, while lower numbers (orange to red) indicate low compatibility.

This example shows high compatibility among AIX and HP/UX systems (in the top left and center areas, respectively), and the relatively low compatibility among Solaris systems (bottom right). The ruleset used for this analysis includes, among other things, checks for binary and library compatibility, explaining the high intra-platform and low inter-platform scores.

HOW CONSOLIDATION ANALYSIS WORKS

As discussed, Server Consolidation Analysis is a multi-step process that analyzes multiple areas of system configuration and operation, and combines these into a single composite scorecard. The two general areas of analysis are *Configuration Compliance* and *Workload Compatibility*.

CONFIGURATION COMPLIANCE ANALYSIS

Configuration Compliance Analysis begins by performing a deep N-to-N comparison of all systems in the scope of the analysis. This yields a comparison matrix that indicates the complete set of differences between each server and all of its counterparts. In such a raw form this is not yet in a state that where conclusions can be drawn, as it is common for servers to have many differences from one another, particularly if they perform different functions.

The next step involves the application of a compliance ruleset, or consolidation “cookbook”, in order to analyze the differences between servers and derive weighted scores of how compatible they are for a particular purpose, such as running applications or sharing data. This step produces a System Compliance Index matrix, or SCI matrix, that provides a visual representation of compatibilities between systems.

SCI %	a433-key.int.cirba.com	aix.int.cirba.com	aix53-32.int.cirba.com	aix53-64.int.cirba.com	aix64.int.cirba.com	coffey.int.cirba.com	sles8.int.cirba.com	intuition.int.cirba.com	messier.int.cirba.com	rh73.int.cirba.com	fairy.int.cirba.com	s8-doc.int.cirba.com	sundin.int.cirba.com
a433-key.int.cirba.com	100	92	66	77	89	0	0	0	0	0	16	12	17
aix.int.cirba.com	92	100	72	80	96	0	0	0	0	0	16	14	16
aix53-32.int.cirba.com	66	72	100	86	75	0	0	0	0	0	11	13	13
aix53-64.int.cirba.com	77	80	86	100	83	0	0	0	0	0	12	13	13
aix64.int.cirba.com	89	96	75	83	100	0	0	0	0	0	14	14	15
coffey.int.cirba.com	0	0	0	0	0	100	2	20	10	20	0	0	0
sles8.int.cirba.com	0	0	0	0	0	2	100	0	0	7	0	0	0
intuition.int.cirba.com	0	0	0	0	0	20	0	100	24	57	0	0	0
messier.int.cirba.com	0	0	0	0	0	10	0	30	100	56	0	0	0
rh73.int.cirba.com	0	0	0	0	0	20	7	57	56	100	0	0	0
fairy.int.cirba.com	16	14	11	12	14	0	0	0	0	0	100	50	46
s8-doc.int.cirba.com	16	14	12	13	14	0	0	0	0	0	41	100	46
sundin.int.cirba.com	17	16	13	13	15	0	0	0	0	0	41	51	100

Another important aspect of the configuration compliance analysis step is the modeling of remediation costs. Individual rules within a compliance ruleset may optionally contain a remediation cost estimate, and if that rule is not satisfied then the cost is factored in. This allows the analysis to provide more than just a representation of compatibility; it can also compute an aggregate remediation cost that represents what it would take to make the systems compliant.

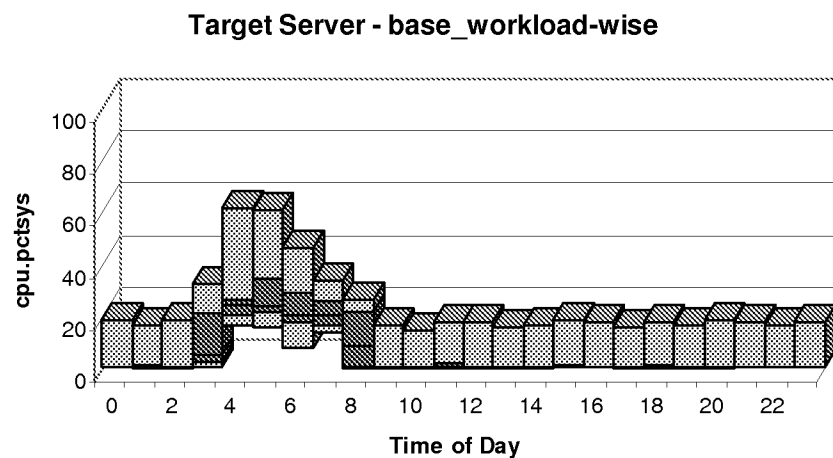
System Compatibility Index (SCI) Percentage		56%			
Estimated Remediation Cost		\$2,950			
Item	Instance	Baseline	Target	Weight	Remediation Cost
CIRBA Config Monitor Name Service Entry Value	hosts	files dir: nis	dir: nis: files	2%	
CIRBA Config Monitor Locale Settings Time Zone	-	US/Eastern	Canada/Eastern	10%	
CIRBA Config Monitor Kernel Parameters Table Entry	SEMMNI	absent	present	2%	\$90
CIRBA Config Monitor Kernel Parameters Table Entry	SEMMNS	absent	present	2%	\$90
CIRBA Config Monitor Kernel Parameters Table Entry	SEMMRU	absent	present	2%	\$90
CIRBA Config Monitor Kernel Parameters Table Entry	SEMNSL	absent	present	2%	\$90
CIRBA Config Monitor Kernel Parameters Table Entry	SEMOPM	absent	present	2%	\$90
CIRBA Config Monitor Kernel Parameters Table Entry	SENMIF	absent	present	2%	\$90
CIRBA Config Monitor Kernel Parameters Table Entry	SENMVK	absent	present	2%	\$90
CIRBA Config Monitor Kernel Parameters Table Entry	SEMAEM	absent	present	2%	\$90
CIRBA Config Monitor Kernel Parameters Table Entry Value	VER	Generic 117171-15	Generic 117235-11	5%	\$100
CIRBA Config Monitor Non-Volatile Settings (PIROMBOS) Entry	error-reset-recovery	absent	present	10%	
CIRBA Software Inventory Software Application Inventory Table Entry Version	oracle	-	8.1.7	5%	\$2,000
CIRBA Software Inventory Software Application Inventory Table Entry Version	apache	1.3.31 (Unix)	1.3.27 (Unix)	5%	\$190

This is useful for several types of analysis, including estimating upgrade costs and calculating the software overhead of virtualization, but in consolidation analysis it is used to represent the cost of “conditioning” a platform to allow application stacking.

WORKLOAD COMPATIBILITY ANALYSIS

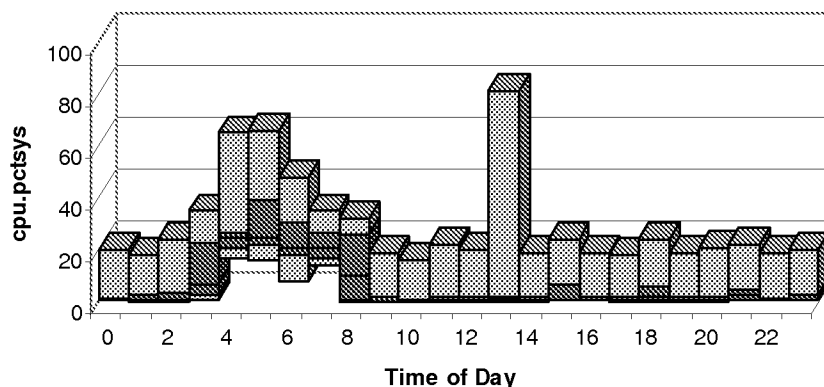
While configuration compliance analyzes the compatibility between systems from a static perspective, workload analysis looks at the dynamics of systems to understand if workload levels and patterns are compatible.

Workload analysis starts with the tracking of key operational statistics of each system in the scope of the analysis, and results in *historical histograms* of the operational patterns over time. Most analysis strategies make use of a few key stats, such as CPU utilization and network activity, but workload tracking facilities can track up to 100 distinct stats on a typical UNIX system, providing considerable breadth and strategy-specific detail to the analysis process.

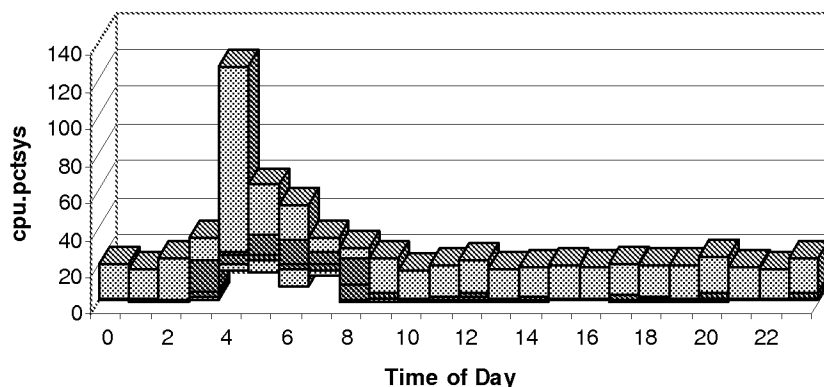


The resulting histograms can be then used to perform a *what-if analysis* by normalizing the histograms and stacking them on one another. Using a matrix correlation strategy similar to the configuration compliance analysis, the selected workload statistics can be stacked on one another for each server combination to identify which combinations will fit within the resources of the target system and which ones will not. This is a multi-step process that independently analyzes peak loads and third quartile (75%) loads, at both like times and worst-case times (simulating detrimental shifts in the processing pattern of the servers), in order to product a weighted score of each scenario.

Source Mapped to Target (Ratio 1)



Source mapped to Target (Ratio 1, -9 Hour Offset)



Workload Compatibility Index (WCI) Computation Scorecard:

Criteria	Value	Score
Peak 4th Quartile (Like Times)	81	100
Peak 4th Quartile (Worst Case)	127	78.74016
Peak 3rd Quartile (Like Times)	39	100
Peak 3rd Quartile (Worst Case)	40	100

Aggregate WCI **98**

The output of this step is a Workload Compatibility Index matrix, or *WCI matrix*, that highlights favorable consolidation candidates from a workload perspective. If multiple stats are being analyzed, then this step is repeated for each, and the resulting matrices are mathematically combined to produce a single composite WCI matrix.

WCI%	a433-key.int.cirba.com	aix.int.cirba.com	aix53-32.int.cirba.com	aix53-64.int.cirba.com	aix64.int.cirba.com	coffey.int.cirba.com	sles8.int.cirba.com	intuition.int.cirba.com	messier.int.cirba.com	rh73.int.cirba.com	fairy.int.cirba.com	s8-doc.int.cirba.com	sundin.int.cirba.com
a433-key.int.cirba.com	100	67	75	71	78	66	98	100	98	100	98	89	83
aix.int.cirba.com	67	100	70	69	68	0	95	99	94	98	91	69	70
aix53-32.int.cirba.com	75	70	100	72	74	70	100	100	96	100	96	75	74
aix53-64.int.cirba.com	71	69	72	100	71	69	99	100	96	100	92	71	71
aix64.int.cirba.com	78	68	74	71	100	68	99	99	98	100	97	70	70
coffey.int.cirba.com	66	0	70	69	68	100	95	99	94	95	91	69	69
sles8.int.cirba.com	98	69	75	71	78	69	99	99	99	98	97	100	89
intuition.int.cirba.com	98	91	96	93	97	91	100	100	100	100	100	97	97
messier.int.cirba.com	100	99	100	100	99	99	100	100	100	100	100	99	99
rh73.int.cirba.com	98	94	96	95	96	94	100	100	100	100	100	98	98
fairy.int.cirba.com	99	95	100	98	99	95	100	100	100	100	100	98	99
s8-doc.int.cirba.com	100	98	100	100	100	98	100	100	100	100	100	99	100
sundin.int.cirba.com	83	70	74	71	70	69	99	99	98	100	97	89	100

GENERATING A COMPOSITE SCORECARD

After the completion of the configuration compliance and workload compatibility analyses, the two can be overlaid and mathematically combined to produce a Co-Habitation Index matrix, or *CHI Matrix*. This is the ultimate scorecard in the analysis process, and provides a visual representation of consolidation candidates.

All other analysis outputs are indexed to this matrix, allowing a promising opportunity to be investigated by looking into the workload scores, the individual workload patterns, the compliance scores, the detailed differences between servers, and the remediation costs.

FACTORS AFFECTING RULE SETS

A key element of the configuration compliance analysis is the rule set, or cookbook, that is being used. It must reflect the nature of the opportunity being investigated, and different rulesets exist for different purposes:

HEAVYWEIGHT / BINARY APPLICATIONS

The analysis of heavyweight applications is mainly concerned with binary compatibility and the global settings on the system that affect the application. For example, the following factors are typically scrutinized when determining if two such applications can co-exist on a single OS image:

- OS version
- Maintenance and patch levels
- kernel settings
- name service settings
- locale and time zone settings

- infrastructure software versions
- library versions

A typical cookbook for this purpose will contain dozens of distinct rules, and may also include site-specific or application-specific conditions that must be met, such as the existence of a specific user account, in order to produce a high score.

JAVA / .NET APPLICATIONS

Because applications that run within virtual machines are somewhat isolated from the OS environment, the rulesets typically focus on the environmental components affecting their operation, and not necessarily the platform compatibility itself. Examples of configuration areas that are scrutinized in this case include:

- JVM versions
- class paths
- name service settings
- registry settings
- file systems and shares
- queue manager settings
- web archives

As in other scenarios, rulesets may be augmented with specific conditions that must be met, such as the presence of specific Java/XML resources.

DATA / CONTENT CONSOLIDATION

The consolidation of data content, such as is the focus of database consolidation initiatives, focuses on the compatibility of the data management infrastructure. Examples of system properties scrutinized in this type of analysis include:

- RDBMS manufacturer and version
- System Global Area (SGA) settings
- Data Dictionary settings
- Date formatting settings
- Server Time zone settings

Again, many other factors can be considered, including security patch levels and other data integrity-related requirements, when performing this type of analysis.

FACTORS AFFECTING WORKLOAD ANALYSIS

As with configuration compliance analysis, workload analysis can focus on specific aspects of system operation in order to portray an accurate picture of the post-consolidation world.

WHICH STATISTICS TO ANALYZE

Almost every workload analysis begins with the *CPU utilization*, and the most common variant to use is the overall usage (i.e. the inverted idle time, rather than system or user time) as it is typically reflective of the overall impact of an application on the system and its compute resources.

Applications that communicate over networks (which means virtually all applications) will benefit from the analysis of *network utilization*, as stacking such applications will incur an aggregate load on the network interfaces that should be understood.

Other statistics that can be factored into the analysis include:

- per-process CPU utilization
- run queues
- file system I/O
- swap space and commit charge
- inter-process communication activity (e.g. semaphores)

USING BENCHMARKS TO NORMALIZE WORKLOADS

When performing the what-if analysis, systems are first normalized to reflect their relative power. This is critical, as it allows the modeling of scenarios where small servers are being consolidated onto larger ones, or where new servers are being introduced. For example, a heavy workload on a low end server may translate into a negligible workload on a powerful server, and vice versa. There are several mechanisms available for normalizing workloads, including referencing industry benchmarks as well as empirical benchmarking.

The use of industry benchmarks allows a consolidation scenario to be analyzed in a manner consistent with the use pattern of the application being consolidated. For example, applications that perform integer operations will be best represented if the analysis uses SPECint numbers to normalize the CPU workloads. Similarly, floating point, I/O and Java-specific benchmarks can be utilized depending on the scenario.

If purely data-driven analysis is desired, then empirical benchmarking can be used to represent the relative power of two systems. Workload tracking facilities can be used to provide the data for the analysis. The results of this can be used to estimate relative computational power without referencing industry benchmarks, allowing the analysis to be completely data-driven.

For even more accurate results, the benchmark can be replaced with an application-specific benchmark that more precisely mimics the operation of the target application, providing an extremely accurate answer.

OTHER TECHNICAL ANALYSIS FACTORS

There are a variety of other factors beyond configuration and workload that can greatly impact the compatibility of consolidation candidates. Consideration for the following should be included in any consolidation exercise:

CHANGE HISTORY COMPATIBILITY

If the sample history gathered during the consolidation analysis includes a sufficient span of time then the change histories of the target systems can be scrutinized. The main consideration with change histories is the change tolerance of given applications; if the target system is frequently changed in ways that may disrupt the operation of an application then this may be cause for concern. Patch activity is an example of change where the tolerances of two applications may not line up, and combining them onto a single OS image may be detrimental.

ACCESS CONTROL COMPATIBILITY

Although this is an area often covered by the configuration compatibility analysis, it warrants special mention. If an application is designed to allow user-level access to a server then it may be incompatible with another application that assumes its server is locked-down. In other words, if data access rules are designed in a non-shared environment then moving to a shared environment may compromise data integrity. In such cases the answer may be to go to virtualized partitions and not share a common OS image.

NON-TECHNICAL ANALYSIS FACTORS

There is more to server consolidation analysis than number crunching, and there are a number of non-technical considerations that must be addressed in most initiatives. These include:

BUSINESS GROUPS

Crossing business group boundaries as part of a consolidation initiative can be difficult, as business service boundaries are often not addressed through empirical analysis alone. To be truly effective this must be part of an overall infrastructure management strategy, and take chargeback models and other agreements into account in the analysis phase.

GEOGRAPHY AND TIME ZONES

Consolidation across time zones can be problematic, as applications may be sensitive to the global locale settings of the system and therefore may experience problems if they were to change. Given this, analysis should avoid stacking servers that are in different locations unless it is part of the overall strategy, and if it is part of the strategy then all location-specific aspects of system operation should be factored into the analysis.

SUCCESSFUL IMPLEMENTATION

It is often helpful to combine risky changes with features or improvements that are considered value-add to the end user. By bundling performance or functional improvements with the overall technology transition, an implicit buy-in can be fostered in the user community, thus mitigating the risks of operational issues by providing a perceived upside that justifies this risk.

BENEFITS OF A QUANTITATIVE AND STATISTICAL APPROACH

A methodical, analytical approach to Asset Optimization provides several key benefits:

LITTLE OR NO RESEARCH OVERHEAD

Empirical and data driven results are produced from actual system data, signatures and rulesets rather than manual evaluation and comparison.

ACCURATE RESULTS

Because results are data-driven they are not as prone to errors in judgment or data entry. Detailed mathematical analysis ensures that the problem is treated with the same degree of sophistication as the systems being analyzed.

BROAD, MULTI-DIMENSIONAL COVERAGE

The unique combination of configuration and workload analysis give a broader view of opportunities and provides a higher degree of validation for existing plans. Furthermore, the ability to use multiple workload statistics, weighted through a variety of industry and empirical benchmarks, adds further dimensions to the overall scoring mechanism.

FLEXIBLE RULESETS AND SCORING

Because there are several flavors of optimization and consolidation, and because applications, systems and environments must be analyzed within a broader ecosystem, the ability to employ multiple distinct rulesets allows meaningful analysis to be performed and multiple strategies to be simultaneously pursued.

REALIZATION OF COST REDUCTIONS

ESTIMATING TARGET REDUCTIONS

The detailed hardware, OS and software inventories that are generated as part of the data gathering process are indispensable in the planning stages of a consolidation project. Accurate reporting of “actuals” at the outset of an analysis helps set goals and identifies areas for directed analysis.

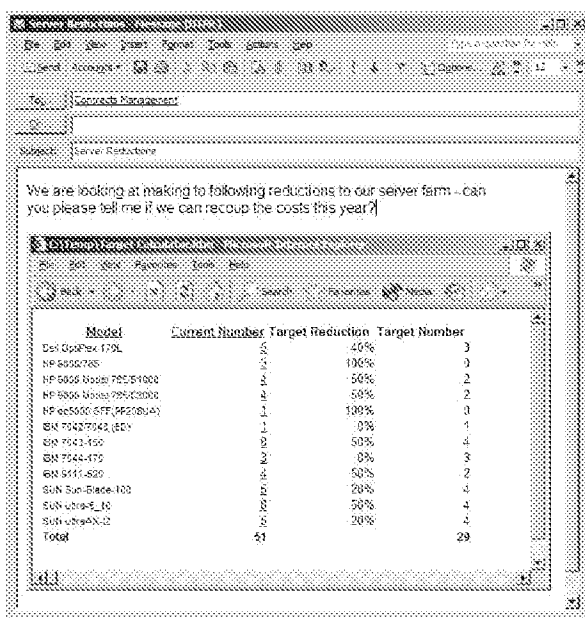
By using these inventories, a calculator can be used to plan overall strategies and estimate reduction levels:

Model	Current Number	Target Reduction	Target Number
Dell OptiPlex 170L	5	40%	3
HP 9000/785	3	100%	0
HP 9000 Model 785/B1000	4	50%	2
HP 9000 Model 785/C3000	4	50%	2
HP dc5000 SFF (PP208UA)	1	100%	0
IBM 7042/7043 (ED)	1	0%	1
IBM 7043-150	2	50%	1
IBM 7044-175	3	0%	3
IBM 8111-525	4	50%	2
SUN Sun-Blade-100	5	20%	4
SUN Ultra-5_10	20	50%	10
SUN Ultra-X-12	10	20%	8
Total	51		29

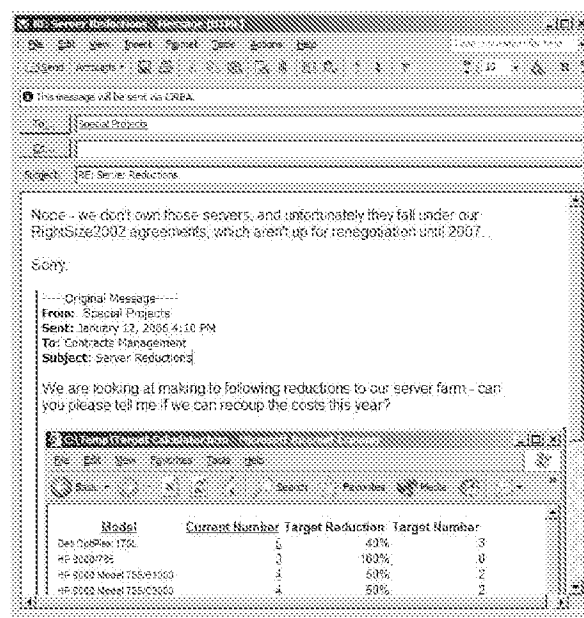
VALIDATING TARGETS WITH CONTRACTS MANAGEMENT

It is critical that any consolidation initiative target gains that are attainable. The act of consolidating servers is as much a political and contractual exercise as it is a technical one. To that end, it is critical that all affected parties be engaged, and that buy-in is sought in order to facilitate the project and mitigate risk.

One very important aspect of this process is the “contracts heads-up”, a process in which contracts groups are consulted regarding all planned asset reductions. Communicating with contracts groups early on allows them to verify whether the targeted gains can be realized from a contracts perspective, and long-term contracts and outsourcing agreements will often affect whether financial benefits can be realized.



You should send this early...



...so you don't receive this too late

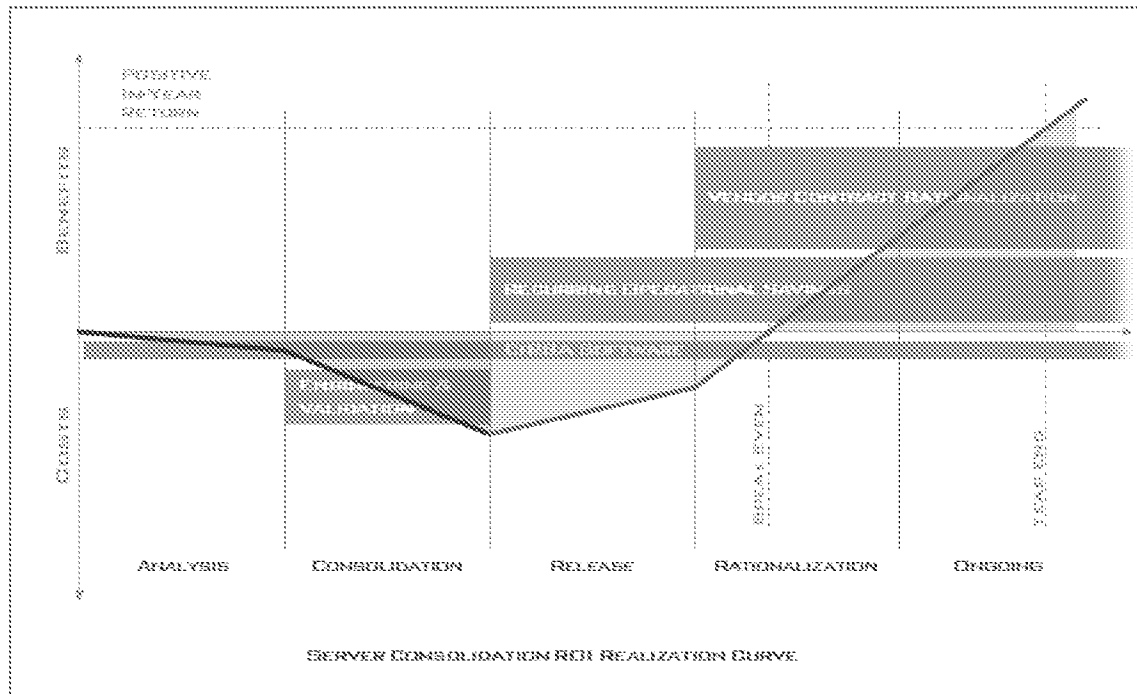
The other major benefit of the contracts heads-up is that it gives contracts management groups important information for future negotiations by communicating the architectural direction that the organization is taking. It may turn out that contractual options for more capacity are less important than options on less. It may also turn out that a contract can be renegotiated within the timeframe of a long consolidation initiative, thus allowing a strategy to proceed.

DETERMINING TRUE ROI

By basing a consolidation strategy on factual data and accurate analysis, the likelihood of achieving a substantial ROI is greatly increased. Also, by providing quality information to all affected groups early on in the process, strategies that are “red herrings” and are not realizable can be avoided. Both of these factors are critical to the successful planning and execution of server consolidation

projects. A common mistake when considering ROI is forgetting about the need to do some engineering and validation of the proposed consolidations. Depending on the nature of the applications that will be moved, this cost can vary considerably. The following chart illustrates how although you might incur an up front cost for analysis and validation, it isn't uncommon to see an "in year" return on the investment. A significant portion of the ROI is from the removal of the operational costs associated with keeping a server running. By far, the most significant contributor is the rationalization of vendor contracts around software that is no longer required.

CONSOLIDATION CUMULATIVE ROI



CONCLUSION

The motivations behind asset optimization and specifically server consolidation are usually very clear. Identifying excess capacity and license costs frees precious budget for initiatives that are more strategic and helpful to the business. It would be difficult to find a less strategic spend than unused hardware and unneeded software licenses. That said, spending significant effort and budget on consolidation efforts only chews into the ROI – good money after bad. The issue for many is the realization that it's the right thing to do, but paralysis with respect to how to get it done in a reasonably efficient and cost effective way. Many are turning to major hardware vendors for advice and even software that helps analyze consolidation potential. Turning to hardware vendors is a bit like letting a fox into the hen house; one has to wonder about motivations. A self guided quantitative and analytical approach makes it pure, removes the mystery and makes the exercise one that everyone can understand. Additionally, it optimizes the process so that not only is it "optimally inclusive" with respect to the hardware you can consider, it also significantly raises the likelihood of success and drives out risk.

There are a number of consolidation ROI calculators available that estimate the savings and payback period of server consolidation analysis. For more information on this, and details on how consolidation analysis can be evaluated in a specific environment, please contact CiRBA at the coordinates below.



1595 16th Avenue, Suite 400

Richmond Hill, Ontario

Canada, L4B 3N9

Telephone: +1.905.731.0090

Fax: +1.905.731.0092

<http://www.cirba.com>

Copyright © 2006, CIRBA Inc. All rights reserved.

CIRBA Server Consolidation White Paper – January 2006

Draft Claims

A method for analyzing computer system configurations and system resource utilization in order to determine inter-system compatibilities. Method consists of comparative analysis of configurations using rule-based weighting factors as well as combinatorial “what-if” analysis of normalized resource utilization to provide overall scoring and rankings. Such rankings are indicative of the propensity of software applications to migrate between systems or to be combined together on a single system, revealing optimization opportunities in areas such as server consolidation, application stacking and server virtualization.

We claim:

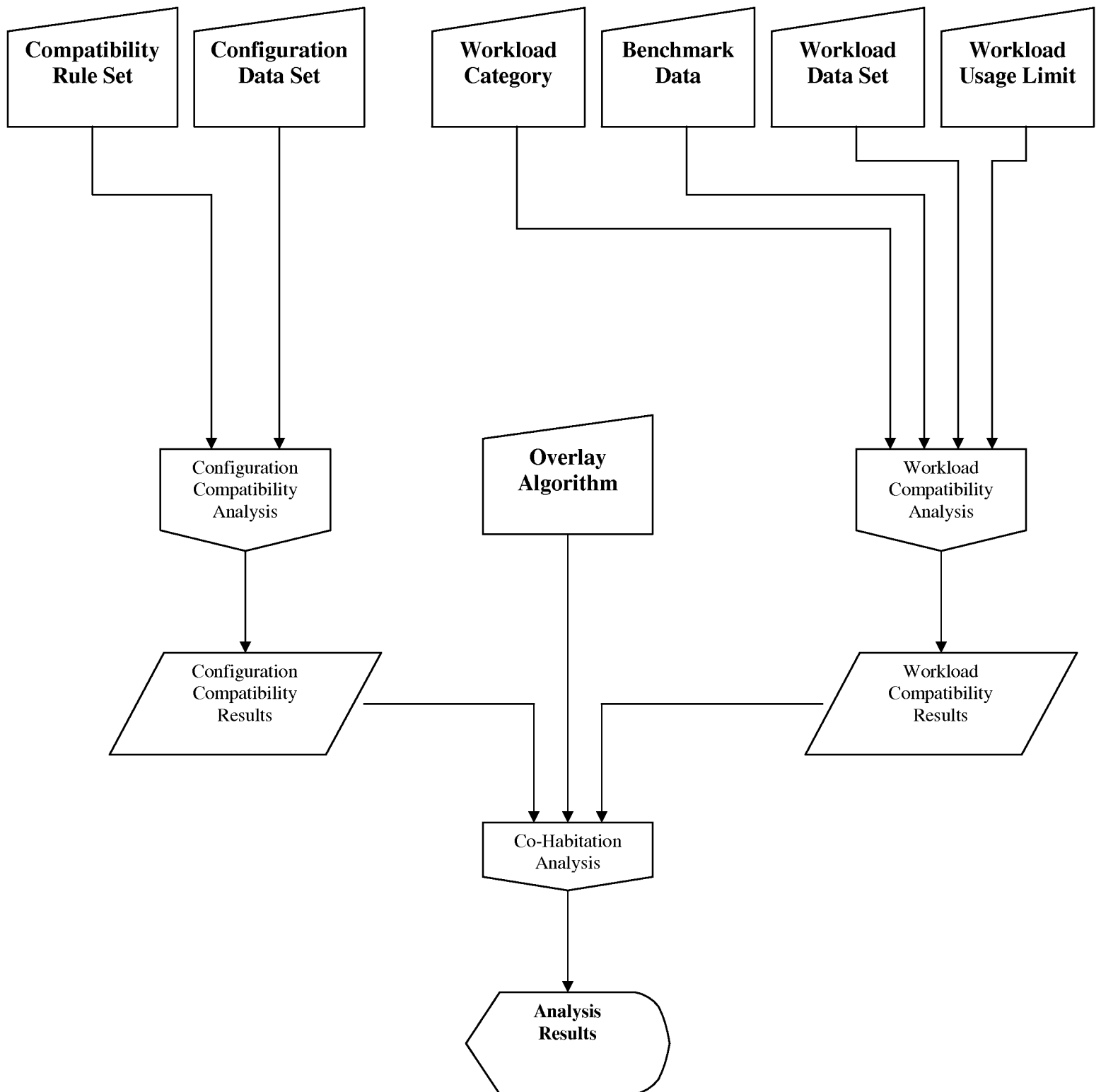
1. A method for analyzing computer system configurations and system resource utilization in order to determine inter-system compatibilities. Method consists of comparative analysis of configurations using rule-based weighting factors and combinatorial “what-if” analysis of normalized resource utilization to provide normalized scoring and rankings.
2. A method to render analysis results in matrix form
3. Method consists of cross-correlation of system hardware and software configurations, scoring of configuration correlation results, cross-correlation of system resource utilization (workloads) through normalization of metrics and additive “what-if” analysis, scoring of workload stacking results, and the combination of the configuration and workload scores into an overall co-habitation score for each system combination.
4. Software to perform said analysis and provide results in electronic form.

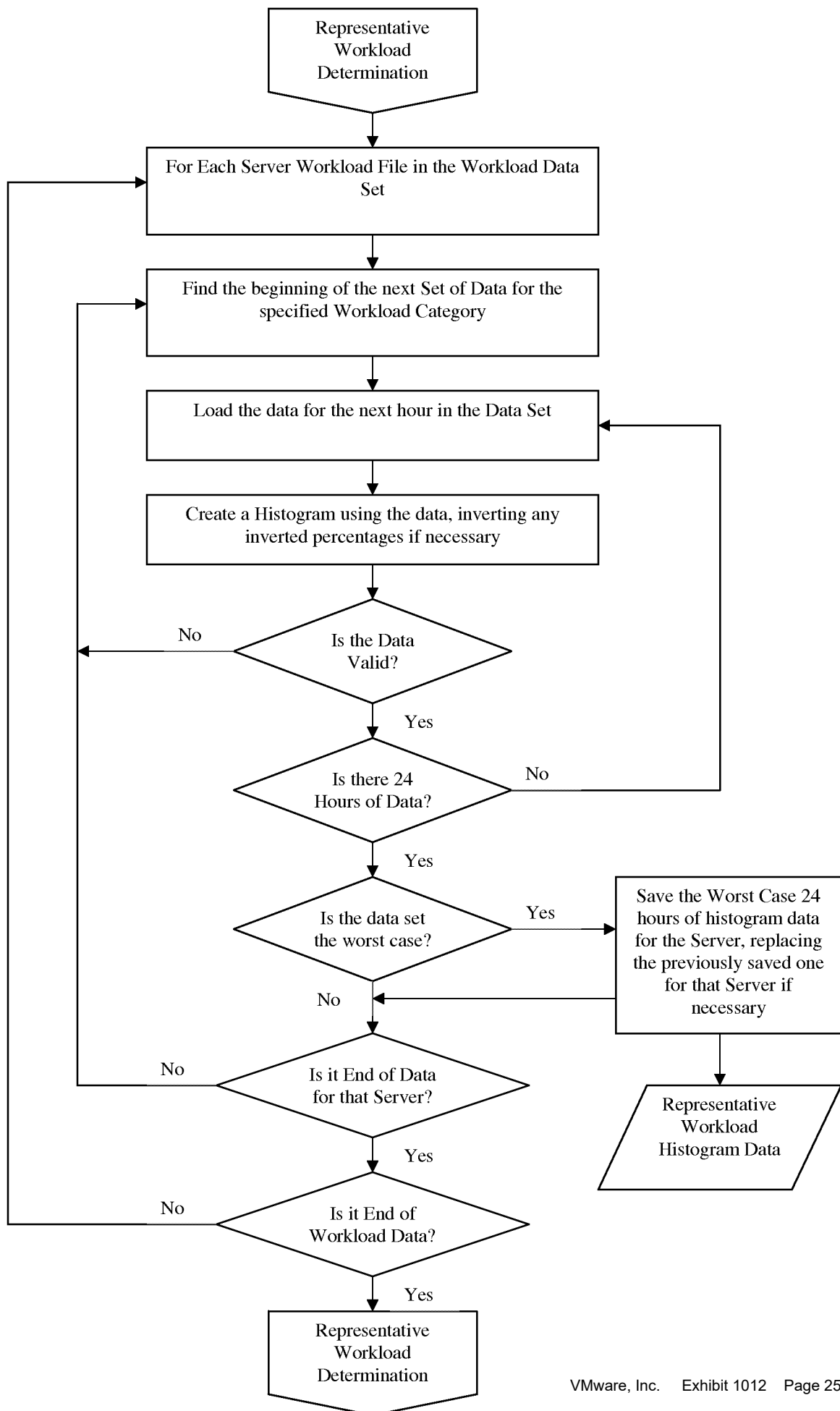
A method for analyzing computer system configurations and system resource utilization in order to determine inter-system compatibilities. Method consists of comparative analysis of hardware and software configurations using rule-based weighting factors as well as combinatorial “what-if” analysis of normalized resource utilization to provide overall scoring and rankings. Such rankings are indicative of the propensity of software applications to migrate between systems or to be combined together on a single system, revealing optimization opportunities in areas such as server consolidation, application stacking and server virtualization.

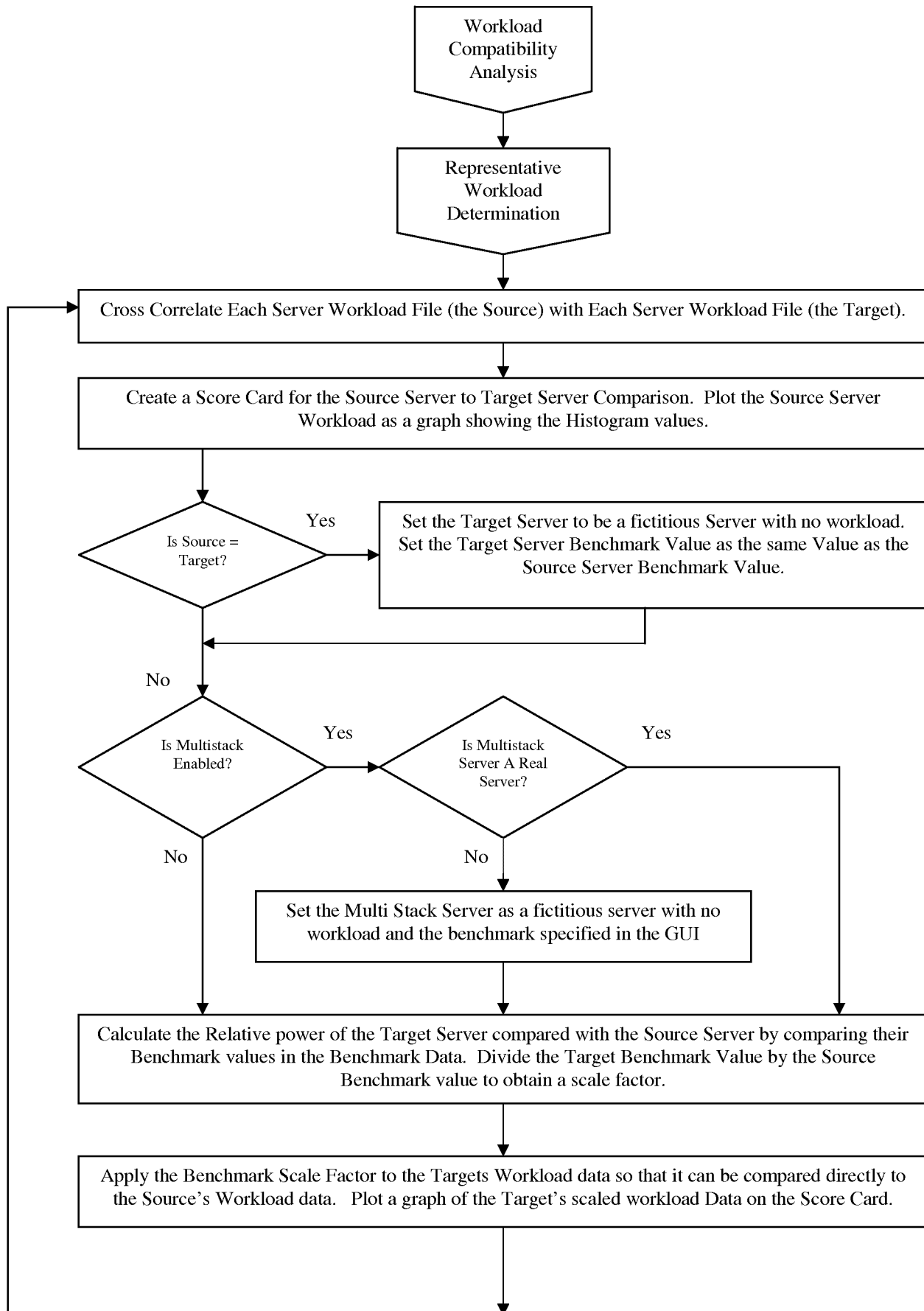
We claim:

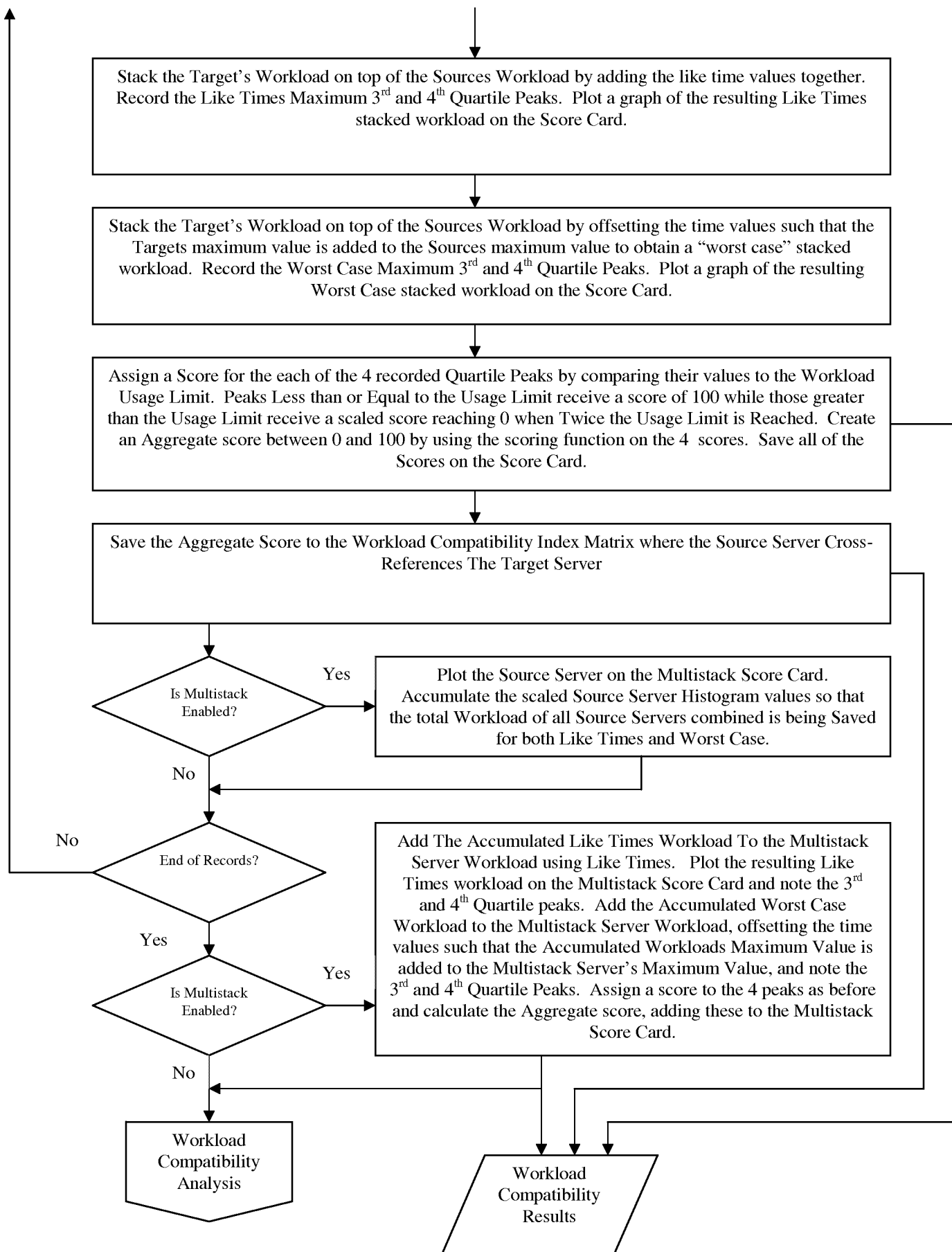
1. A method for analyzing system configurations in order to determine system compatibilities (SCI Analysis)
2. A method for analyzing system workload patterns in order to determine workloads compatibilities (complementary resource utilization) between systems (WCI Analysis)
3. A method for assigning scores to inter-system configuration compatibilities using rule sets that correlate weighting factors and remediation costs to key items that can potentially differ between systems
4. A method for assigning scores to inter-system workload combinations that are complementary in nature, with such method accounting for peak and typical usage in both well-behaved (like-times) and worst-case (time shifted) scenarios
5. A method for scoring N:1 workload stacking scenarios (“multi stack”) that allows multiple workloads to be assessed against an actual or theoretical server
6. A method for combining the configuration and workload scores to produce an overall co-habitation index for each combination of systems (CHI Computation)
7. A method for rendering the various scoring outputs in a colour-coded matrix form, allowing visualization of system compatibilities
8. A method for sorting scoring matrices that allows ranking of system by both propensity to move applications off as well as propensity to receive new applications
9. Software to perform the analysis, scoring and rendering steps described above

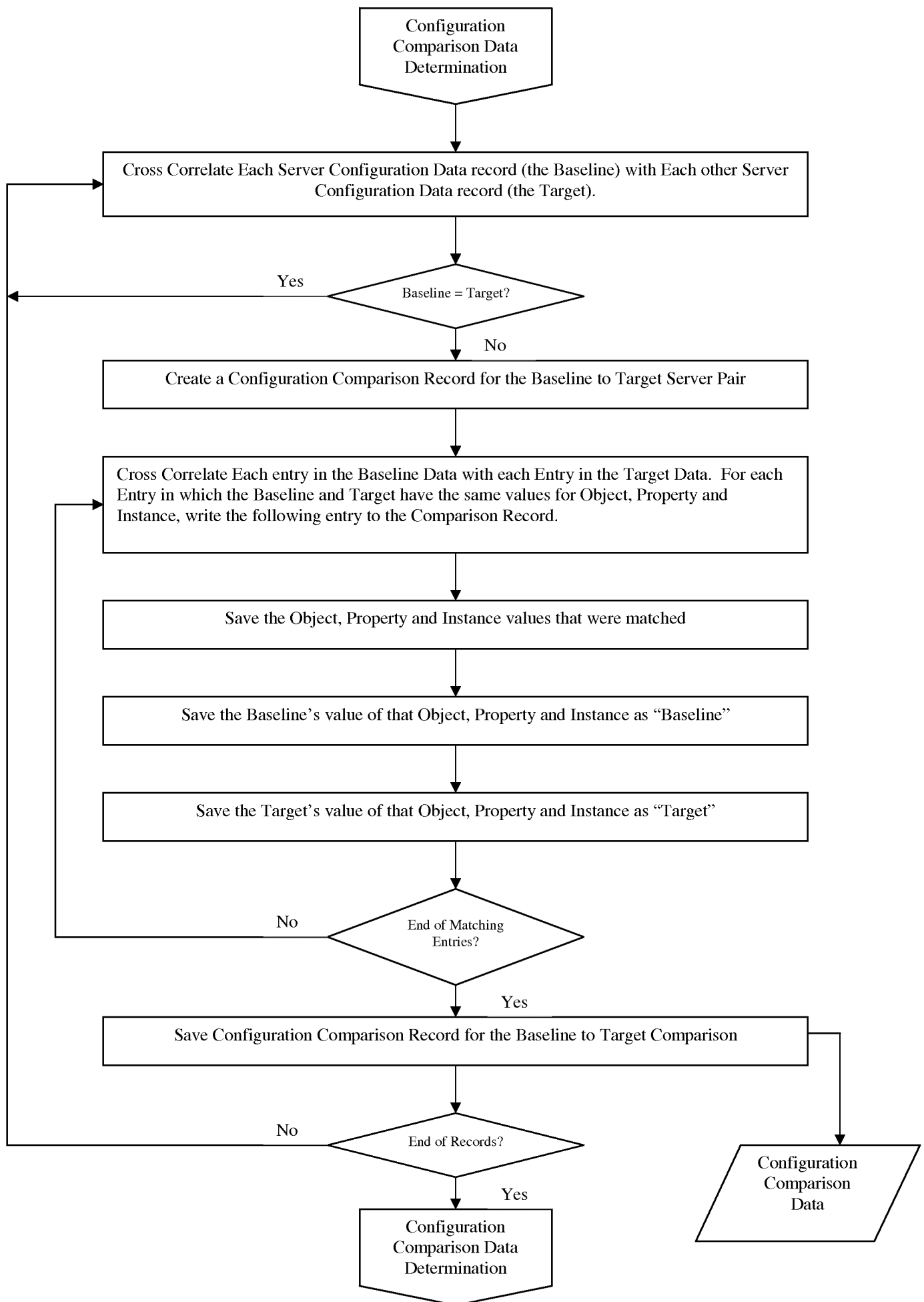
Flowcharts

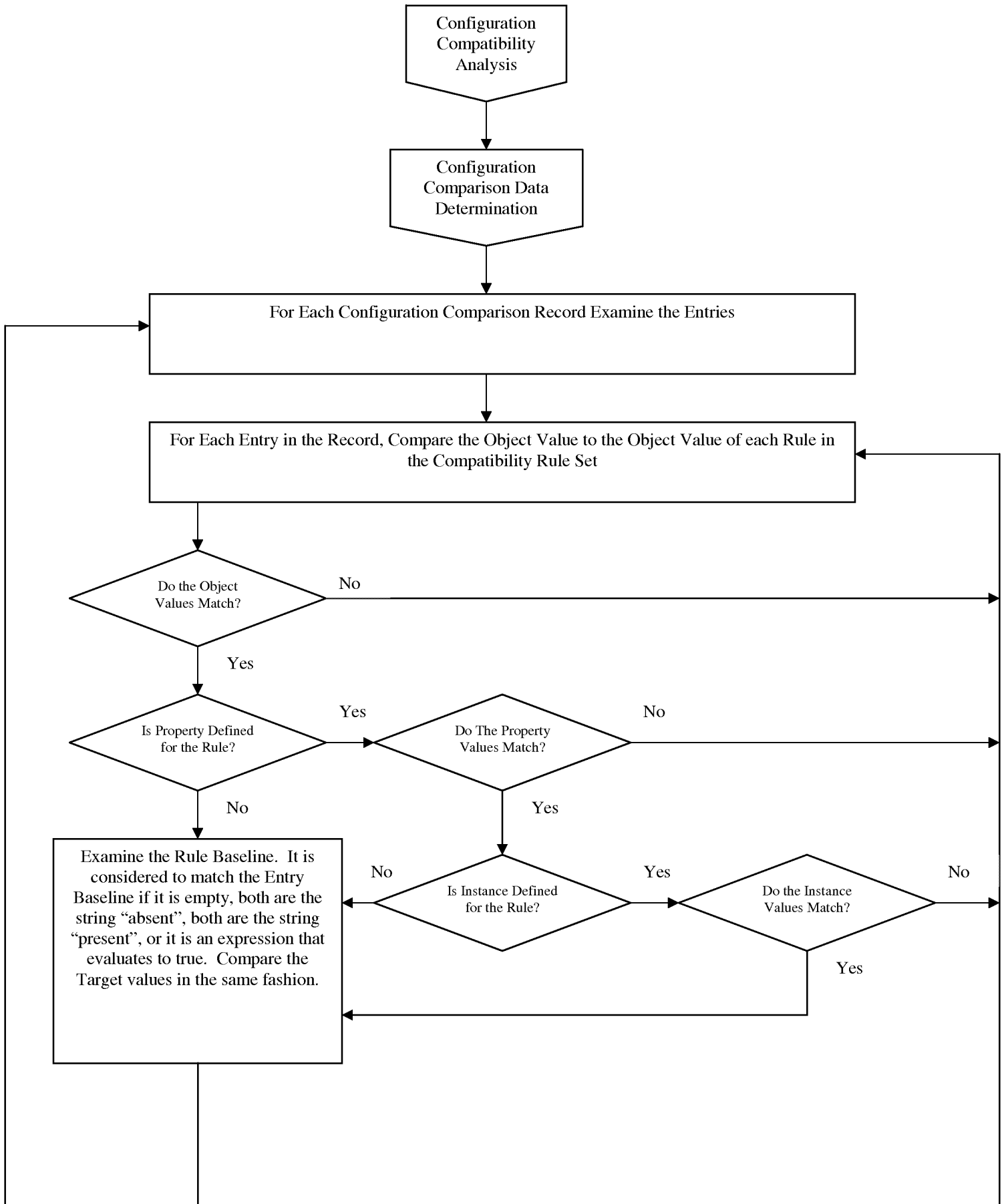


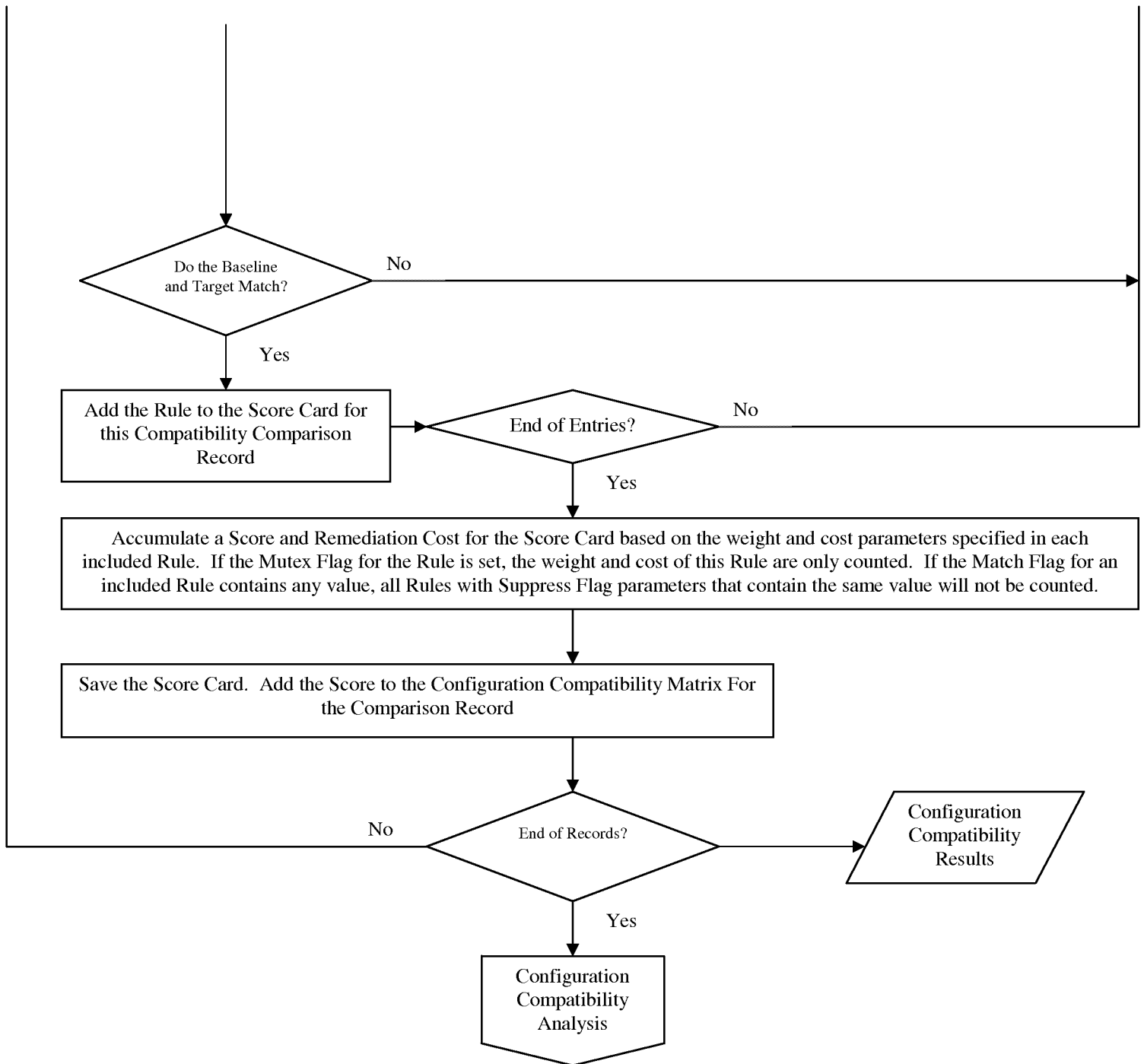


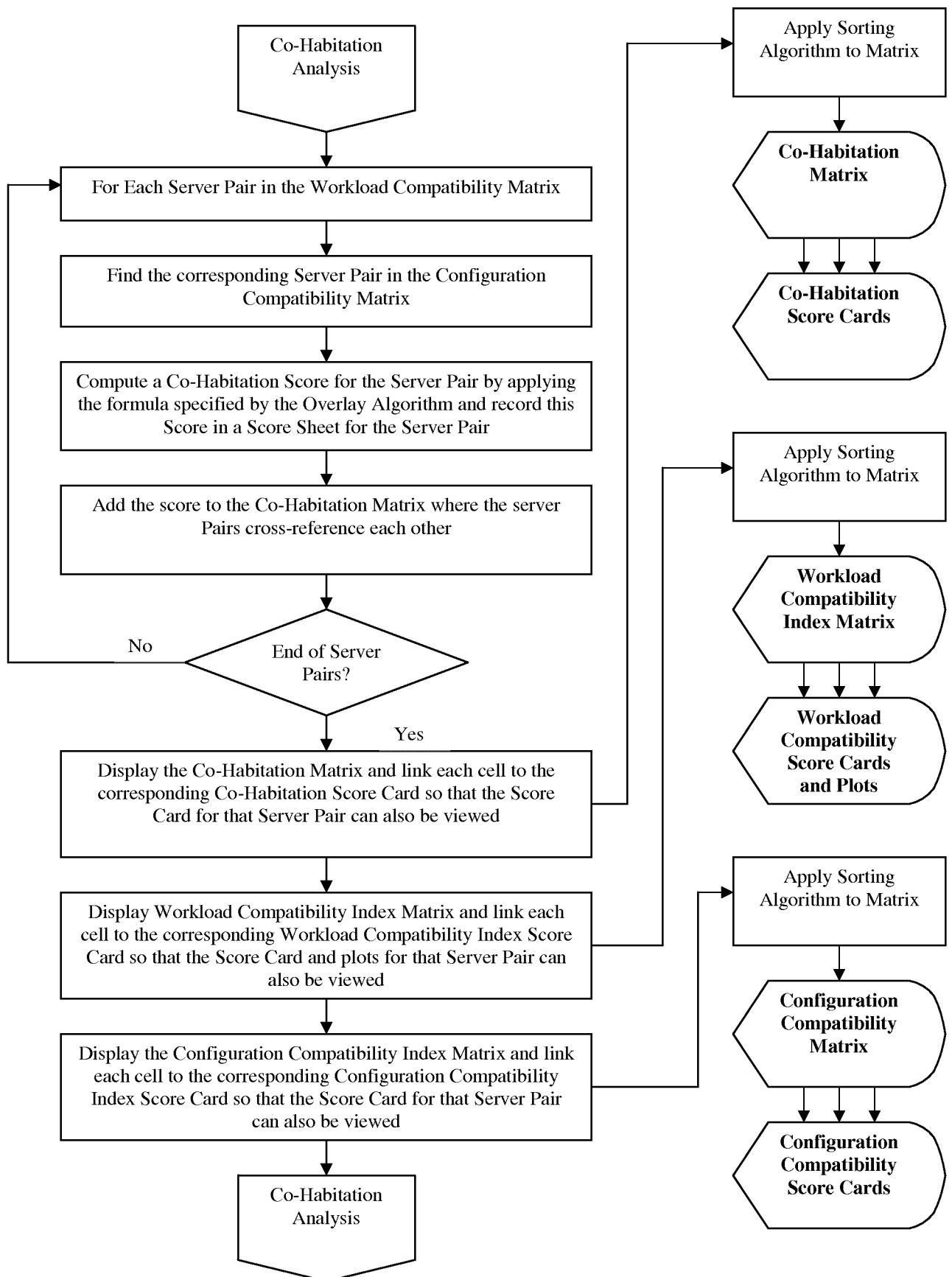












Sample Inputs and Outputs

Figure 1 Workload Usage Limit

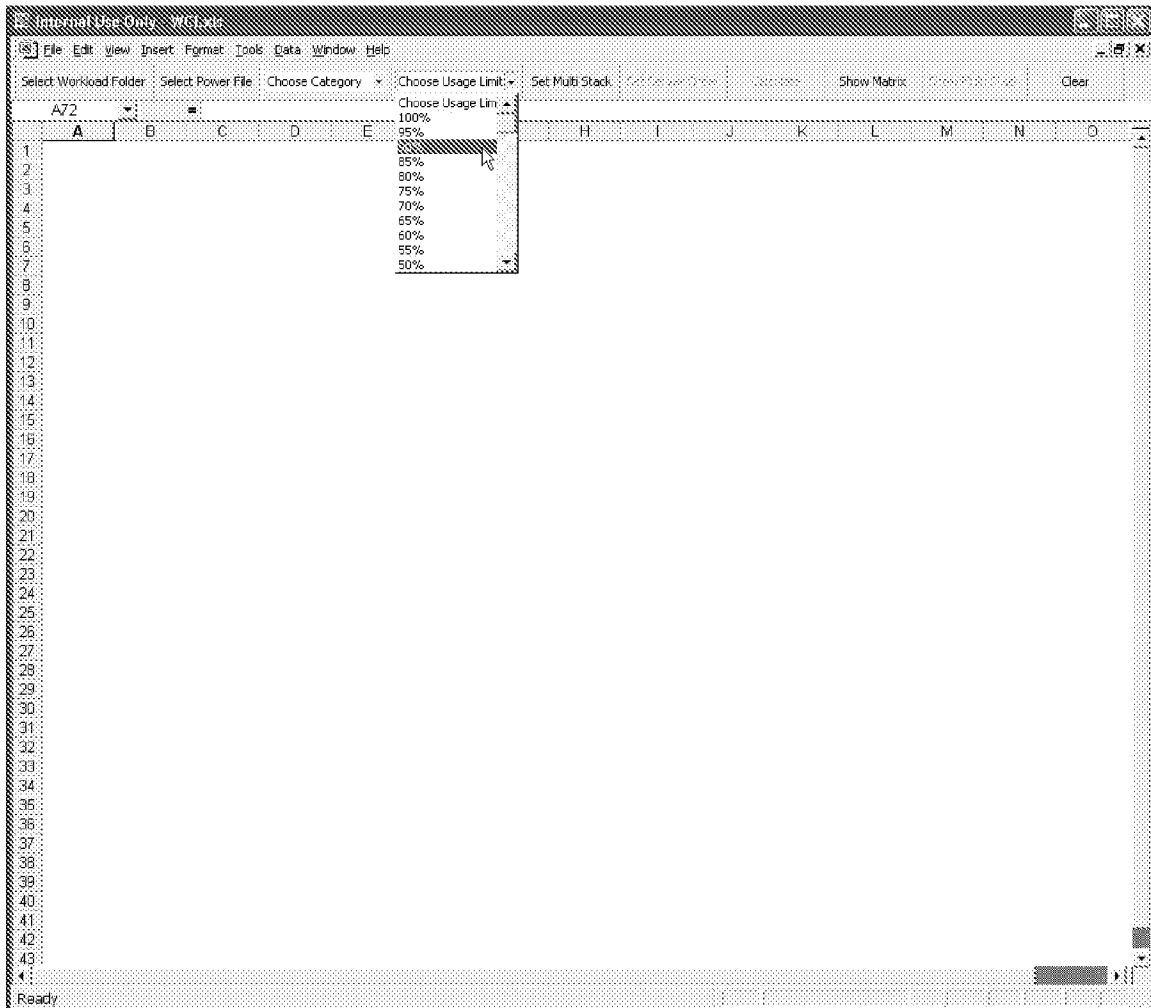


Figure 2 Workload Category

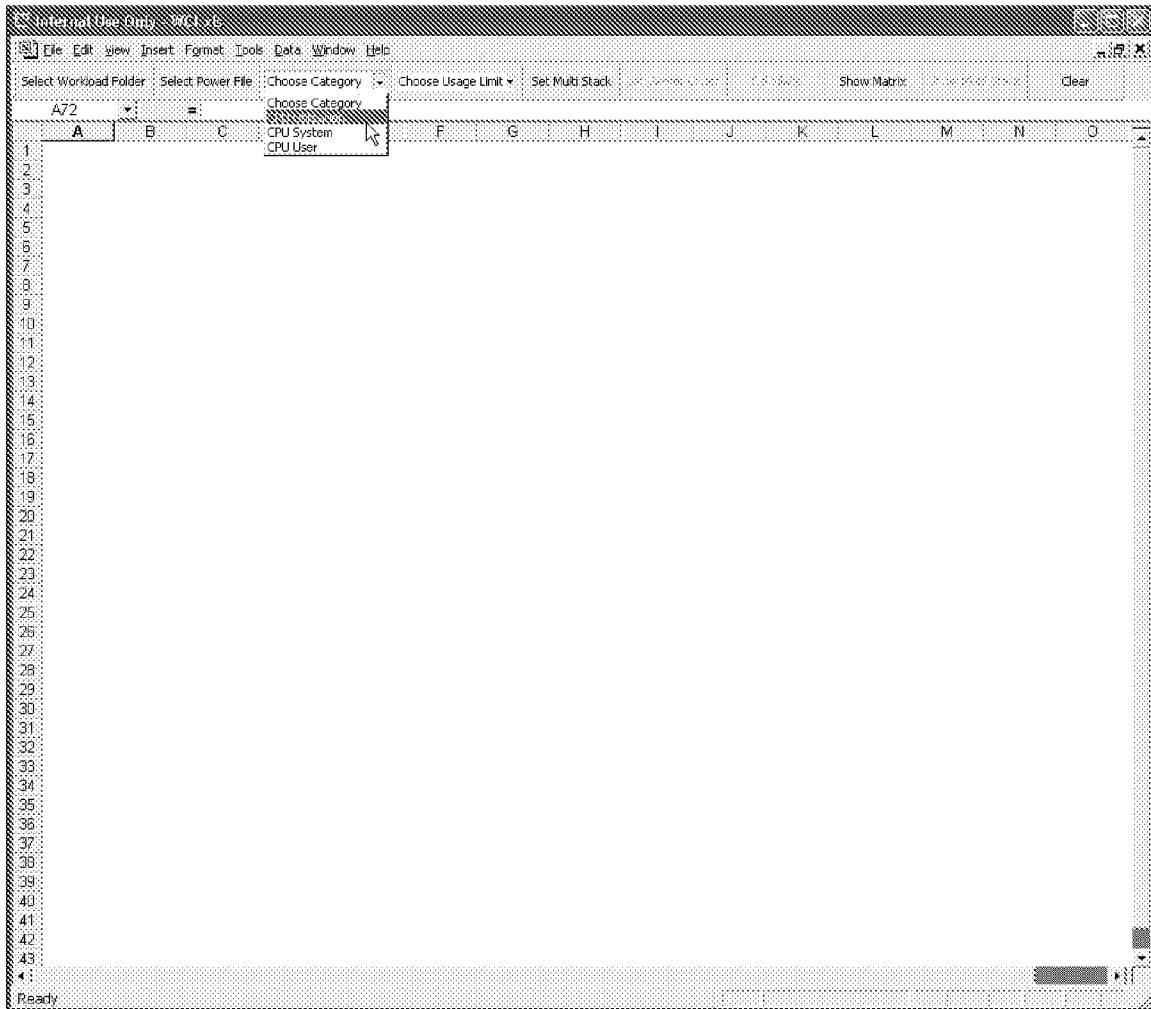


Figure 3 Detailed Workload Data

a433-key.int.cirba.com.csv													
File Edit View Insert Format Tools Data Window Help													
Select WCI File Select SCI File Average													
A1 * = #target													
#Target	Timestamp	Category	Property	Instance	Minimum	Q1	Q2	Q3	Maximum	Average	# Samples		
2	a433-key.int.cirba.com	20/2/2006 0:00	medium	cpu.pctidle	-	94	96	96	96	99	95.83333	12	
3	a433-key.int.cirba.com	20/2/2006 1:00	medium	cpu.pctidle	-	89	93	94	96	99	94.16667	12	
4	a433-key.int.cirba.com	20/2/2006 2:00	medium	cpu.pctidle	-	93	96	99	99	99	97.66667	12	
5	a433-key.int.cirba.com	20/2/2006 3:00	medium	cpu.pctidle	-	97	99	99	99	99	98.83333	12	
6	a433-key.int.cirba.com	20/2/2006 4:00	medium	cpu.pctidle	-	98	99	99	99	99	98.83333	12	
7	a433-key.int.cirba.com	20/2/2006 5:00	medium	cpu.pctidle	-	97	99	99	99	99	98.66667	12	
8	a433-key.int.cirba.com	20/2/2006 6:00	medium	cpu.pctidle	-	94	98	99	99	99	97.91667	12	
9	a433-key.int.cirba.com	20/2/2006 7:00	medium	cpu.pctidle	-	98	99	99	99	99	98.83333	12	
10	a433-key.int.cirba.com	20/2/2006 8:00	medium	cpu.pctidle	-	97	99	99	99	99	98.66667	12	
11	a433-key.int.cirba.com	20/2/2006 9:00	medium	cpu.pctidle	-	98	99	99	99	99	98.83333	12	
12	a433-key.int.cirba.com	20/2/2006 10:00	medium	cpu.pctidle	-	94	96	97	99	99	97.08333	12	
13	a433-key.int.cirba.com	20/2/2006 11:00	medium	cpu.pctidle	-	94	98	99	99	99	96.33333	12	
14	a433-key.int.cirba.com	20/2/2006 12:00	medium	cpu.pctidle	-	98	98	99	99	99	98.66667	12	
15	a433-key.int.cirba.com	20/2/2006 13:00	medium	cpu.pctidle	-	98	99	99	99	99	98.91667	12	
16	a433-key.int.cirba.com	20/2/2006 14:00	medium	cpu.pctidle	-	95	98	99	99	99	98.25	12	
17	a433-key.int.cirba.com	20/2/2006 15:00	medium	cpu.pctidle	-	94	99	99	99	99	98.41667	12	
18	a433-key.int.cirba.com	20/2/2006 16:00	medium	cpu.pctidle	-	98	98	98.5	99	99	98.5	12	
19	a433-key.int.cirba.com	20/2/2006 17:00	medium	cpu.pctidle	-	98	99	99	99	99	98.75	12	
20	a433-key.int.cirba.com	20/2/2006 18:00	medium	cpu.pctidle	-	0	99	99	99	99	90.58333	12	
21	a433-key.int.cirba.com	20/2/2006 19:00	medium	cpu.pctidle	-	93	95	98	99	99	97.08333	12	
22	a433-key.int.cirba.com	20/2/2006 20:00	medium	cpu.pctidle	-	96	99	99	99	99	98.58333	12	
23	a433-key.int.cirba.com	20/2/2006 21:00	medium	cpu.pctidle	-	97	99	99	99	99	98.5	12	
24	a433-key.int.cirba.com	20/2/2006 22:00	medium	cpu.pctidle	-	97	99	99	99	99	98.58333	12	
25	a433-key.int.cirba.com	20/2/2006 23:00	medium	cpu.pctidle	-	94	96	98.5	99	99	97.66667	12	
26	a433-key.int.cirba.com	20/2/2006 0:00	medium	cpu.pctsys	-	0	3	3	3	4	2.75	12	
27	a433-key.int.cirba.com	20/2/2006 1:00	medium	cpu.pctsys	-	0	3	4	5.5	6	3.91667	12	
28	a433-key.int.cirba.com	20/2/2006 2:00	medium	cpu.pctsys	-	0	0	0	3	6	1.41667	12	
29	a433-key.int.cirba.com	20/2/2006 3:00	medium	cpu.pctsys	-	0	0	0	0.5	2	0.33333	12	
30	a433-key.int.cirba.com	20/2/2006 4:00	medium	cpu.pctsys	-	0	0	0	0	1	0.16667	12	
31	a433-key.int.cirba.com	20/2/2006 5:00	medium	cpu.pctsys	-	0	0	0	0.5	3	0.41667	12	
32	a433-key.int.cirba.com	20/2/2006 6:00	medium	cpu.pctsys	-	0	0	0.5	2	3	1	12	
33	a433-key.int.cirba.com	20/2/2006 7:00	medium	cpu.pctsys	-	0	0	0	0.5	1	0.25	12	
34	a433-key.int.cirba.com	20/2/2006 8:00	medium	cpu.pctsys	-	0	0	0	1	3	0.5	12	
35	a433-key.int.cirba.com	20/2/2006 9:00	medium	cpu.pctsys	-	0	0	0	0	1	0.16667	12	
36	a433-key.int.cirba.com	20/2/2006 10:00	medium	cpu.pctsys	-	0	0	1	3	4	1.41667	12	
37	a433-key.int.cirba.com	20/2/2006 11:00	medium	cpu.pctsys	-	0	0	0	1	1	0.33333	12	
38	a433-key.int.cirba.com	20/2/2006 12:00	medium	cpu.pctsys	-	0	0	0	0	1	0.16667	12	
39	a433-key.int.cirba.com	20/2/2006 13:00	medium	cpu.pctsys	-	0	0	0	0	1	0.08333	12	
40	a433-key.int.cirba.com	20/2/2006 14:00	medium	cpu.pctsys	-	0	0	0	1	4	0.83333	12	
41	a433-key.int.cirba.com	20/2/2006 15:00	medium	cpu.pctsys	-	0	0	0	0.5	3	0.58333	12	
42	a433-key.int.cirba.com	20/2/2006 16:00	medium	cpu.pctsys	-	0	0	0.5	1	1	0.5	12	
43	a433-key.int.cirba.com	20/2/2006 17:00	medium	cpu.pctsys	-	0	0	0	1	2	0.41667	12	
44	a433-key.int.cirba.com	20/2/2006 18:00	medium	cpu.pctsys	-	0	0	0	0.5	61	5.25	12	
45	a433-key.int.cirba.com	20/2/2006 19:00	medium	cpu.pctsys	-	0	0	2	3	4	1.75	12	
46	a433-key.int.cirba.com	20/2/2006 20:00	medium	cpu.pctsys	-	0	0	0	1	3	0.5	12	
47	a433-key.int.cirba.com	20/2/2006 21:00	medium	cpu.pctsys	-	0	0	1	1	1	0.58333	12	
48	a433-key.int.cirba.com	20/2/2006 22:00	medium	cpu.pctsys	-	0	0	0	0.5	1	0.25	12	
49	a433-key.int.cirba.com	20/2/2006 23:00	medium	cpu.pctsys	-	0	0	1	2.5	3	1.25	12	
50	a433-key.int.cirba.com	20/2/2006 0:00	medium	cpu.pctusr	-	0	1	1	1	3	1	12	
51	a433-key.int.cirba.com	20/2/2006 1:00	medium	cpu.pctusr	-	0	1	1	2	3	1.16667	12	
52	a433-key.int.cirba.com	20/2/2006 2:00	medium	cpu.pctusr	-	0	0	0	1	1	0.33333	12	
53	a433-key.int.cirba.com	20/2/2006 3:00	medium	cpu.pctusr	-	0	0	0	0	1	0.16667	12	
54	a433-key.int.cirba.com	20/2/2006 4:00	medium	cpu.pctusr	-	0	0	0	0	1	0.16667	12	
55	a433-key.int.cirba.com	20/2/2006 5:00	medium	cpu.pctusr	-	0	0	0	0.5	2	0.33333	12	
56	a433-key.int.cirba.com	20/2/2006 6:00	medium	cpu.pctusr	-	0	0	0	0.5	1	0.25	12	
57	a433-key.int.cirba.com	20/2/2006 7:00	medium	cpu.pctusr	-	0	0	0	0	1	0.16667	12	
58	a433-key.int.cirba.com	20/2/2006 8:00	medium	cpu.pctusr	-	0	0	0	0.5	1	0.25	12	
59	a433-key.int.cirba.com	20/2/2006 9:00	medium	cpu.pctusr	-	0	0	0	0.5	2	0.33333	12	

Figure 4 Benchmark Data

	A	B	C	D	E	F	G	H	I	J	K
1	#Target	CINT2000 Total									
2	a433-key.int.cirba.com	95									
3	aix.int.cirba.com	210									
4	aix53-32.int.cirba.com	1090									
5	aix53-64.int.cirba.com	1090									
6	aix64.int.cirba.com	1440									
7											
8											
9											
10											
11											
12											
13											
14											
15											
16											
17											
18											
19											
20											
21											
22											
23											
24											
25											
26											
27											
28											
29											
30											
31											
32											
33											
34											
35											
36											
37											
38											
39											
40											
41											
42											
43											
44											
45											

Figure 5 Multistack GUI

The image shows a window titled "Multi Stack" with a close button in the top right corner. The window contains three radio button options:

- ☐ No Multi Stack
- ☐ Multi Stack on Existing Server
- ☒ Multi Stack on Fictitious Server

Below the third option, there are two input fields:

- Name:
- Power:

At the bottom of the window are two buttons: "OK" and "Cancel".

Figure 6 Server Configuration Data Record

Kernel Parameters Table [Module Index](#)

Source: CiRBA Config Monitor Module

<i>Parameter</i>	<i>Value</i>
bufhwm	10346496
v.v_proc	7898
MAXCLSYSPRI	99
v.v_maxup	7893
NAUTOUP	30
GPGSLO	25
FSFLUSHR	5
MINARMEM	25
MINASMEM	25
REL	5.8
NODE	wise
SYS	SunOS
VER	Generic_108528-14
NSTRPUSH	9
STRMSGSZ	65536
STRCTLSZ	1024
SHMMAX	1048576
SHMMIN	1
SHMMNI	100
SHMSEG	6
TSMAXUPRI	60
SYS_NAME	SYS

Locale Settings [Module Index](#)

Source: CiRBA Config Monitor Module

Time Zone US/Eastern

Figure 7 Compatibility Rule Set

Compatibility Rule Set									
A	B	C	D	E	F	G	H	I	J
Rule	Instance	Baseline	Target	Weight	Remediation Cost	Monitor Flag	Watch Flag	Suppress Flag	Description
1: CIRBA Config Monitor Server Information Operating System Version Number				2%	\$100.00		OS		Different Major Operating System Versions
2: CIRBA Config Monitor Server Information Operating System Version				3%			REV	OS	Different Minor Operating System Versions
3: CIRBA Patch Auditor Installed Patch Table			absent	3%		Y		OS REV	Patch Differences Between Systems
4: CIRBA Config Monitor Server Information Physical Memory			<	5%					Target Has Less Memory
5: CIRBA Config Monitor Kernel Parameters Table Value	NOPE			0%					
6: CIRBA Config Monitor Kernel Parameters Table Value	VER			0%					
7: CIRBA Config Monitor Kernel Parameters Table Value	REL			0%					
8: CIRBA Config Monitor Server Information Host Platform Name				1%					Different host model
9: CIRBA Config Monitor Server Information Operating System Kernel Bits				2%					Not running same kernel bits
10: CIRBA Config Monitor Local Settings Time Zone				50%					Different time zone/regions
11: CIRBA Software Inventory Software Application Inventory Table	Resin Webserver	present	absent	0%					Resin Webserver missing
12: CIRBA Software Inventory Software Application Inventory Table	Samba	present	absent	1%					Samba Should be present
13: CIRBA Software Inventory Software Application Inventory Table	Adin	present	absent	5%			APP		Adin Should be present
14: CIRBA Software Inventory Software Application Inventory Table	Sybase Client	present	absent	5%			SY		Sybase Should be present
15: CIRBA Software Inventory Software Application Inventory Table	Smart Sockets	present	absent	5%					
16: CIRBA Software Inventory Software Application Inventory Table	Perl	present	absent	0%					
17: CIRBA Software Inventory Software Application Inventory Table	oracle	present	absent	5%			OR		
18: CIRBA Software Inventory Software Application Inventory Table Version	Sybase Client			1%			SY		Sybase has to be the same version
19: CIRBA Software Inventory Software Application Inventory Table Version	oracle			1%			OR		Oracle has to be the same version
20: CIRBA Config Monitor Kernel Parameters Table				2%		Y			
21: CIRBA Config Monitor Kernel Parameters Table Value				1%		Y			
22: CIRBA Software Inventory Software Application Inventory Table	CIRBA_CAT	present	absent	2%			APP		CIRBA Server must be there
23: CIRBA Software Inventory Software Application Inventory Table	Java	present	absent	1%		Y		APP	Java must be there
24: CIRBA Software Inventory Software Application Inventory Table Version	Java			1%		Y		APP	
25: CIRBA Software Inventory Software Application Inventory Table	apache	present	absent	1%					Apache must be present
26: CIRBA Config Monitor Installed Locales				1%		Y			Local settings different
27: CIRBA Config Monitor NonVolatile Settings (PROMBIOS)				1%		Y			PROMBIOS setting different
28: IETF MIB2 Network Interface Information Number of Network Interfaces		<		2%					Number of Network Cards is different
29: IETF MIB2 Network Interface Table		present	absent	0%					
30: CIRBA Config Monitor Cron Jobs Number of Cron Jobs	admindm	present	absent	0%					
31: CIRBA Config Monitor Cron Jobs Number of Cron Jobs	sybase	present	absent	0%					
32: CIRBA Config Monitor Cron Jobs				1%		Y			

Figure 8 Configuration Comparison Data

Module	Object	Property	Instance	Baseline Value	Target Value
CIRBA Config Tracker	Installed Patch Table	MINSearch bin DECS	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	MINSearch bin DECS	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	MINSearch bin DECS	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	MINSearch bin DECS	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	MINSearch bin DECS	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	Jawa sdk decs	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	Jawa sdk includes	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	Jawa sdk src	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	Jawa run-app.docx	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	Jawa run-app.lib	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	Jawa run-app.samples	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	Jawa rtf dr	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	Jawa rtf lib	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	Jawa rtf classes	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	Jawa rtf lib	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	Jawa samples AB-Demos	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	Jawa samples demos	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	Jawa samples examples	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	Jawa security lib	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	Jawa swing ab	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	Jawa swing src	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 Dr Toolkit	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 Dr alt	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 Dr bitmaps	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 Dr compat	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 Dr helpwin	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 Dr helpwin	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 Dr lib	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 Dr me	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 Dr subtools	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 drt bitmaps	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 drt ext	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 drt icons	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 drt include	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 drt lib	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 drt motif	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 apps autorm	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 apps clients	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 apps config	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 apps custom	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 apps event	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 apps pm	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 apps rfc	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 apps util	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 apps xdmcp	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 base common	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 compat set Motif 2	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 compat for pc	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 compat lib Motif 4	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 compat lib X11R2	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 compat lib X11R4	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 compat lib X11R5	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 lib comd	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 lib defaultFont	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 lib font	absent	000000	000000
CIRBA Patch Auditor	Installed Patch Table	X11 lib on Linux Dr helpwin	absent	000000	000000

Figure 9 Overlay Algorithm

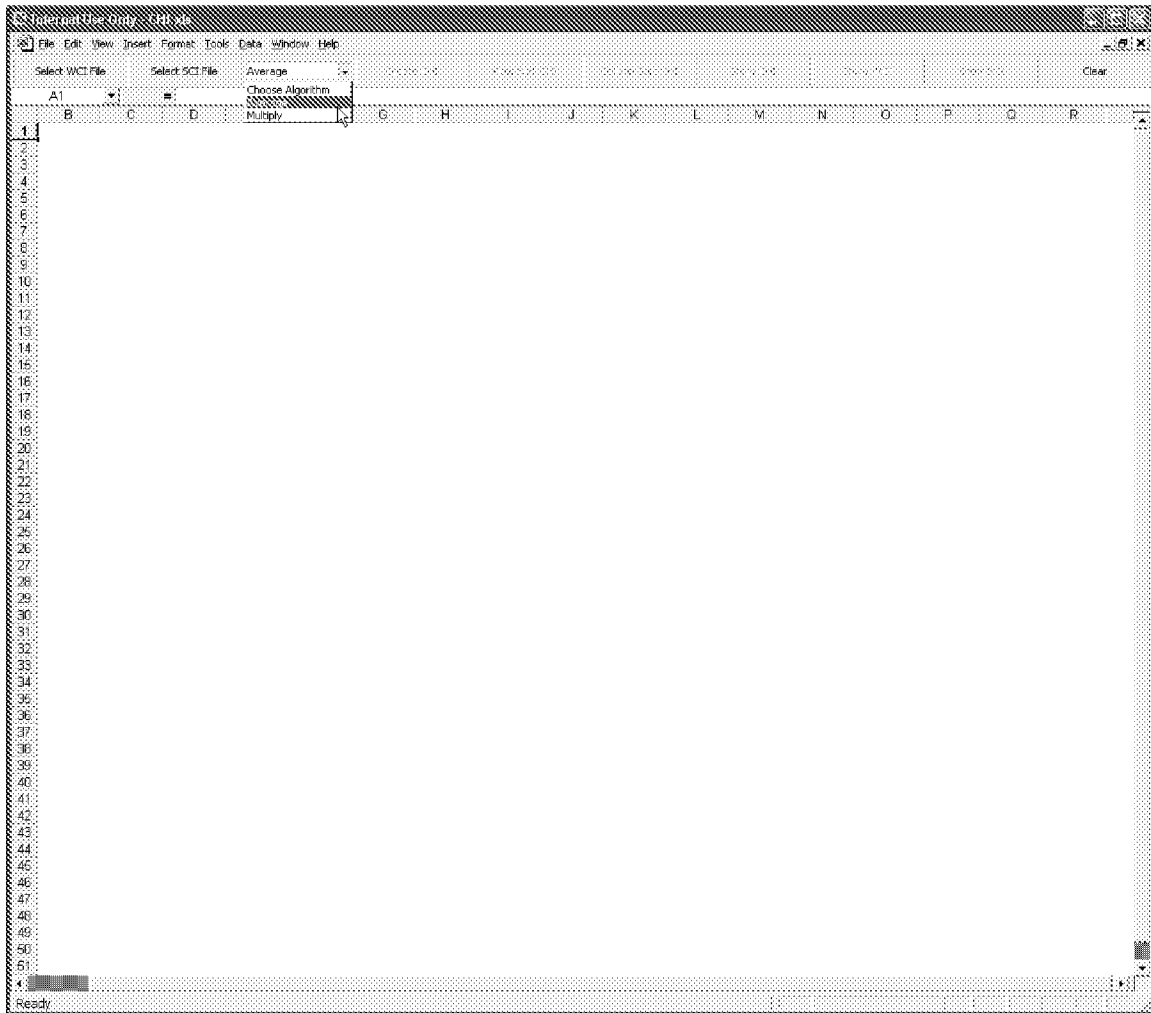


Figure 10 Co-Habitation Matrix

	A	B	C	D	E	F
1	CHI%	aix53-64.int.cirba.com	aix53-32.int.cirba.com	aix64.int.cirba.com	aix.int.cirba.com	a433-key.int.cirba.com
2	aix53-64.int.cirba.com	100	60	77	76	80
3	aix53-32.int.cirba.com	58	100	53	38	23
4	aix64.int.cirba.com	62	34	94	45	44
5	aix.int.cirba.com	91	70	82	98	46
6	a433-key.int.cirba.com	94	72	88	87	99

Figure 11 Co-Habitation Score Card

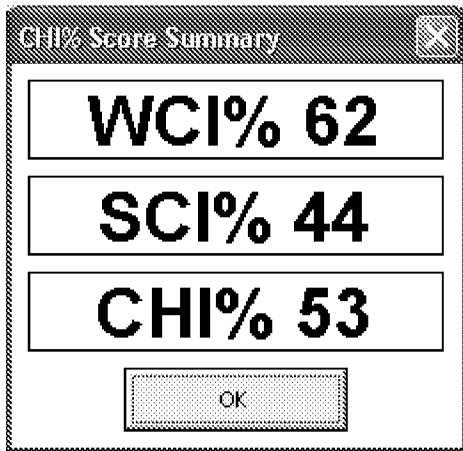


Figure 12 Workload Compatibility Matrix

	A	B	C	D	E	F
1	WCI%	aix53-64.int.cirba.com	aix53-32.int.cirba.com	aix64.int.cirba.com	aix.int.cirba.com	a433-key.int.cirba.com
2	aix53-64.int.cirba.com	100	71	64	63	70
3	aix53-32.int.cirba.com	71	100	62	30	0
4	aix64.int.cirba.com	64	62	100	0	0
5	aix.int.cirba.com	63	30	0	100	0
6	a433-key.int.cirba.com	70	0	0	0	100

Figure 13 Workload Compatibility Score Card

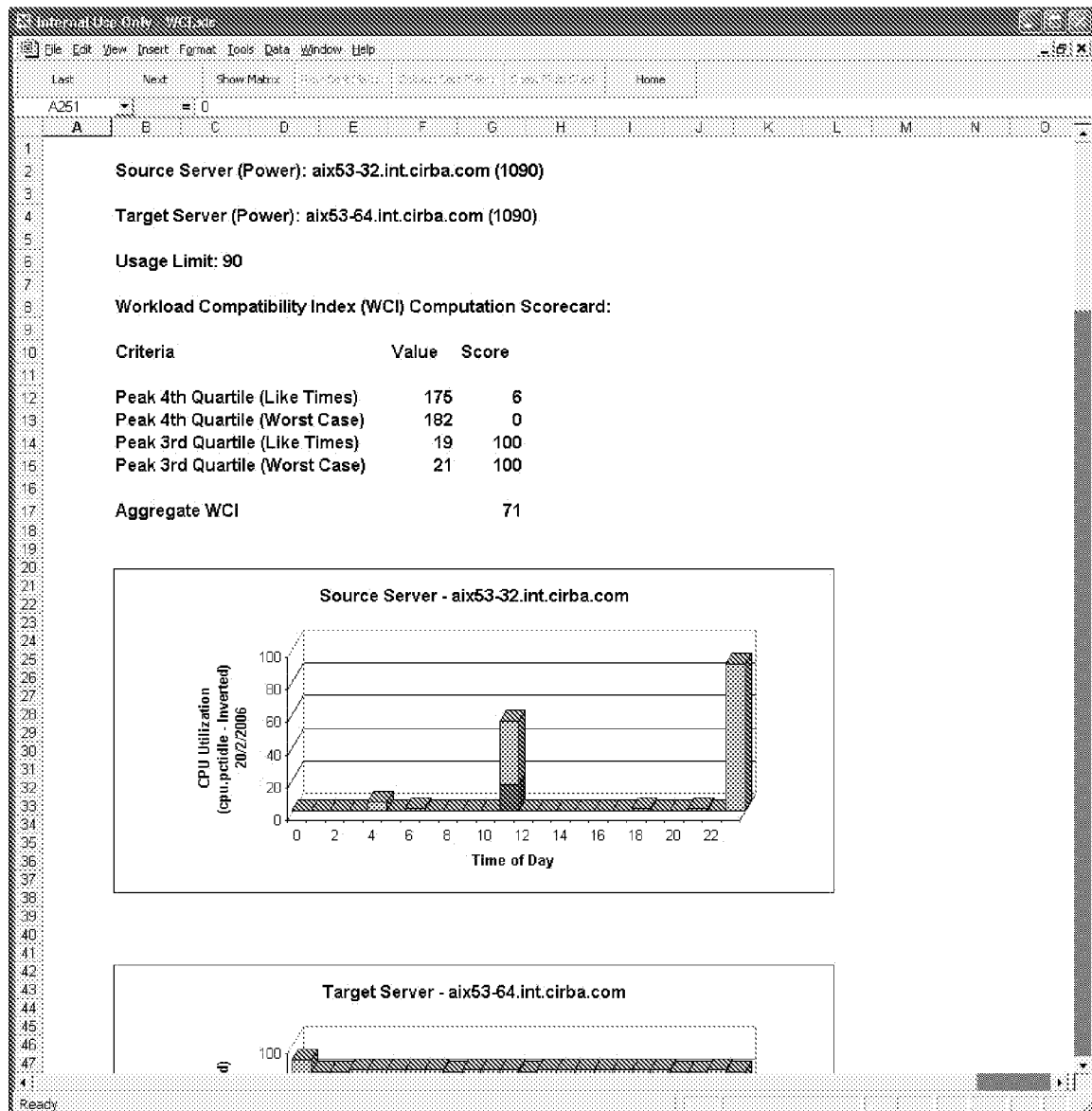


Figure 14 Workload Compatibility Plots 1

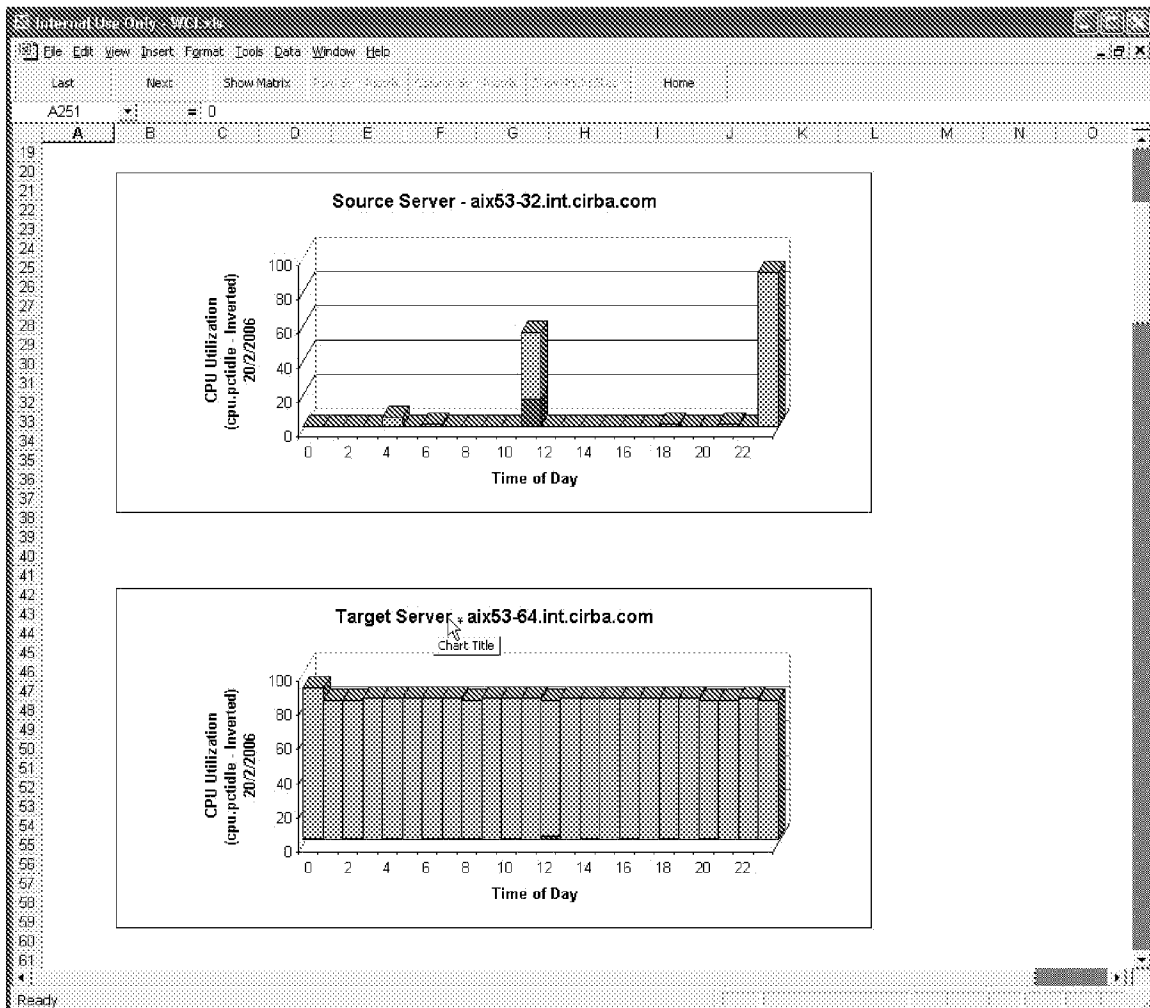


Figure 15 Workload Compatibility Plots 2

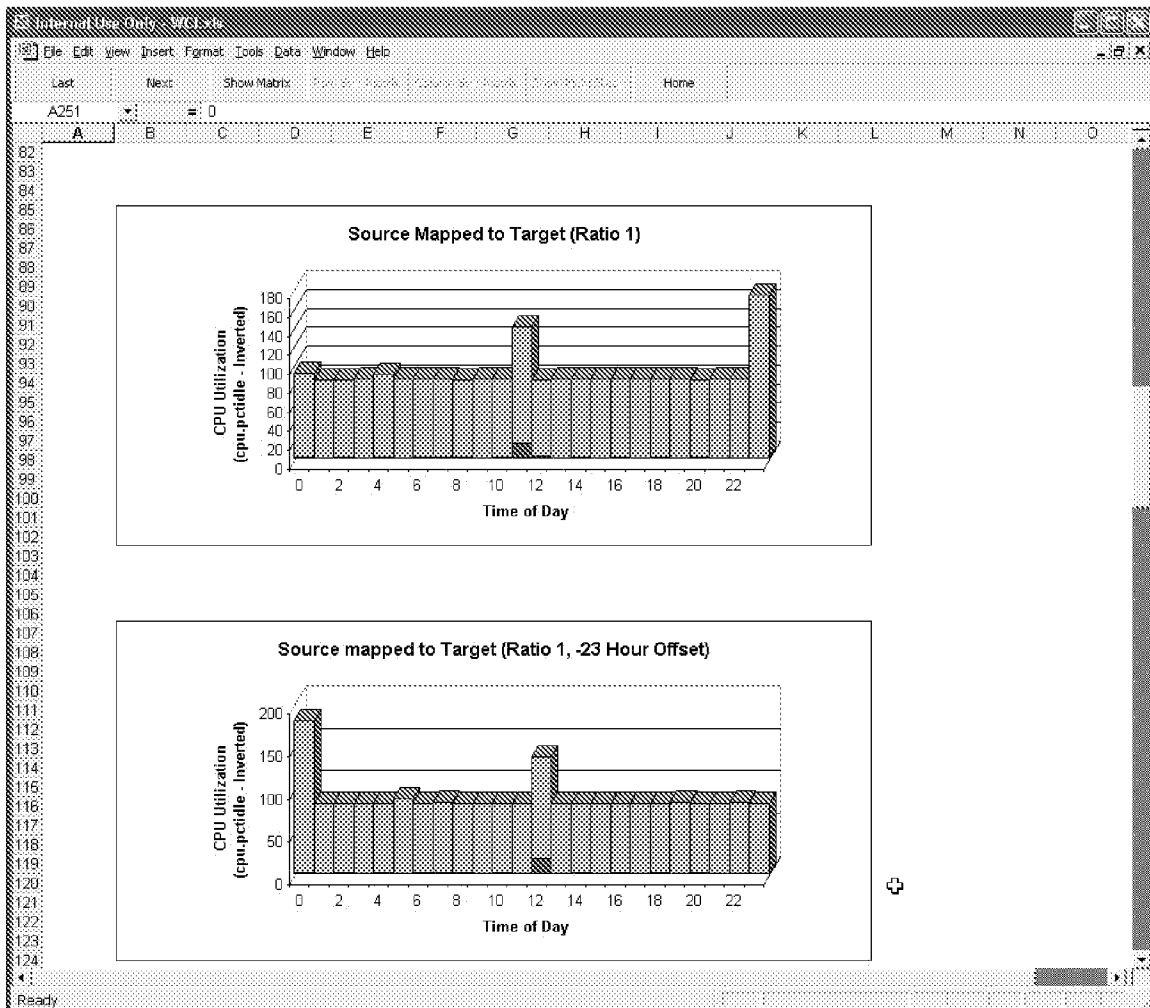


Figure 16 Configuration Compatibility Matrix

	A	B	C	D	E	F
1	SCI%	aix.int.cirba.com	aix53-64.int.cirba.com	a433-key.int.cirba.com	aix64.int.cirba.com	aix53-32.int.cirba.com
2	aix.int.cirba.com	100	89	92	90	46
3	aix53-64.int.cirba.com	89	100	89	90	42
4	a433-key.int.cirba.com	92	89	100	88	48
5	aix64.int.cirba.com	90	90	89	100	44
6	aix53-32.int.cirba.com	46	46	46	44	100

Figure 17 Configuration Compatibility Score Card

System Compatibility Index (SCI) Percentage							
1	System Compatibility Index (SCI) Percentage		90%				
2			\$500				
3							
4	Different Major Operating System Versions		5%				
5	Different Minor Operating System Versions		3%				
6	Patch Differences Between Systems		3%				
7	Different host model		1%				
8	PROM BIOS setting different		1%				
9							
10							
11							
12	Root	Distance	Baseline	Target	Weight	Remediation Cost	Need Remed
13	CIRBA Config Monitor Kernel Parameters Table	capacity_inc	present	present	2%		
14	CIRBA Config Monitor Kernel Parameters Table	cpupage	present	present			
15	CIRBA Config Monitor Kernel Parameters Table Value	cpuguard	enable	enable	1%		
16	CIRBA Config Monitor Kernel Parameters Table	dedicated	present	present			
17	CIRBA Config Monitor Kernel Parameters Table	ent_capacity	present	present			
18	CIRBA Config Monitor Kernel Parameters Table Value	frequency	50000000	914257			
19	CIRBA Config Monitor Kernel Parameters Table Value	hwmon	IBM SP222 (P4)	IBM SP4300			
20	CIRBA Config Monitor Kernel Parameters Table Value	id_to_partition	06000110C85A0000	0420686600000000			
21	CIRBA Config Monitor Kernel Parameters Table Value	id_to_system	06000110C85A0000	0420686600000000			
22	CIRBA Config Monitor Kernel Parameters Table	max_capacity	present	present			
23	CIRBA Config Monitor Kernel Parameters Table	max_bpsrate	present	present			
24	CIRBA Config Monitor Kernel Parameters Table	min_capacity	present	present			
25	CIRBA Config Monitor Kernel Parameters Table Value	modelName	IBM 9111-520	IBM 1044-170			
26	CIRBA Config Monitor Kernel Parameters Table Value	realmem	786432	524288			
27	CIRBA Config Monitor Kernel Parameters Table Value	systemid	IBM 021085FHE	IBM 011064FHE			
28	CIRBA Config Monitor Kernel Parameters Table	variable_weight	present	present			
29	CIRBA Config Monitor NonVolatile Settings (PROM BIOS)	cd0	absent	present	1%		PROM BIOS setting different
30	CIRBA Config Monitor Server Information Host Platform Name		IBM 9111-520	IBM 1044-170	1%		Different host model
31	CIRBA Config Monitor Server Information Operating System Version Number		5.3.0.0	5.2.0.0	5%		Different Major Operating System Versions
32							
33							
34							
35							
36							
37							
38							
39							
40							
41							
42							
43							
44							
45							
46							
47							
48							
49							
50							
51							
52							
53							
54							
55							
56							
57							
58							
59							
60							
61							
62							
63							
64							
65							
66							
67							
68							
69							
70							
71							
72							
73							
74							
75							
76							
77							
78							
79							
80							
81							
82							
83							
84							
85							
86							
87							
88							
89							
90							
91							
92							
93							
94							
95							
96							
97							
98							
99							
100							

Figure 18 Multistack Score Card

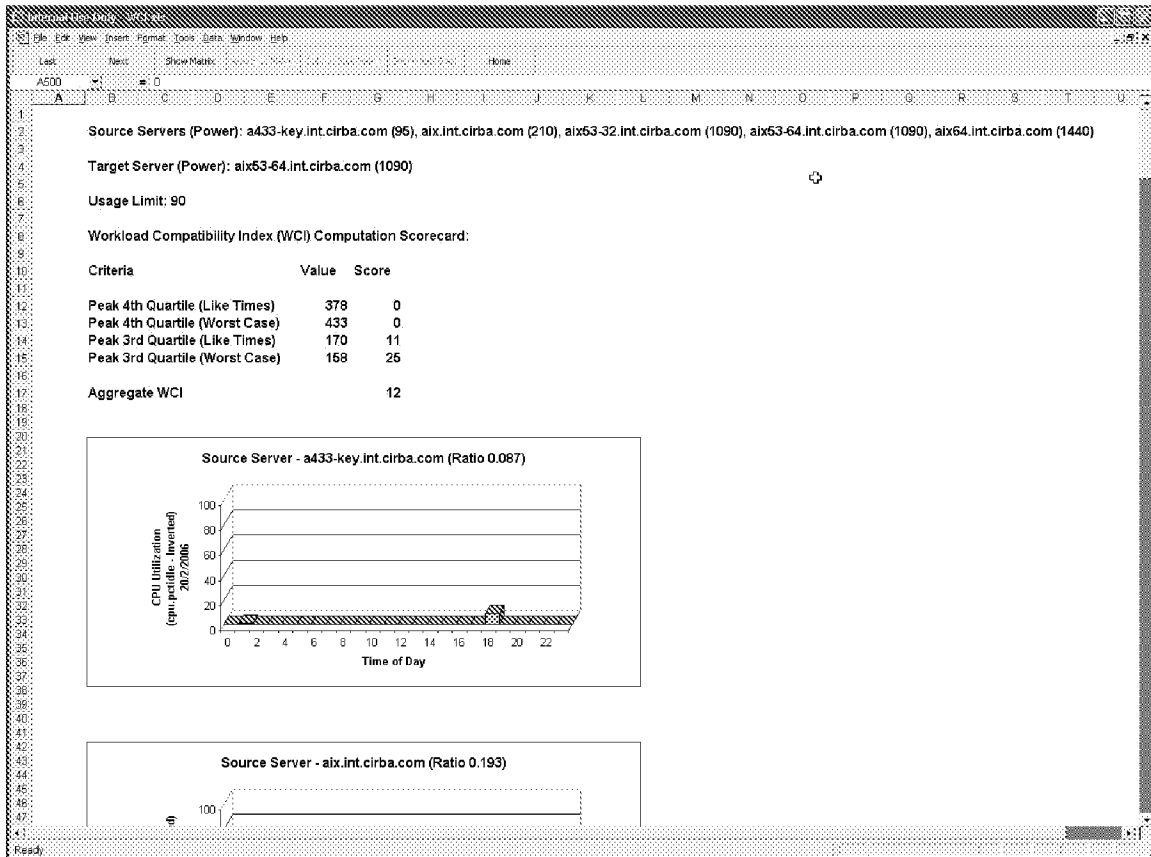


Figure 19 Multistack Plots

