

ULACIT
UNIVERSIDAD LATINOAMERICANA DE CIENCIA Y
TECNOLOGÍA

PROGRAMA DE DOCTORADO EN CIENCIAS ECONÓMICAS
Y EMPRESARIALES
(PROLACE)

TESIS DE DOCTORADO

DISEÑO DE UN MODELO MATEMÁTICO QUE DETERMINE LA
EFICIENCIA DE LOS ALGORITMOS DE DISTRIBUCIÓN DE
CARGA EN LA INFORMÁTICA DISTRIBUIDA (GRID
COMPUTING) APLICADA A LOS SECTORES ECONÓMICOS Y
EMPRESAS

EDUARDO CASTRO MARTÍNEZ

San José, Costa Rica

FEBRERO 2009

TRIBUNAL EXAMINADOR

AUTORIDADES DE LA UNIVERSIDAD LATINOAMERICANA DE
CIENCIA Y TECNOLOGÍA-ULACIT

(RECTOR)

DRA. SARA CORDERO
DIRECTOR DE PROGRAMA DE POSTGRADO

AUTORIDADES DEL DOCTORADO

DR. LEOPOLDO SANTIBAÑEZ
(PROFESOR TUTOR)

DR. SAÚL NUÑEZ
(LECTOR)

DRA. EUGENIA RODRIGUEZ
(LECTOR)

DEDICATORIA

*A los científicos de ayer, hoy y de siempre
que viven mirando la innovación y el
crecimiento del desarrollo de la
tecnología.*

AGRADECIMIENTOS

Por todo el conocimiento compartido deseo agradecer al Dr. Leopoldo Santibañe, al Dr. Saúl Nuñez y a la Dra Eugenia Rodríguez por la motivación y apoyo para realizar esta tesis. Especialmente le agradezco a Joaquín Zuñiga por todo su esfuerzo durante las todas las etapas de realización de esta tesis.

ABSTRACT

The hereby research, aside from fulfilling a lifelong dream, also meets a graduation requirement of the Latin American Economics and Business Program in the attainment of a Doctoral degree in Economics and Business, with emphasis in Economics.

The Grid computing technology is emerging as a new paradigm for the next-generation businesses and enables the sharing, selection, and use of geographically distributed computational resources for solving problems in science, engineering, corporations and commerce.

Grid computing is focused in providing easier access to remote processing resources that are usually locally limited. The main idea is to join distributed computer systems in such a grid environment, in which users and companies can submit processing jobs that are automatically assigned to proper processing resources. Grid computing takes a non typical approach by including networks, data, small and big computers. In addition to access unavailable local resources the companies can use a larger number of resources for a single job, which will lead to a reduction of the average job response and execution time.

Due to the resource distribution and the different management policies, the management of the grid environment becomes rather complex, especially the scheduling and evaluation of the grid efficiency. To this end, the economic models for the scheduling and grid efficiency measurement are an adequate way to solve this problem. These economic models provide support for access and service policies to be applied by resource owners to the grid users. The use of economic models allows the management to include cost analysis and helps to measure the grid overall use. In this research one presents a mathematical and economic model to evaluate the efficiency of the grid systems, this model helps the grid owners to be able to compare its efficiency with other grid systems, and they can analyze why a given grid is more efficient than other and then take the actions to improve inefficient grid systems.

The scheduling is the mechanism used to allocate computational resources to problems over time. In grid computing these problems are called jobs, they are described by several parameters like run time, number of processors needed, submission time, finished time, required processing time, etc. In this research one evaluates different economic models used to allocate resources: time shared and space shared, one uses simulations and real data to determine the costs related with the used of this economic models in grid environments. Also the jobs parameters are analysed by means of statistical comparison.

The grid computing vision promises to provide the new platform to support an innovative and more demanding range of enterprise applications. For this to become true, the design and deployment of adequate resource management and information services need to be created. In this context, understanding the characteristics of real grid systems is a crucial step for improving the economic efficiency of existing grid services; a method to isolate the efficiency problems needs to be created, in this research one defined such method based on an economic based mathematical model. Towards this goal, in this research we analyse different work loads of real Grid Environments which are among the largest production grids currently deployed. One focuses the analysis on the economic impact of the grid management and on individual jobs characteristics, one further attempt to quantify the efficiency of the whole grid systems as way to measure and improve the performance of the grid systems.

This research also addresses the job scheduling algorithms as part of the grid management infrastructure; one also includes the analysis of network and data management, information collection of job execution and data transfer over the network.

In this research one defines the economic and mathematical model based on the study and analysis of the following main areas: economic management (IGE), infrastructure

efficiency (IEI) and operational efficiency (IGO). The analysis of each area is supported by several indicators which are explained in the detail in the methodology section.

The resources in the Grid are distributed in different geographical locations and the availability and cost policies vary depending on the priorities, use time, business goals and administrative procedures. The resource management in such large-scale environment is a complex task. This thesis proposes a distributed mathematical and economy based algorithm as an effective way to measure the economic performance of the computational resources. It proposes a mechanism (IBECOD) to build a ranking of grid systems in order to evaluate the efficiency using the following criteria: economic management (IGE), infrastructure efficiency (IEI) and operational efficiency (IGO).

The economic evaluation of grid systems efficiency, subject of this research, is a tool to be used by grid owners in order to evaluate different aspects of their current systems; this evaluation will help them to identify the parts of the grid that need to be improved so the efficiency of the whole system is optimal. This thesis defines the IBECOD index which resumes the global grid efficiency by measuring 25 different indicators on three main areas: economic management, infrastructure efficiency and operational efficiency.

This research is based on a descriptive and explanatory methodology. As instruments and tools for the data collection, several methods were applied: surveys, interviews, paper revisions, system log files analysis, simulations and economic analysis.

Given the incompleteness of the information available for analysis purposes, one focuses on a specific period of time of three years based on the Grid Workloads Archive. In this research one discusses the applicability of the designed model in three areas of grid computing: economic models, scheduling algorithms and benchmarking.

Finally, one concludes that the design of the research methodology allows the implementation and creation of an mathematical and economic based model to evaluate the grid efficiency, this model was proven as valid method to help grid administrators to find the elements that need to be improved in the infrastructure in order to get the best economic return value of the grid system.

PRESENTACIÓN

La investigación realizada, además de concluir con un antiguo anhelo de escribir parte de mis sueños, a la vez, satisface el requisito de graduación del Programa Latinoamericano de Ciencias Económicas y Empresariales para alcanzar el grado de Doctor en ciencias Económicas y Empresariales con énfasis en Economía.

La tecnología de “grid computing” está emergiendo como un nuevo paradigma para los negocios neo-generacionales; por otra parte, permite compartir, seleccionar y utilizar recursos computacionales geográficamente distribuidos con el fin de resolver problemas científicos, de ingeniería, de corporaciones y comercios.

El grid computing está enfocado en proveer fácil acceso a recursos de procesamiento remoto que son limitados localmente. La idea principal es unir sistemas de computación distribuidos en un ambiente de grid, en el cual los usuarios y compañías pueden enviar trabajos de procesamiento que son asignados automáticamente a los recursos de procesamiento apropiados. Grid computing opta por un enfoque no tradicional ya que incluye redes, datos, computadores pequeños y grandes. Además de acceder recursos que son disponibles localmente, las empresas pueden utilizar una cantidad más grande de recursos para ejecutar un trabajo, lo cual conlleva una reducción en el tiempo de respuesta y del tiempo de ejecución del mismo.

Debido a la distribución de recursos y distintas políticas de administración, la administración de un ambiente de grid se vuelve compleja, especialmente la calendarización y la evaluación de la eficiencia del grid. Para resolver este problema los modelos económicos aplicados a la calendarización y medición de la eficiencia son una solución adecuada. Estos modelos económicos proveen soporte que las políticas de acceso y servicio sean aplicadas por los dueños del grid. El uso de modelos económicos permite a la administración incluir análisis de costos y ayuda a medir el uso global del grid. En esta

investigación se presenta un modelo económico matemático que evalúa la eficiencia de los sistemas de grid computing, este modelo ayuda a los dueños del grid para que realicen comparaciones de su eficiencia con otros sistemas, y de esa forma analizar por qué uno es más eficiente que el otro, esto con el fin de tomar las medidas correctivas necesarias.

La calendarización es el mecanismo utilizado para asignar recursos computacionales a problemas. En grid computing estos son conocidos como trabajos, los cuales son descritos mediante distintos parámetros tales como tiempo de ejecución, cantidad de procesadores necesitados, tiempo de inicio, tiempo de finalización, tiempo de procesamiento necesario, entre otros. En esta investigación se evalúan distintos modelos económicos utilizados para asignar recursos: tiempo compartido y espacio compartido, se utilizan simulaciones y datos reales para determinar los costos relacionados con el uso de estos modelos económicos en ambientes de grid computing. Además, se analizan los parámetros de las tareas mediante una serie de comparaciones estadísticas.

La visión de grid computing promete proveer una nueva plataforma que soporte una variedad de aplicaciones empresariales innovadoras. Para que esto se convierta en realidad, se necesita el diseño servicios de asignación de recursos e información adecuados. En este contexto, entender las características de los sistemas de grid computing reales es un paso crítico para mejorar la eficiencia económica de los servicios de grid existentes; es necesaria la creación de un método que permita aislar los problemas de eficiencia, en esta investigación se define ese método con base en un modelo económico-matemático. Para lograr esta meta, en esta investigación se analizan distintas cargas de trabajo de sistemas de grid computing reales, que se encuentran entre las instalaciones más grandes que están en producción actualmente. Se enfoca en el análisis del impacto económico de la administración del grid y en las características individuales de los trabajos, se intenta cuantificar la eficiencia de todo el sistema del grid como una forma de medir y mejorar el desempeño de los sistemas de grid computing.

Esta investigación también se analizan los algoritmos de calendarización como parte de la infraestructura de administración del grid; se incluye un análisis de la administración de la red y de los datos, la recolección de la información de ejecución de los trabajos y de las transferencias de datos a través de la red.

En esta investigación se define el modelo económico-matemático con base en el estudio y análisis de las siguientes áreas principales: administración económica (IGE), eficiencia de la infraestructura (IEI) y eficiencia operacional (IGO). El análisis de cada área está soportado por varios indicadores que son explicados en detalle en la sección de metodología.

Los recursos del “grid” están distribuidos en distintos lugares geográficos y su disponibilidad y políticas de costos varían dependiendo de las prioridades, tiempo de uso, metas del negocio y procedimientos administrativos. La administración de recursos, en dichos ambientes, es una tarea compleja. Esta tesis propone un algoritmo matemático y económico como una forma efectiva de medir el desempeño económico de los recursos computacionales. Además, propone un mecanismo (IBECOD) para construir un ranking de los sistemas de grid computing, con el fin de evaluar la eficiencia por medio de los siguientes criterios: análisis de la gestión económica (IGE), de la eficiencia en infraestructura (IEI) y de la gestión operacional del grid computing (IGO).

La evaluación económica de la eficiencia de los sistemas de grid computing –que es sujeto de estudio- es una herramienta que será utilizada por los dueños de los grid para evaluar distintos aspectos de sus sistemas actuales; esta evaluación los ayuda a identificar las partes del grid que necesitan ser mejoradas para que la eficiencia de todo el sistema sea óptima. Esta tesis define el índice IBECOD el cual resume la eficiencia global del grid, por medio de la medición de 25 indicadores de tres áreas: administración económica, eficiencia en infraestructura y eficiencia operacional.

Esta investigación está basada en una metodología descriptiva e explicativa. Se utilizaron distintos instrumentos como medios y herramientas de recolección de datos: entrevistas, encuestas, revisión bibliográfica, análisis de bitácoras de los sistemas, simulaciones y análisis económico.

Dada la falta de información disponible para realizar el análisis, esta investigación se enfoca en un periodo de tres años basado en el Grid Workloads Archive. En esta investigación se discute la aplicabilidad del modelo diseñado en tres áreas del grid computing: modelos económicos, algoritmos de calendarización y benchmarking.

Finalmente, se concluye que el diseño de la metodología de investigación permite la implementación y creación de un modelo económico-matemático que evalúa la eficiencia de grid computing, este modelo fue probado como un método válido para ayudar a los administradores en la identificación de los elementos que necesitan ser mejorados en la infraestructura, esto con el fin de obtener el mejor retorno económico del sistema de grid computing.

CARTA DE PROPIEDAD INTELECTUAL

XX de noviembre del 2006

Señores
Unidad de Postgrado
Universidad Latinoamericana de Ciencia y Tecnología
ULACIT
San José

DECLARACIÓN JURADA

Yo, Eduardo Castro Martínez alumno de la Universidad Latinoamericana de Ciencia y Tecnología (ULACIT), declaro bajo la fe de juramento y consciente de la responsabilidad penal de este acto, que soy el autor intelectual de la Tesis de Doctorado titulada: LA EFICIENCIA DE LOS ALGORITMOS DE DISTRIBUCIÓN DE CARGA EN LA INFORMÁTICA DISTRIBUIDA (GRID COMPUTING), por lo que libero a la ULACIT, de cualquier responsabilidad en caso de que mi declaración sea falsa.

Brindada en San José – Costa Rica en el día XX del mes de noviembre del año dos mil seis.

Con las muestras de mi más alta consideración y estima, me suscribo.

Cordialmente,

EDUARDO CASTRO MARTÍNEZ
CÉDULA No. 8-0068-0066

CARTA DE REVISIÓN FILOLÓGICA

Aserrí, 12 de noviembre de 2008

Doctora

Sara Cordero

Decana de la Facultad de Ciencias Económicas

Universidad Latinoamericana de Ciencia y Tecnología

Respetada doctora:

Yo, Margarita Cascante Chinchilla, en mi condición de Bachiller en Filología Española e incorporada al Colegio de Licenciados y Profesores en Letras, Filosofía, Ciencias y Artes hago constar que realicé la revisión filológica a la **TESIS DE DOCTORADO de EDUARDO CASTRO MARTÍNEZ**, titulada **“DISEÑO DE UN MODELO MATEMÁTICO QUE DETERMINE LA EFICIENCIA DE LOS ALGORITMOS DE DISTRIBUCIÓN DE CARGA EN LA INFORMÁTICA DISTRIBUIDA (GRID COMPUTING) APLICADA A LOS SECTORES ECONÓMICOS Y EMPRESAS”** y fueron aplicadas todas las correcciones al respecto.

Margarita Cascante Chinchilla

Carné de colegiada y cédula: 1-446-236

Teléfonos: 2230-4961

8893-8265

Índice General

I. INTRODUCCIÓN	12
II. PLANTEAMIENTO DEL PROBLEMA	22
A. INTRODUCCIÓN	22
B. FINES PRIORITARIOS DE LA INVESTIGACIÓN	23
1. <i>Propósitos primarios de la investigación</i>	25
C. DEFINICIÓN DEL PROBLEMA	26
D. OBJETIVO GENERAL Y ESPECÍFICOS DE LA INVESTIGACIÓN	31
1. <i>Objetivo general</i>	31
2. <i>Objetivos específicos</i>	31
E. HIPÓTESIS	32
1. <i>Relación entre objetivos e hipótesis</i>	32
III. MARCO TEÓRICO	36
A. INTRODUCCIÓN	36
B. LA INFORMÁTICA DISTRIBUIDA	36
1. <i>Ventajas de Grid Computing</i>	55
2. <i>Clasificación de los modelos de informática distribuida</i>	60
3. <i>Aspectos que deben tomar en cuenta los gerentes de informática</i>	68
C. COMPARACIÓN ENTRE EL GRID ELÉCTRICO Y EL GRID COMPUTACIONAL.....	71
D. COMPONENTES DE GRID COMPUTING	79
E. ORIGEN DE LOS MODELOS ECONÓMICOS EN LA INFORMÁTICA DISTRIBUIDA.	82
1. <i>Uso de los modelos económicos en la informática distribuida</i>	86
2. <i>Participantes en una economía de informática distribuida</i>	87
F. MODELO ECONÓMICO PARA LA ADMINISTRACIÓN DE RECURSOS	88
G. ESTADO DEL ARTE	92
H. MODELOS ECONÓMICOS APLICADOS EN PROBLEMAS COMPUTACIONALES	102

1.	<i>Características que debe poseer un modelo económico aplicado a la informática distribuida</i>	103
2.	<i>Modelos de determinación de precios</i>	105
3.	<i>Modelos económicos aplicados a la informática distribuida</i>	107
I.	NEGOCIOS DE GRID COMPUTING EN LOS ESTADOS UNIDOS DE AMÉRICA.....	118
1.	<i>Introducción</i>	118
2.	<i>Iniciativas de grid computing</i>	118
3.	<i>Sistemas de administración de recursos</i>	123
4.	<i>Comentarios sobre los sistemas existentes</i>	134
5.	<i>El impacto económico de la informática distribuida</i>	135
6.	<i>Impacto en las empresas</i>	145
J.	NEGOCIOS DE LA GRID COMPUTING	155
IV.	METODOLOGÍA	164
A.	INTRODUCCIÓN	164
B.	MATRIZ BÁSICA DE INVESTIGACIÓN.....	165
C.	MATRIZ DE OPERACIONALIZACIÓN DE ÍNDICES	167
1.	<i>Índices dependientes</i>	168
2.	<i>Índices Independientes</i>	168
3.	<i>Índice de Gestión Económica (IGE)</i>	170
4.	<i>Índice de Gestión Operacional (IGO)</i>	172
5.	<i>Índice de Eficiencia en Infraestructura (IEI)</i>	174
6.	<i>Matriz de Operacionalización</i>	177
D.	TIPO DE INVESTIGACIÓN	179
E.	UNIVERSO DE ESTUDIO	183
F.	DETERMINACIÓN DE LA MUESTRA	184
G.	FUENTES DE INFORMACIÓN.....	185
H.	SUJETO DE ESTUDIO	189
I.	TÉCNICAS DE OBTENCIÓN DE LA INFORMACIÓN	190
1.	<i>Recopilación documental</i>	190

2.	<i>Revisión estadística</i>	191
3.	<i>Entrevistas y consulta a expertos</i>	191
4.	<i>Simulación</i>	193
5.	<i>Discusiones y apreciaciones</i>	193
J.	ESTADO DE LA CUESTIÓN	194
K.	ALCANCES DEL ESTUDIO.....	198
L.	LIMITACIONES EXPERIMENTADAS EN EL ESTUDIO	199
M.	TIPOS DE ANÁLISIS PRACTICADOS	200
N.	DESCRIPCIÓN DE METODOLOGÍA PARA LOS ANÁLISIS PRACTICADOS	201
1.	<i>Representación histórica de datos</i>	201
2.	<i>Metodología para el análisis de indicadores</i>	202
3.	<i>Metodología para los análisis realizados utilizando simulación</i>	204
O.	CRONOGRAMA DE INVESTIGACIÓN.....	207
V.	CONSTRUCCIÓN DE UN MODELO PARA DEFINIR LA EFICIENCIA EN INFRAESTRUCTURA DE LOS ALGORITMOS DE CALENDARIZACIÓN EN INFORMÁTICA DISTRIBUIDA	208
A.	INTRODUCCIÓN.....	208
B.	FACTORES BÁSICOS DEL TRANSPORTE Y PROCESAMIENTO DE INFORMACIÓN	208
C.	DEFINICIÓN DEL MODELO MATEMÁTICO	209
VI.	DESCRIPCIÓN DE LA HERRAMIENTA DE SIMULACIÓN DE CALENDARIZACIONES EN LA INFORMÁTICA DISTRIBUIDA	224
A.	INTRODUCCIÓN.....	224
B.	DESCRIPCIÓN DE GRIDSIM.....	224
C.	SIMULACIÓN ECONÓMICA DEL AGENTE INTERMEDIARIO.....	226
1.	<i>La arquitectura del agente intermediario</i>	227
D.	SISTEMA DE ADMINISTRACIÓN DE RECURSOS Y CALENDARIZACIÓN.....	231
VII.	ANÁLISIS DE RESULTADOS.....	238

A.	INTRODUCCIÓN	238
B.	ÍNDICE DE GESTIÓN ECONÓMICA (IGE).....	239
1.	<i>SDSC Blue Horizon (San Diego Supercomputer Center)</i>	240
2.	<i>Sistema OSC Linux Cluster (Ohio Supercomputer Center)</i>	252
3.	<i>Cálculo del Índice de Gestión Económica (IGE)</i>	262
C.	ÍNDICE DE GESTIÓN OPERACIONAL (IGO)	264
1.	<i>Introducción</i>	264
2.	<i>SDSC Blue Horizon (San Diego Supercomputer Center)</i>	266
3.	<i>Sistema OSC Linux Cluster (Ohio Supercomputer Center)</i>	280
4.	<i>Cálculo del Índice de Gestión Operacional (IGO)</i>	293
D.	ÍNDICE DE EFICIENCIA EN INFRAESTRUCTURA (IEI).....	295
1.	<i>Eficiencia de procesamiento y la importancia del tiempo de estadía de la información en los nodos del grid</i>	295
2.	<i>Modelo para definir la eficiencia de los algoritmos de calendarización en informática distribuida</i>	296
3.	<i>Análisis de sensibilidad con cargas de información ilimitada.</i>	304
4.	<i>Diseño de un conjunto de nodos de procesamiento y su relación con la velocidad de procesamiento y administración de carga</i>	305
5.	<i>Análisis adicionales para el índice de eficiencia en infraestructura</i>	311
6.	<i>Cálculo del Índice de Eficiencia en Infraestructura</i>	313
7.	<i>Cálculo del Índice Básico de Eficiencia de Computación Distribuida (IBECOD)</i> 314	
E.	COMPROBACIÓN DE HIPÓTESIS	316
VIII.	CONCLUSIONES	322
IX.	RECOMENDACIONES	330
X.	SUGERENCIAS.....	336
XI.	ANEXO I	340
A.	INTRODUCCIÓN	340

B. INDICADOR DE EFICIENCIA EN INFRAESTRUCTURA (IEI)	340
XII. ANEXO II.....	350
XIII. GLOSARIO	360
XIV. BIBLIOGRAFÍA	402

Índice de Figuras

FIGURA N° 1 LA INFORMÁTICA DISTRIBUIDA PERMITE UTILIZAR RECURSOS GEOGRÁFICAMENTE DISPERSOS....	14
FIGURA N° 2 DISTRIBUCIÓN DE CARGA ENTRE LOS DISTINTOS COMPUTADORES	16
FIGURA N° 3 VISTA GENERAL DE UN SISTEMA DE INFORMÁTICA DISTRIBUIDA	42
FIGURA N° 4 LOS ADMINISTRADORES PUEDEN AJUSTAR LOS RECURSOS DE HARDWARE PARA ATENDER LAS CARGAS DE INFORMACIÓN.....	45
FIGURA N° 5 GRID COMPUTING COMO PARTE DE LA EVOLUCIÓN DE LA TECNOLOGÍA.....	47
FIGURA N° 6 OLAS DE EVOLUCIÓN DE LA TECNOLOGÍA.....	48
FIGURA N° 7 EVOLUCIÓN DE LA COMPUTACIÓN Y LAS COMUNICACIONES	49
FIGURA N° 8 MAPA DE LA RED ABILENE	ERROR! BOOKMARK NOT DEFINED.
FIGURA N° 9 PUNTOS DE CONEXIÓN DE LA RED ABILENE.....	50
FIGURA N° 10 TOPOLOGÍA DE LA UNITED KINGDOM NATIONAL BACKBONE RESEARCH AND EDUCATION NETWORK	51
FIGURA N° 11 EUROPEAN BACKBONE RESEARCH NETWORK GEANT.....	52
FIGURA N° 12 APAN, ASIAN NETWORK	53
FIGURA N° 13 REDES INTERNACIONALES DE GRID COMPUTING	53
FIGURA N° 14 GLOBAL TERABIT RESEARCH NETWORK.....	54
FIGURA N° 15 VISTA FÍSICA DEL GLOBAL TERABIT RESEARCH NETWORK	54
FIGURA N° 16 CARACTERÍSTICAS REDUNDANTES DE LA INFORMÁTICA DISTRIBUIDA	59
FIGURA N° 17 EVOLUCIÓN DE LA TECNOLOGÍA DE INFORMÁTICA DISTRIBUIDA.....	62
FIGURA N° 18 INFRAESTRUCTURA COMPUTACIONAL ESTÁNDAR	63
FIGURA N° 19 IMPLEMENTACIÓN LOCAL DE UN GRID	63
FIGURA N° 20 IMPLEMENTACIÓN REMOTA DE UN GRID	64
FIGURA N° 21 EJEMPLO DE GRID CORPORATIVO SIMPLE.....	65
FIGURA N° 22 LOS GRIDS CON SOCIOS PUEDEN COMUNICAR DOS O MÁS EMPRESAS	65
FIGURA N° 23 GRID GEOGRÁFICAMENTE DISTRIBUIDO	66

FIGURA N° 24 RELACIÓN DE GRID CORPORATIVO (INTRAGRID), GRID CON SOCIOS (EXTRAGRID) Y GRID DE SERVICIOS (INTERGRID).....	67
FIGURA N° 25 EJEMPLO DE INTERGRID QUE COMBINA LOS RECURSOS DE VARIAS CORPORACIONES.....	68
FIGURA N° 26 MAPA DE LA RED ELÉCTRICA DE ESTADOS UNIDOS DE AMÉRICA.....	72
FIGURA N° 27 SISTEMAS DE TRANSMISIÓN DE RED ELÉCTRICA	73
FIGURA N° 28 COMPONENTES PRINCIPALES DE UNA INFRAESTRUCTURA DE GRID COMPUTING	80
FIGURA N° 29 COMPONENTES DE LA INFRAESTRUCTURA ADMINISTRATIVA DEL GRID COMPUTING	81
FIGURA N° 30. PARTICIPANTES EN UN MERCADO DE INFORMÁTICA DISTRIBUIDA	88
FIGURA N° 31 ELEMENTOS DE UN CALENDARIZADOR DE PROCESOS Y RECURSOS	124
FIGURA N° 32 PLATAFORMA AVAKI.....	126
FIGURA N° 33 PLATAFORMA DE DATASYNAPSE	127
FIGURA N° 34 PLATAFORMA BASADA EN ENTROPIA	129
FIGURA N° 35 INFRAESTRUCTURA BASADA EN PLAFORM COMPUTING	131
FIGURA N° 36 INFRAESTRUCTURA BASADA EN UNITED DEVICES.....	132
FIGURA N° 37 PRIMERA FASE DE ADOPCIÓN DE LA TECNOLOGÍA.....	148
FIGURA N° 38 SEGUNDA FASE DE ADOPCIÓN DE LA TECNOLOGÍA.....	149
FIGURA N° 39 ARQUITECTURA GENERAL DE SIMGRID.....	225
FIGURA N° 40 ARQUITECTURA GENERAL DE UN GRID DE DATOS.....	226
FIGURA N° 41 COMPONENTES DE GRIDSIM.....	228
FIGURA N° 42 ARQUITECTURA DE NIMROD/G.....	233
FIGURA N° 43 INTERACCIÓN DE LOS COMPONENTES DE NIMROD/G.....	234
FIGURA N° 44 IMAGEN DE UN MAINFRAME HONEYWELL-BULL DPS 7 MAINFRAME BWW	384

Índice de tablas

TABLA N° 1 DESARROLLO ECONÓMICO TRADICIONAL Y COLABORATIVO	55
TABLA N° 2 SEGMENTACIÓN DEL MERCADO DE INFORMÁTICA DISTRIBUIDA SEGÚN TIPO DE RECURSO	60
TABLA N° 3 SEGMENTACIÓN DEL MERCADO DE INFORMÁTICA DISTRIBUIDA SEGÚN TIPO DE ORGANIZACIÓN...	61
TABLA N° 4 COMPARACIÓN ENTRE LA RED DE TRANSMISIÓN ELÉCTRICA Y LA INFORMÁTICA DISTRIBUIDA. FUENTE: CHETTY Y BUYYA (2002)	75
TABLA N° 5 PRINCIPALES PROYECTOS DE GRID COMPUTING	92
TABLA N° 6 CONSORCIOS DE GRID COMPUTING	94
TABLA N° 7 INICIATIVAS CIENTÍFICAS DE GRID COMPUTING – APLICACIONES	94
TABLA N° 8 INICIATIVAS CIENTÍFICAS DE GRID COMPUTING –SISTEMAS Y APLICACIONES P2P.....	96
TABLA N° 9 INICIATIVAS CIENTÍFICAS DE GRID COMPUTING – GRID DE DATOS	96

TABLA N° 10 SISTEMAS INTERMEDIARIOS DE GRID COMPUTING	97
TABLA N° 11 SISTEMAS DE GRID	97
TABLA N° 12 CALENDARIZADORES DE GRID	98
TABLA N° 13 PORTALES DE GRID	99
TABLA N° 14 AMBIENTES DE PROGRAMACIÓN GRID	99
TABLA N° 15 HERRAMIENTAS DE MONITOREO Y DE PREDICCIÓN	100
TABLA N° 16 COMPANÍAS COMERCIALES DE GRID Y P2P	101
TABLA N° 17 SISTEMAS DE COMPUTACIÓN DISTRIBUIDA QUE UTILIZAN MODELOS ECONÓMICOS	120
TABLA N° 18 PROPÓSITO E IMPACTO EN EL NEGOCIO DE LA INFORMÁTICA DISTRIBUIDA	145
TABLA N° 19 COLAS DEL SISTEMA SDSC BLUE HORIZON	187
TABLA N° 20 COMPOSICIÓN DEL OSC LINUX CLUSTER	188
TABLA N° 21 PARÁMETROS DEL MODELO DE MEDICIÓN DE EFICIENCIA EN INFRAESTRUCTURA	297

Índice de gráficos

GRÁFICO N° 1 DISTRIBUCIÓN DEL PODER COMPUTACIONAL MUNDIAL.....	119
GRÁFICO N° 2 GANANCIAS EN EFICIENCIA GRACIAS A LA IMPLEMENTACIÓN DE GRID COMPUTING.....	137
GRÁFICO N° 3 CAMBIO PROYECTADO EN LAS GANANCIAS NETAS DE LAS INDUSTRIAS EN EL AÑO 2010.....	138
GRÁFICO N° 4 CAMBIOS EN LOS INGRESOS, PRODUCTIVIDAD, EMPLEO Y PRECIOS GRACIAS A LA ADOPCIÓN DE LA INFORMÁTICA DISTRIBUIDA, SEGÚN INDUSTRIA	140
GRÁFICO N° 5 CAMBIO DE PRODUCTIVIDAD LABORAL EN LA INDUSTRIA FARMACÉUTICA	140
GRÁFICO N° 6 CAMBIO EN PRECIOS EN LA INDUSTRIA FARMACÉUTICA	141
GRÁFICO N° 7 CAMBIOS EN LAS GANANCIAS NETAS DE LAS INDUSTRIAS DE CAROLINA DE NORTE GRACIAS A LAS INVERSIONES EN INFORMÁTICA DISTRIBUIDA	143
GRÁFICO N° 8 GANANCIAS EN EMPLEO GRACIAS A LAS INVERSIONES EN INFORMÁTICA DISTRIBUIDA	144
GRÁFICO N° 9 GRIDS Y SERVICIOS DISTRIBUIDOS EN LA INDUSTRIA FARMACEÚTICA	151
GRÁFICO N° 10 GRIDS Y SERVICIOS DISTRIBUIDOS EN LA INDUSTRIA AUTOMOVILÍSTICA	151
GRÁFICO N° 11 GRIDS Y SERVICIOS DISTRIBUIDOS EN LA INDUSTRIA DE SERVICIOS FINANCIEROS.....	152
GRÁFICO N° 12 PARTICIPANTES SEGÚN INDUSTRIA.....	155
GRÁFICO N° 13 INGRESOS ANUALES DE LAS EMPRESAS PARTICIPANTES	156
GRÁFICO N° 14 CANTIDAD DE EMPLEADOS DE LAS EMPRESAS PARTICIPANTES.....	156
GRÁFICO N° 15 PRIORIDADES DE INVERSIÓN DE TI.....	158
GRÁFICO N° 16 PRORIDADES LOS EJECUTIVOS CON RESPECTO A LA INVERSIÓN EN INFORMÁTICA DISTRIBUIDA	159
GRÁFICO N° 17 INVERSIONES QUE VAN A TENER UN RETORNO SOBRE LA INVERSIÓN MÁS ALTO.....	160

GRÁFICO N° 18 PRINCIPALES BARRERAS EN LA IMPLEMENTACIÓN DE INFORMÁTICA DISTRIBUIDA	160
GRÁFICO N° 19 APLICACIONES MÁS IMPORTANTES QUE SE PUEDEN EJECUTAR EN UNA PLATAFORMA DE INFORMÁTICA DISTRIBUIDA.....	162
GRÁFICO N° 20 EJEMPLO DE EJECUCIÓN DE NIMROD/G.	236
GRÁFICO N° 21 COMPORTAMIENTO DE LOS INGRESOS DE SDSC BLUE HORIZON	244
GRÁFICO N° 22 COMPORTAMIENTO DE LOS INGRESOS DE OSC LINUX CLUSTER	255
GRÁFICO N° 23 SDSC BLUE TIEMPO DE ESPERA PROMEDIO POR MES.....	268
GRÁFICO N° 24 SDSC BLUE TIEMPO GLOBAL DE ESPERA PROMEDIO POR MES	269
GRÁFICO N° 25 SDSC BLUE TIEMPO DE ESPERA POR DÍA DE LA SEMANA	270
GRÁFICO N° 26 SDSC BLUE TIEMPO DE ESPERA PROMEDIO POR DÍA DE LA SEMANA.....	271
GRÁFICO N° 27 SDSC BLUE TIEMPO DE DURACIÓN DE LAS TAREAS DENTRO DEL GRID	272
GRÁFICO N° 28 SDSC BLUE TIEMPO PROMEDIO DE DURACIÓN DE LAS TAREAS DENTRO DEL GRID	273
GRÁFICO N° 29 SDSC BLUE RELACIÓN ENTRE MEMORIA REQUERIDA Y TAREAS ATENDIDAS POR MES.....	275
GRÁFICO N° 30 SDSC BLUE CANTIDAD DE TAREAS ATENDIDAS POR MES.....	276
GRÁFICO N° 31 SDSC BLUE PORCENTAJE DE TAREAS RECIBIDAS POR DÍA Y PROCESADORES UTILIZADOS POR DÍA	277
GRÁFICO N° 32 SDSC BLUE CANTIDAD DE TRABAJOS ATENDIDOS SEGÚN HORA DEL DÍA	279
GRÁFICO N° 33 SDSC BLUE PROMEDIO DE CANTIDAD DE TRABAJOS ATENDIDOS SEGÚN HORA DEL DÍA.....	279
GRÁFICO N° 34 OSC LINUX CLUSTER TIEMPO DE ESPERA POR MES.....	281
GRÁFICO N° 35 OSC LINUX CLUSTER TIEMPO DE ESPERA PROMEDIO POR MES.....	282
GRÁFICO N° 36 OSC LINUX CLUSTER TIEMPO DE ESPERA POR DÍA DE LA SEMANA	283
GRÁFICO N° 37 OSC LINUX CLUSTER TIEMPO PROMEDIO DE ESPERA POR DÍA DE LA SEMANA.....	284
GRÁFICO N° 38 OSC LINUX CLUSTER TIEMPO DE LAS TAREAS DENTRO DEL GRID.....	286
GRÁFICO N° 39 OSC LINUX CLUSTER. TIEMPO PROMEDIO DE DURACIÓN DE LAS TAREAS DENTRO DEL GRID	286
GRÁFICO N° 40 OSC LINUX CLUSTER CANTIDAD DE TAREAS ATENDIDAS POR MES	288
GRÁFICO N° 41 OSC LINUX CLUSTER CANTIDAD PROMEDIO DE TAREAS ATENDIDAS POR MES.....	289
GRÁFICO N° 42 OSC LINUX CLUSTER PORCENTAJE DE TAREAS RECIBIDAS POR DÍA Y PROCESADORES UTILIZADOS POR DÍA.....	290
GRÁFICO N° 43 OSC LINUX CLUSTER CANTIDAD DE TRABAJOS RECIBIDOS POR HORA DEL DÍA	292
GRÁFICO N° 44 OSC LINUX CLUSTER CANTIDAD PROMEDIO DE TRABAJOS RECIBIDOS POR HORA DEL DÍA....	292
GRÁFICO N° 45 COMPORTAMIENTO DE LOS INGRESOS RELACIONADOS CON EL AUMENTO EN LA EFICIENCIA DE PROCESAMIENTO DE DATOS	303
GRÁFICO N° 46 RELACIÓN ENTRE LOS INGRESOS Y AUMENTO DE PROCESAMIENTO DE INFORMACIÓN	304
GRÁFICO N° 47 RELACIÓN ENTRE CANTIDAD DE NODOS DISPONIBLES Y LA CANTIDAD DE INFORMACIÓN ATENDIDA	309

GRÁFICO N° 48 RELACIÓN ENTRE EFICIENCIA DEL GRID Y CAPACIDAD DE PROCESAMIENTO	310
GRÁFICO N° 49 RELACIÓN DE LA EFICIENCIA DE PROCESAMIENTO Y LA INFORMACIÓN ATENDIDA.....	311

Índice de cuadros

CUADRO N° 1 IMPACTO DE LA INFORMÁTICA DISTRIBUIDA EN LA INDUSTRIA	139
CUADRO N° 2 CAMBIOS EN LOS INGRESOS SEGÚN INDUSTRIA QUE ADOpte INICIALMENTE LA TECNOLOGÍA...	139
CUADRO N° 3 CAMBIOS EN LOS INGRESOS SEGÚN INDUSTRIA	142
CUADRO N° 4 CAMBIOS EN LOS INGRESOS SEGÚN INDUSTRIA	143
CUADRO N° 5 INVERSIONES EN INFORMÁTICA DISTRIBUIDA EN MILLONES DE US\$.....	154
CUADRO N° 6 PLANTEAMIENTO DE HIPÓTESIS DE TRABAJO SEGÚN LOS OBJETIVOS DE INVESTIGACIÓN	166
CUADRO N° 7 ÍNDICE DE GESTIÓN ECONÓMICA (IGE).....	171
CUADRO N° 8 ÍNDICE DE GESTIÓN OPERACIONAL (IGO)	172
CUADRO N° 9 ÍNDICE DE EFICIENCIA EN INFRAESTRUCTURA (IEI).....	174
CUADRO N° 10. RELACIÓN ENTRE HIPÓTESIS, ÍNDICES E INDICADORES.....	178
CUADRO N° 11 INVESTIGACIONES RELACIONADAS CON EL TEMA DE ESTUDIO	195
CUADRO N° 12 MUESTREO ALEATORIO DE TAREAS DE PROCESAMIENTO	206
CUADRO N° 13 COMPOSICIÓN DEL ÍNDICE DE GESTIÓN ECONÓMICA (IGE).....	239
CUADRO N° 14 INGRESOS DE SDSC BLUE HORIZON.....	241
CUADRO N° 15. RESUMEN DE INGRESOS DE SDSC BLUE HORIZON	242
CUADRO N° 16 EVOLUCIÓN DE LOS INGRESOS DE SDSC BLUE HORIZON	243
CUADRO N° 17 PORCENTAJE DE TAREAS NO ATENDIDAS EN SDSC BLUE HORIZON.....	245
CUADRO N° 18 TAREAS NO ATENDIDAS EN SDSC BLUE HORIZON	246
CUADRO N° 19 SDSC BLUE TIEMPO DE TRANSPORTE MODELO TIEMPO COMPARTIDO	248
CUADRO N° 20 SDSC BLUE TIEMPO DE TRANSPORTE MODELO ESPACIO COMPARTIDO	248
CUADRO N° 21 SDSC BLUE TIEMPO DURACIÓN MODELO TIEMPO COMPARTIDO	249
CUADRO N° 22 SDSC BLUE TIEMPO DE DURACIÓN MODELO ESPACIO COMPARTIDO	249
CUADRO N° 23 SDSC BLUE COMPARACIÓN DE COSTOS USUARIO	250
CUADRO N° 24 SDSC BLUE COMPARACIÓN DE COSTOS TAREAS PERSONALIZADAS	251
CUADRO N° 25 INGRESOS DE OSC LINUX CLUSTER.....	253
CUADRO N° 26 EVOLUCIÓN DE LOS INGRESOS DE OSC LINUX CLUSTER	254
CUADRO N° 27 TAREAS ATENDIDAS EN OSC LINUX CLUSTER.....	256
CUADRO N° 28 OSC LINUX CLUSTER TIEMPO DE TRANSPORTE MODELO TIEMPO COMPARTIDO	258
CUADRO N° 29 OSC LINUX CLUSTER TIEMPO DE TRANSPORTE MODELO ESPACIO COMPARTIDO.....	258

CUADRO N° 30 OSC LINUX CLUSTER TIEMPO DE DURACIÓN MODELO ESPACIO COMPARTIDO.....	259
CUADRO N° 31 OSC LINUX CLUSTER TIEMPO DE DURACIÓN MODELO ESPACIO COMPARTIDO.....	259
CUADRO N° 32 OSC LINUX CLUSTER COMPARACIÓN DE COSTOS.....	260
CUADRO N° 33 OSC LINUX CLUSTER COMPARACIÓN DE COSTOS TAREAS PERSONALIZADAS.....	261
CUADRO N° 34 CÁLCULO DEL ÍNDICE DE GESTIÓN ECONÓMICA.....	263
CUADRO N° 35 ÍNDICE DE GESTIÓN OPERACIONAL (IGO)	265
CUADRO N° 36 SDSC BLUE TIEMPO DE ESPERA PROMEDIO POR MES	267
CUADRO N° 37 SDSC BLUE TIEMPO DE ESPERA POR DÍA DE LA SEMANA	269
CUADRO N° 38 SDSC BLUE DURACIÓN DE LAS TAREAS DENTRO DEL GRID (EN SEGUNDOS)	272
CUADRO N° 39 SDSC BLUE CANTIDAD DE MEMORIA Y TRABAJOS POR MES.....	274
CUADRO N° 40 SDSC BLUE CANTIDAD DE TRABAJOS RECIBIDOS POR DÍA	277
CUADRO N° 41 SDSC BLUE CANTIDAD DE TRABAJOS ATENDIDOS SEGÚN HORA DEL DÍA.....	278
CUADRO N° 42 OSC LINUX CLUSTER TIEMPO DE ESPERA POR MES	280
CUADRO N° 43 OSC LINUX CLUSTER TIEMPO DE ESPERA POR DÍA DE LA SEMANA.....	283
CUADRO N° 44 OSC LINUX CLUSTER TIEMPO DE EJECUCIÓN DE LAS TAREAS	285
CUADRO N° 45 OSC LINUX CLUSTER TAREAS ATENDIDAS POR MES	287
CUADRO N° 46 OSC LINUX CLUSTER CANTIDAD DE TRABAJOS ATENDIDOS POR DÍA	290
CUADRO N° 47 OSC LINUX CLUSTER CANTIDAD DE TRABAJOS ATENDIDOS SEGÚN HORA DEL DÍA.....	291
CUADRO N° 48 CÁLCULO DEL ÍNDICE DE GESTIÓN OPERACIONAL	294
CUADRO N° 49 INFORMACIÓN BASE DEL MODELO	299
CUADRO N° 50 SIMULACIÓN 1: VELOCIDAD DE MANIPULACIÓN DE CARGAS DE DATOS VARIABLE	302
CUADRO N° 51 SIMULACIÓN: COMPARACIÓN DE INFORMACIÓN PROCESADA	306
CUADRO N° 52 SIMULACIÓN: RELACIÓN ENTRE LA CANTIDAD DE NODOS Y LA VELOCIDAD DE PROCESAMIENTO	307
CUADRO N° 53 SIMULACIÓN: RELACIÓN ENTRE LA CANTIDAD DE NODOS Y LA VELOCIDAD DE PROCESAMIENTO	307
CUADRO N° 54 RESULTADO DE SIMULACIÓN Y ANÁLISIS DE CANTIDAD DE NODOS.....	308
CUADRO N° 55 CÁLCULO DEL ÍNDICE DE EFICIENCIA EN INFRAESTRUCTURA IEI.....	313
CUADRO N° 56 DETALLES DEL CÁLCULO DEL IEI	313
CUADRO N° 57 ÍNDICE DE EFICIENCIA EN INFRAESTRUCTURA NORMALIZADO	314
CUADRO 58 CÁLCULO DE IBECOD.....	315

I. Introducción

“La infraestructura de grid computing proveerá a las empresas la habilidad de enlazar dinámicamente recursos como un todo para soportar la ejecución de aplicaciones distribuidas corporativas e intensivas en el uso de recursos.”

Fran Berman, 2003, página 9.

Desde el momento en que se desarrollaron las redes de computadoras y estas se conectan entre sí, surge la idea de unir la potencia ociosa de cada una para resolver problemas, los cuales únicamente podrían ser resueltos por las supercomputadoras de organizaciones gubernamentales, universidades o corporaciones multinacionales. Fuentes A., Huedo E., Montero S. Y Llorente M (2004) sobre este tema explican:

La investigación en computación de alto rendimiento ha estado orientada durante las últimas décadas al diseño de arquitecturas avanzadas que permitan resolver los problemas de gran desafío (intensivos en uso de procesador o en manejo de datos); al desarrollo de sistemas operativos, lenguajes de programación o extensiones de lenguajes existentes que permitan explotar estos sistemas; y a la optimización de aplicaciones y algoritmos para que aprovechen de forma más eficiente estas arquitecturas. Tradicionalmente la investigación realizada ha seguido siempre un modelo centralizado orientado a un único sistema que presta servicios de alto rendimiento. Sin embargo, a mediados de los años 90¹ comenzaron a popularizarse otras alternativas distribuidas que consiguen rendimientos comparables a los proporcionados por las arquitecturas más avanzadas a un precio más razonable. Estas tendencias de computación

¹ De Roure, David (2002). *The evolution of the grid*. University of Southampton, Reino Unido, 2002.

en red (Network Computing) consisten básicamente en interconectar sistemas distribuidos para aprovechar sus recursos, principalmente potencia de cálculo. (Página 24)

Conforme evolucionó la tecnología y surgió la red Internet, nace la idea de utilizar el potencial de todos los computadores conectados a red. Éste fue el principio de la tecnología de la informática distribuida o grid computing², la cual aprovecha los ciclos de procesamiento que no están siendo utilizados por los computadores.

Con respecto a los recursos no utilizados, Fuentes A., Huedo E., Montero S. y Llorente M (2004) opinan que

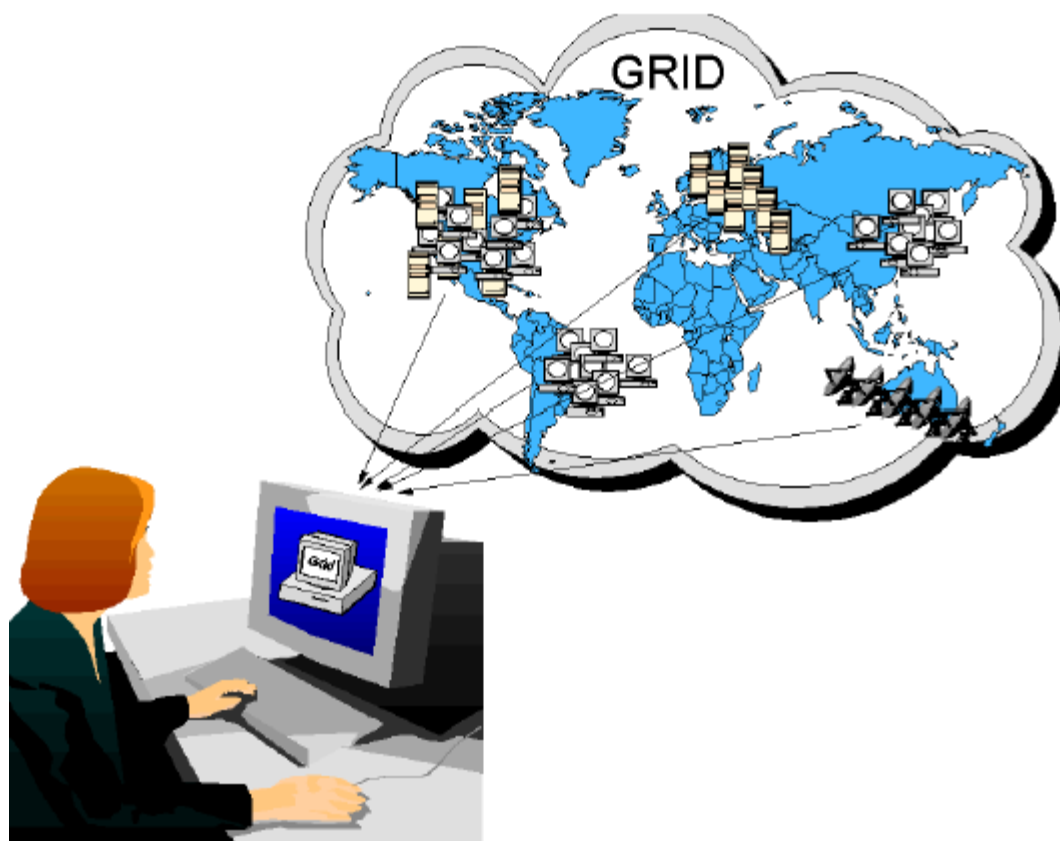
...la necesidad de aprovechar los recursos disponibles en los sistemas informáticos conectados a Internet y simplificar su utilización ha dado lugar a una nueva forma de tecnología de la información conocida como Grid Computing. Esta tecnología es análoga a las redes de suministro eléctrico: la idea es ofrecer un único punto de acceso a un conjunto de recursos distribuidos geográficamente (supercomputadores, clusters, almacenamiento, fuentes de información, instrumentos, personal). De este modo, los sistemas distribuidos se pueden emplear como un único sistema virtual en aplicaciones intensivas en datos o con gran demanda computacional”

El mecanismo básico de la informática distribuida incluye una aplicación que se instala en los computadores, ésta inicia su ejecución cuando se encuentra en estado inactivo. En ese preciso momento, empieza a ejecutar las tareas de procesamiento de información que le

² Fuentes A., Huedo E., Montero S. Y Llorente M (2004). Impacto de la interconexión en la planificación de trabajos en Grids. Boletín de RedIris, número 68-69, setiembre 2004.

fueron asignadas desde un controlador central³. En la Figura N° 1, se observa que existen recursos computacionales distribuidos en distintas partes del mundo y gracias a las facilidades tecnológicas de redes de alta velocidad y mecanismos seguros de comunicación, es posible que desde una consola central se puedan administrar.

Figura N° 1 La informática distribuida permite utilizar recursos geográficamente dispersos



Fuente: IBM Corporation, Introduction to Grid Computing with Globus.

³ Ferreira Luis, Berstis Viktors, et al. (2003). *Introduction to Grid Computing with Globus*. IBM Corporation 2003.

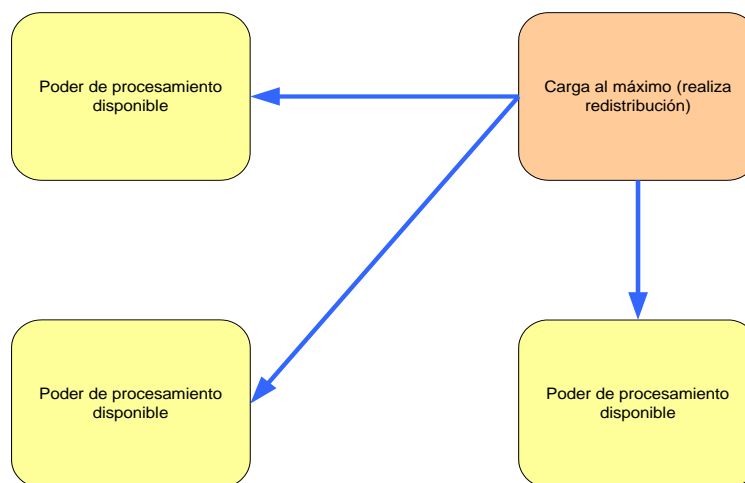
Desde 1990, la madurez de Internet y tecnologías asociadas en conjunto con la disponibilidad de computadoras potentes enlazadas por redes interconectadas, han logrado que la computación distribuida alcance niveles globales. Esto ha apoyado el surgimiento de grids computacionales de redes P2P⁴ (Peer-to-Peer) que permiten compartir recursos distribuidos.

Los grids se han enfocado en la unión de los recursos de las computadoras para resolver problemas de investigación tales como simulaciones, diseño de medicamentos, modelos ecológicos y otros; en cambio, la comunidad de P2P lo ha hecho con el fin de compartir música, por ejemplo Napster, Kazaa y AudioGalaxy⁵.

Uno de los mecanismos primordiales de la informática distribuida consiste en balancear la carga de trabajo entre distintos nodos o computadoras, tal y como se observa en la Figura N° 2: cuando la carga de trabajo de un nodo aumenta, los trabajos de procesamiento son desviados hacia los nodos que tienen recursos de procesamiento disponibles.

⁴ Peer-To-Peer, es una red informática entre iguales (en inglés peer-to-peer -que se traduciría de par a par- o de punto a punto, y más conocida como p2p), no tiene clientes ni servidores fijos, sino una serie de nodos o computadoras que se comporten simultáneamente como clientes y como servidores de los demás nodos de la red, con el fin de compartir archivos e información. Fuente: <http://es.wikipedia.org>, Recuperado el 29 de julio del 2006.

⁵ Buyya Rajkumar, Stockinger Heinz, Giddy Jonathan, Abramson David. (2004). *Economic Models for Management of Resources in Peer-to-Peer and Grid Computing*. Monash University, Melbourne, Australia y CERN, European Organization for Nuclear Research. 2004.

Figura N° 2 Distribución de carga entre los distintos computadores

Fuente: Elaboración propia.

Los impulsores de la tecnología de grid computing proceden de las investigaciones científicas, especialmente de la biología y la física, ya que necesitan computadoras de gran capacidad para analizar y procesar la información obtenida durante los procesos de investigación. Entre los proyectos de grid computing más importantes se destaca SETI@Home, (Search of Extraterrestrial Intelligence), en el cual millones de usuarios de todo el mundo aportan sus ciclos de procesamiento inutilizados para buscar signos de vida extraterrestre, esta búsqueda se realiza por medio del análisis de las señales procedentes del espacio.

Otros grandes proyectos de grid computing, distribuidos alrededor del mundo, abarcan temas tan diversos como el análisis de ADN, la lucha contra el cáncer o el sida, el estudio del genoma humano, el cambio climático, la inteligencia artificial y la criptografía, entre otros.

La organización que está participando en el desarrollo de esta tecnología es el Globus Project⁶, con sede Laboratorio Nacional Argonne y con el apoyo de la Universidad del Instituto de Ciencias de la Información de California del Sur (EE UU). En este proyecto se han desarrollado las técnicas básicas para la creación de grids (redes de computadoras). Su mayor aporte ha sido el Globus Toolkit⁷: consiste de una serie de componentes de software que pueden ser utilizados para el desarrollo y administración de grids.

Palazón (2003) opina que “a pesar de que todavía no es un recurso demasiado explotado en el ámbito empresarial, Grid Computing posee enormes expectativas de crecimiento. Según Gartner Group, al menos la mitad de las 500 grandes empresas presentes en el ranking de la revista Fortune⁸ tendrán su propia red antes de 2008, una especie de tendido informático similar a la red eléctrica. El siguiente paso será la integración de estas redes internas en una Great Global Grid (GGG), que en opinión de algunos expertos terminará sustituyendo a la World Wide Web”.

Grid computing permite aprovechar los momentos de baja carga de todos los computadores conectados mediante una red de alta velocidad. A diferencia de un clúster computacional convencional⁹, una plataforma grid de alto rendimiento involucra un conjunto heterogéneo (diferentes sistemas operativos o hardware) de nodos con capacidad de procesamiento de

⁶ Más información en la página <http://www.globus.org>

⁷ El Globus Toolkit es un software que se ha convertido en un estándar para la creación de aplicaciones que utilizan la tecnología de informática distribuida. Más detalle puede encontrarse en la página oficial de la herramienta: <http://www.globus.org>.

⁸ El ranking de la “Revista Fortune” se publica anualmente e incluye un ranking de las 500 empresas más grandes en América. Este ranking puede ser consultado en http://money.cnn.com/magazines/fortune/fortune500/industries/Computers_Office_Equipment/1.html

⁹ Un cluster computacional convencional consiste en utilizar un conjunto de computadoras para que trabajen como si fuesen una sola gran computadora. En este tipo de configuración, todos los miembros del cluster deben poseer características similares -tales como: sistema operativo, tipo de procesador y sistema de almacenamiento en disco, entre otros.

información; los cuales pueden residir en diferentes ubicaciones geográficas, poseer una estructura básica diferente y utilizar diferentes políticas de administración.

Ian Foster and C. Kesselman (1999) comparan la tecnología grid con la red de electricidad, de forma tal que el usuario debe utilizar los recursos computacionales de la misma forma que la energía eléctrica: desde cualquier lugar, con un funcionamiento adecuado y un costo razonable. La comparación de la informática distribuida con los servicios de electricidad, teléfono e incluso el agua, es válida debido a que se pueden identificar los siguientes factores comunes:

- Se utilizan los recursos únicamente cuando se necesitan.
- El usuario o cliente emplea la cantidad de recursos necesarios conforme el mecanismo de oferta y demanda. Una característica importante es que el usuario no precisa conocer ningún detalle con respecto a la infraestructura interna, la cual da soporte al recurso utilizado.
- Se paga por la cantidad de recursos usados o consumidos.

La realidad actual está rodeada por la tecnología, son pocas las actividades humanas en las cuales ésta no se aplica. El desarrollo de los países está ligado al uso correcto de sus recursos tecnológicos y, entre éstos, el poder computacional resulta uno de los más valiosos.

Por tanto, no es de extrañar que los países desarrollados posean mayor cantidad de procesamiento de información; por ejemplo: Estados Unidos, Japón, Inglaterra, Australia y -más recientemente- China¹⁰. Estos países identifican el poder computacional como una herramienta esencial para mantener su ventaja competitiva, aspecto que no ha sido

¹⁰ Dongara Jack, 2004. *High Performance Computing, Clusters and Grids, Oh My!*. University of Tennessee and Oak Ridge National Laboratory, 2004.

establecido como una prioridad en los países en desarrollo y se ve reflejado en la poca o nula inversión para aumentar el poder computacional.

La mayoría de los países de Latinoamérica todavía se encuentran en la etapa de sanar sus finanzas públicas, eliminar la pobreza y el desempleo; por tanto, no han identificado el potencial comercial que significa poseer un gran poder computacional.

Se debe resaltar que en este siglo existe un nuevo esquema económico, el cual posee sus bases en el neoliberalismo, conocido como la liberalización de mercados o modelo de mercado global. Estos modelos se han promovido como la solución al subdesarrollo impulsado por las escuelas de pensamiento de la Universidad de Harvard¹¹.

Con el surgimiento de las redes globales, prácticamente el mundo está conectado y las barreras físicas son superadas y sustituidas por el mundo digital, en el cual no existen fronteras. Si un país posee un gran poder computacional, éste puede ser vendido a quienes no lo poseen; de esta manera, parte de los ingresos de un país pueden provenir del procesamiento de información de otros.

Se puede identificar el poder de procesamiento como uno de los nuevos productos por comercializar en el mercado global¹²; sin embargo, los enfoques tradicionales son muy costosos. Por ejemplo, para procesar toda la información recibida de sus investigaciones, La NASA cuenta con varias supercomputadoras, cuyo costo es de millones de dólares.

¹¹ Kenichi Omae (1995). *The evolving global economy: making sense of the new world order*. Boston, MA : Harvard Business School, 1995.

¹² Cohen, Robert B. (2003). *Grid Computing - Projected Impact on North Carolina's Economy and Broadband Use through 2010*. Rural Internet Access Authority Research Report, Setiembre 2003.

Como respuesta a la necesidad de contar con supercomputadoras, a finales de la década de los ochenta y principio de los noventa¹³, surge la idea de unir cientos o miles de computadoras personales de bajo costo para que funcionen como una sola, de forma tal que se cuenta con el poder de procesamiento de una supercomputadora pero con costos más bajos.

La aplicación de este modelo de procesamiento de información, conocido como informática distribuida, tiene sus orígenes en el área académica¹⁴. En las universidades se utilizó este poder computacional para procesar la información proveniente de investigaciones de distintas áreas tales como biología, astronomía, ciencias médicas, matemática, entre otros. Poco después, las empresas de tecnología como Oracle, IBM, Sun Microsystems y Hewlett-Packard¹⁵, comenzaron a comercializar esta tecnología como una solución a las necesidades de procesamiento de las empresas.

Debido a la naturaleza económica de este modelo de procesamiento de información, esta tesis se concentra en diseñar un modelo matemático que permita determinar la eficiencia de los algoritmos de carga en la informática distribuida, aplicada a los sectores económicos y empresariales.

En este ámbito, las señales del mercado indican que los gobiernos no deben de ver a la computación como un accesorio, sino como una fuente de ingresos que puede atender la demanda mundial de procesamiento de información.

La naturaleza global de la economía muestra que está regida por criterios comerciales; estos han transformado los servicios, la tecnología y la competencia y los insumos necesarios

¹³ De Roure, David (2002). *The evolution of the grid*. University of Southhampton, Reino Unido, 2002.

¹⁴ Idem

¹⁵ Scannell, Ed, Krill, Paul (2003). Oracle Bangs on Grid Drum. *InfoWorld*, 08 de setiembre del 2003, volumen 25, número 35, p16-17, 2p.

para crear un producto final. Además, han cambiado. En este ámbito cabe la informática distribuida, la cual se puede comercializar como un producto de alto valor agregado. Por tanto, una estrategia de desarrollo deberá incluir el poder computacional como un producto que puede ser comercializado a nivel mundial.

II. Planteamiento del problema

“La ubicuidad de Internet así como la disponibilidad de computadoras poderosas y tecnologías de redes de alta velocidad están cambiando rápidamente el horizonte computacional y la sociedad”

Rajkummar Buyya, Universidad de Melbourne, Australia

A. Introducción

Esta parte de la investigación muestra, de manera fundamental y concreta, la definición del problema objeto de estudio, así como los objetivos generales y específicos, los cuales son su guía y se complementan con los antecedentes teóricos de este trabajo.

En este capítulo se construyen y establecen los lineamientos que originan esta investigación: se explica el surgimiento de la informática distribuida y su evolución hasta convertirse en un producto por comercializarse a nivel mundial. En este sentido, es necesario comparar las experiencias existentes en este campo y aplicar, en ellas, un modelo que permita determinar su eficiencia.

Esta investigación pretende diseñar un modelo matemático que permita comparar las distintas implementaciones de informática distribuida, sin importar su naturaleza, ya sea comercial o académica.

Así, es necesario determinar qué elementos inciden en la economía del manejo de infraestructuras de informática distribuida y, más aún, cuál es el impacto que tiene éste en los beneficios económicos recibidos por las empresas dueñas de esta infraestructura. Se debe tomar en consideración que los distintos nodos de informática distribuida están

dispersos alrededor del mundo y su ubicación geográfica podría incidir en su eficiencia, porque si la infraestructura de conexión de red que poseen no es la apropiada, la información ingresada al nodo y los resultados obtenidos tardarían mucho en ser transportados desde su punto origen hasta su punto de destino.

Con base en los aspectos señalados, esta investigación pretende establecer un tema central: la eficiencia de la infraestructura de la informática distribuida de manera específica y su influencia en los procesos de comercialización de procesamiento de información.

La globalización de la economía hace necesario que las empresas establezcan políticas y estrategias, las cuales les permitan optimizar sus operaciones hasta ser competitivas. Por tanto, si se conocen los factores influyentes en la eficiencia de sus operaciones, será posible que esta competitividad sea incrementada y -de esta forma- la empresa puede brindar una respuesta moderna en el comercio internacional.

Este enfoque de optimización de las operaciones de la informática distribuida, será el principal aporte de esta investigación, el cual -de manera cuantitativa- podrá ser utilizado para medir la eficiencia del manejo de la infraestructura.

Adicionalmente, esta investigación incluye un apartado sobre el diagnóstico de la tecnología de informática distribuida en los Estados Unidos de América. En éste, se reúnen los aportes existentes hasta el momento sobre los modelos económicos aplicados a la informática distribuida, cuyo origen está principalmente en el área académica y comercial.

B. Fines prioritarios de la investigación

Esta investigación se enfocará en identificar elementos provenientes de los modelos económicos aplicados en la calendarización en computación distribuida. Estos elementos

ayudarán a establecer la base de un modelo de la evaluación de la eficiencia de los modelos económicos.

Se profundizará en el estudio de los modelos que han sido sugeridos como viables para su aplicación en un ambiente de informática distribuida. Para esto, se estudiarán los artículos científicos más recientes y relacionados con el tema; igualmente, se consultará a expertos del Departamento de Computación Distribuida de la Universidad de Melbourne en Australia.

Debido al gran número de publicaciones relacionadas con la computación distribuida, este estudio se ejecutará con base en las publicaciones durante un periodo definido (de 1990 hasta el 2006) y dichas publicaciones deben estar relacionadas con la aplicación de modelos económicos como entes regidores de un ambiente de grid computing.

En esta investigación, se analiza la eficiencia de los algoritmos de asignación de tareas y manejo de recursos en un ambiente de informática distribuida, de forma tal que se pueda establecer un modelo económico, que permita determinar cuál algoritmo es más eficiente en un ambiente dado.

Se propondrá un sistema de evaluación de eficiencia que incluya los siguientes factores:

- Cómo está siendo utilizada la infraestructura existente en las empresas que brindan servicios de grid computing y su aporte en la eficiencia.
- La gestión operacional de los sistemas de grid computing y su impacto en la eficiencia.
- Comportamiento de indicadores económicos de los sistemas de grid computing.

1. Propósitos primarios de la investigación

Dentro de los propósitos primarios de la presente investigación se incluyen los siguientes:

- Probar las hipótesis planteadas en esta investigación con las herramientas que brindan las técnicas y métodos de la teoría económica. Se utilizará la simulación con el fin de comparar distintos modelos de calendarización de tareas y la información base de estas simulaciones provendrá de los registros históricos de sistemas reales de informática distribuida. Además, se analizarán estos registros históricos con el fin de estudiar los siguientes índices:
 - Índice de Gestión Económica. Este ofrece información sobre las finanzas de las empresas que brindan servicios de grid computing. Con base en esta información, se puede determinar si la aplicación del modelo de operación de la informática distribuida tiene impacto en las finanzas de las empresas.
 - Índice de Gestión Operacional. Determina la eficiencia con que se están utilizando los recursos de hardware y software componentes del grid, su tiempo de operación y el de respuesta. De esta manera, se puede determinar la calidad de servicio que se está brindando: tiempo de atención y respuesta de los procesos enviados al grid, disponibilidad de los recursos para ser utilizados por los clientes, tiempo de espera para que la carga de información recibida sea atendida, entre otros.
 - El Índice de Eficiencia en Infraestructura (IEI) incluye -entre otros- indicadores tales como: manipulación de carga dentro del grid y tiempo de transmisión de los datos (desde el punto de origen hasta su destino dentro del grid de procesamiento). Ambos parámetros son de gran importancia en el transporte y procesamiento de datos y la operación del grid computing.
- Recolectar información empírica y formal mediante el uso de:

- Comunicaciones electrónicas con personales, relacionadas con los desarrollos más recientes de modelos económicos aplicados en la informática distribuida. Para ello se cuenta con la colaboración del Departamento de Informática Distribuida de la Universidad de Melbourne Australia y del Departamento de Ciencias Computacionales de la Universidad de Alicante.
- Consulta de publicaciones investigativas de expertos internacionales relacionadas directamente con el tema. Se utilizarán publicaciones realizadas por Rajkummar Buyya e Ian Foster, los cuales han sido pioneros en la aplicación de modelos económicos en la informática distribuida.
- Revisión de publicaciones de empresas multinacionales, las cuales son principales proveedores a escala mundial de informática distribuida. Entre ellas se incluyen a IBM Corporation, Sun Microsystems, Hewlett Packard y Oracle Corporation.
- Revisión de publicaciones especializadas de organismos, tales como el Institute of Electrical and Electronics Engineers (IEEE, por sus siglas en inglés) y la Association for Computing Machinery (ACM, por sus siglas en inglés). Se examinará la última década.
- Entregar al mundo económico y científico, un aporte que ayude a medir la eficiencia de los distintos modelos de distribución de carga en la informática distribuida.

C. Definición del problema

La informática distribuida (Grid Computing) es la próxima generación a nivel computacional y hace posible la selección y trabajo en conjunto de recursos heterogéneos-geográficamente distribuidos- que se utilizan para resolver problemas complejos de la ciencia, ingeniería y comercio.

La informática distribuida integra recursos y servicios heterogéneos. Estos son distribuidos por naturaleza; así, se pueden crear organizaciones virtuales que se extienden más allá de sus fronteras administrativas, pues une entidades de procesamiento de una empresa con aquellas que son proveídas por otras compañías o productores. En este ámbito, Minoli 2005 indica que durante la última década el gobierno de Estados Unidos ha tenido un papel principal en el desarrollo de proyectos de informática distribuida en la NASA, laboratorios nacionales, centros de supercomputadoras e instituciones académicas.

Minoli muestra, además, que la informática distribuida ha sido tema de interés de los matemáticos, los investigadores y los departamentos de computación; pero recientemente esta tecnología puede ser aplicada a las empresas del Fortune 500, las cuales desean reducir costos de ejecución de tareas. Por otra parte, posee el potencial de cambiar la forma como trabajan las empresas de la misma manera que lo hizo Internet.

Este tipo de informática puede ser descrita como una red de computación, la cual soporta el concepto de “utility computing” -o sea- que los usuarios pueden obtener potencia de computación por demanda, sin necesidad de poseer sus propios recursos de hardware. Incluye mecanismos y protocolos para coordinar los recursos compartidos utilizados para resolver problemas de procesamiento de información por medio de activos computacionales distribuidos alrededor del mundo. Estos incluyen computadoras personales, servidores, mainframes, supercomputadoras y dispositivos de almacenamiento.

Los recursos en un grid son heterogéneos y distribuidos por naturaleza. Por lo tanto, la disponibilidad y las políticas relacionadas con su uso y costo cambian dependiendo del usuario, hora, prioridades y metas¹⁶. La administración de los recursos -en estos ambientes- son distribuidos en una tarea compleja. Debido a esto se hace necesario un

¹⁶ Buyya Rajkumar (2002), *Economic-based Distributed Resource Management and Scheduling for Grid Computing*. School of computer science and software engineering Monash University, Melbourne Australia, 2002.

modelo para la administración de los recursos y la asignación de las tareas. Debe contemplar los aspectos relacionados con la comercialización de recursos, la calidad de servicio y la regularización de los recursos de oferta y demanda, de forma tal que se pueda dar una motivación para participar en el grid¹⁷.

Para solventar los problemas de administración de los recursos -geográficamente distribuidos- de manera tradicional se ha recurrido a enfoques basados en la centralización de políticas de uso y distribución¹⁸. Para ser ejecutadas de forma apropiada, necesitan información completa del estado de los recursos y de las necesidades de los consumidores, lo cual es muy difícil de obtener y conservar actualizado. Además, en este tipo de enfoques se necesita una política de administración de recursos común en todos los conjuntos de grids y en cada uno de sus componentes.

Una alternativa, ante estos enfoques, es descentralizar las políticas de empleo de recursos. Sin embargo, se debe llegar a un consenso global sobre la interpretación y aplicación de dichas políticas. Por tanto, el problema de administración y asignación de recursos continúa siendo compleja. Ferguson y Yemini (1996) plantean que es imposible definir una política estática de administración de recursos aplicable a todo el sistema grid.

R. Buyya y D. Abramson (2000) sugieren el uso de un paradigma basado en la economía para administrar la asignación de recursos en un entorno de informática distribuida. Este enfoque provee las bases para la administración exitosa de recursos en un ambiente de recursos distribuidos. Los autores sostienen que los modelos económicos proveen algoritmos, políticas y herramientas para regir los recursos compartidos en un sistema de

¹⁷ Buyya Rajkumar, S. Chapin, and D. DiNucci. (2000). *Architectural Models for Resource Management in the Grid*, First IEEE/ACM International Workshop on Grid Computing (GRID 2000), Springer Verlag LNCS Series, Germany, Dec. 17, 2000, Bangalore, India.

¹⁸ Ian Foster and C. Kesselman. (1999). *The Grid: Blueprint for a Future Computing Infrastructure*, Morgan Kaufmann Publishers, USA, 1999.

informática distribuida. Pueden ser creados con base en precios o intercambios (trueque). En el primer caso, los recursos tienen un precio establecido con base en la demanda, oferta y el valor de los recursos. En el segundo caso, todos los participantes poseen recursos y los intercambian con otros.

Adicionalmente, estos autores establecen que un sistema de administración de recursos, cuya base sea la economía, debe proveer un modelo, el cual permita llevarse a cabo las metas de los proveedores de servicios (dueños de los recursos dentro del grid) y de los consumidores. En este modelo se debe incluir sus necesidades (demanda) y un mecanismo de asignación o calendarización de recursos, por medio del cual se creen estrategias que permitan escoger los proveedores adecuados para suplir las demandas de los consumidores. Igualmente, se hace necesario establecer parámetros dentro del modelo que admitan crear políticas de establecimiento de precios; de esa forma se incrementa la utilización del sistema y permite a los proveedores de servicios ser más competitivos. En un modelo perfecto se requiere que el mercado sea competitivo, para lo cual resultan necesarios mecanismos que ayuden a mantener el precio en equilibrio -o sea- un punto tal que la oferta de un recurso sea igual a su demanda.

Con base en los planteamientos de estos autores, se puede determinar que en la teoría económica existen diferentes modelos del intercambio de bienes y servicios. Estos pueden ser adaptados para que se posibilite regular los bienes y servicios presentes en grid computing.

Una vez que estos modelos han sido adaptados es necesario establecer una forma de medir su eficiencia, de forma tal, que se pueda comparar la de uno u otro. Por tanto, la tarea principal de esta tesis es proponer un modelo matemático para evaluar la eficiencia de los modelos de operación, administración y asignación de recursos aplicados a la informática distribuida (grid computing).

Determinar la eficiencia de un modelo de administración establecido es fundamental para que una empresa, la cual brinda servicios de procesamiento de información, pueda reducir sus costos y mejorar la productividad. Muchos de los nodos del grid podrían perder oportunidades decisivas de intensificar la competitividad de la exportación de servicios de procesamiento. Por esto las empresas deben asegurarse de que sus nodos de procesamiento funcionen como entidades comerciales reales, cuyo propósito fundamental sea la atención y procesamiento eficiente de cargas de información.

La manipulación de las cargas de información en los nodos debe tender a la eficiencia y, además, tomar en cuenta que la globalización de las redes y de la información ha llevado a los nodos a formar parte de redes complejas de información, las cuales deben trabajar con eficiencia en función de los costos y contribuir a la competitividad de las empresas procesadoras de información. Esto implica que los procesadores de información deben poseer equipos e infraestructura moderna que permitan optimizar los movimientos de cargas de información, acceso adecuado a las redes de transporte y sistemas eficientes de distribución de tareas de procesamiento dentro de sus nodos.

Con base en los factores citados anteriormente, se plantea el diseño de un modelo matemático que determine la eficiencia de los algoritmos de distribución de carga en la informática distribuida (grid computing) aplicada a los sectores económicos y empresas.

Para iniciar este planteamiento, a continuación se enumera las siguientes interrogantes:

1. ¿Existe modelo matemático que ayude a medir la eficiencia de la distribución de procesos y asignación de recursos en un ambiente de grid computing?
2. ¿Cuál es el impacto económico y en eficiencia que tienen los algoritmos de distribución de procesos y asignación de recursos en un ambiente de grid computing?

3. ¿Cuál es el estado de gestión operacional de los sistemas de grid computing y su impacto en la eficiencia?
4. ¿Cuál es el comportamiento de indicadores económicos de los sistemas de grid computing?

D. Objetivo general y específicos de la investigación

1. Objetivo general

El objetivo base de la presente investigación es:

Definir un modelo económico matemático que permita medir la eficiencia de la distribución de procesos y asignación de recursos en un ambiente de grid computing.

2. Objetivos específicos

Con el fin de lograr el objetivo general de esta investigación, se han planteado los siguientes objetivos específicos:

- 2.1 Analizar el impacto económico y en eficiencia que tienen los algoritmos de distribución de procesos y asignación de recursos en un ambiente de grid computing.
- 2.2 Analizar la forma como está siendo utilizada la infraestructura existente en las empresas que brindan servicios de grid computing y su aporte en la eficiencia.

2.3 Analizar la gestión operacional de los sistemas de grid computing con el fin de determinar su impacto en la eficiencia.

2.4 Analizar el comportamiento de indicadores económicos de los sistemas de grid computing comprendidos dentro esta investigación.

E. Hipótesis

La hipótesis general de esta investigación es la siguiente: La eficiencia de los sistemas de informática distribuida (IBECOD) puede determinarse mediante el análisis de la gestión económica (IGE), de la eficiencia en infraestructura (IEI) y de la gestión operacional (IGO) del grid computing.

Con base en esta hipótesis, esta investigación tratará de demostrar que los distintos modelos utilizados como mecanismos de calendarización de cargas de trabajos de la computación distribuida inciden en la eficiencia global del sistema.

1. Relación entre objetivos e hipótesis

Objetivos de la investigación	Hipótesis
<p>Objetivo General</p> <p>Definir un modelo económico matemático que permita medir la eficiencia de la distribución de procesos y asignación de recursos en un ambiente de grid computing.</p> <p>El objeto de estudio es el Índice</p>	<p>H_I</p> <p>La eficiencia de los sistemas de informática distribuida (IBECOD) puede determinarse mediante el análisis de la gestión económica (IGE), de la eficiencia en infraestructura (IEI) y de la gestión operacional (IGO) del grid computing. Además, los índices independientes tienen influencia</p>

Objetivos de la investigación	Hipótesis
<p>Básico de Eficiencia de Computación Distribuida (IBECOD), el cual resume la eficiencia de los grids de computadoras encargados del procesamiento de datos.</p> <p>Los índices independientes del estudio son: IGE (Índice de Gestión Económica), IEI (Índice de Eficiencia en Infraestructura) e IGO (Índice de Gestión Operacional).</p> <p>Los datos del estudio comprenden un periodo de cuatro años -del 2000 al 2003- considerando los elementos que permite el desarrollo de esta investigación.</p>	<p>significativa en el índice IBECOD.</p>
<p>Objetivos específicos</p> <ol style="list-style-type: none"> <li data-bbox="261 1367 837 1623">1. Analizar el impacto económico e impacto en eficiencia que tienen los algoritmos de distribución de procesos y asignación de recursos en un ambiente de grid computing <li data-bbox="261 1696 837 1843">2. Analizar la forma en la cual está siendo utilizada la infraestructura existente en las empresas que brindan servicios de 	

Objetivos de la investigación	Hipótesis
grid computing y su aporte en la eficiencia.	
3. Analizar la gestión operacional de los sistemas de grid computing con el fin de determinar su impacto en la eficiencia.	
4. Analizar el comportamiento de indicadores económicos de los sistemas de grid computing comprendidos en esta investigación.	

III. Marco teórico

“El concepto de Grid ha surgido en los últimos años para denominar un conjunto de recursos computacionales heterogéneos distribuidos pertenecientes a distintas organizaciones.”

Fernández, 2004

A. Introducción

En este capítulo se introducen los principales conceptos relacionados con la tecnología de la informática distribuida. Se inicia con un estudio sobre el origen de esta tecnología, así como la evolución que ha tenido hasta convertirse en una realidad, la cual está siendo aplicada tanto en el ámbito académico como comercial. Se analizan las ventajas del grid computing y cómo éste puede ser aplicado exitosamente en el ámbito corporativo con el fin de mejorar el uso de los recursos computacionales existentes. Además, se incluyen los principales aspectos por tomar en cuenta los gerentes de informática cuando se enfrentan a implementaciones que utilizan este tipo de tecnología.

Finalmente, se introducen los conceptos económicos aplicados para facilitar la administración de la informática distribuida, así como las características que debe poseer un modelo económico aplicado a la informática distribuida.

B. La informática distribuida

El desarrollo de la tecnología de grid computing ha sido el resultado de la influencia de distintas tecnologías que se unieron para crear un nuevo concepto. Para analizar su origen debemos remontarnos al inicio de los años ochenta¹⁹: en esta década se realizaron intensos progresos en la investigación, desarrollo de dispositivos de hardware, software y

¹⁹ De Roure, David (2002). *The evolution of the grid*. University of Southampton, Reino Unido, 2002.

aplicaciones de ejecución paralela -dos o más tareas ejecutándose al mismo tiempo. La necesidad de procesar la gran cantidad de información proveniente de los resultados de las investigaciones llevó a los estudiosos a desarrollar algoritmos, programas y arquitecturas que soportaran la ejecución simultánea de tareas. Sin embargo, a medida que la complejidad de los programas aumentaba, los recursos de las computadoras fueron superados, incluso aquellos que soportaban ejecución paralela. Por esto, los investigadores y creadores de software empezaron a explorar la posibilidad de distribuir tareas más allá de la máquina, de forma tal que se pudieran procesar cantidades de información cada vez más grandes.

En la siguiente década, la tecnología de ejecución paralela se enfocó en la creación de mecanismos poderosos y eficientes para administrar la comunicación entre procesadores dentro de una misma máquina; se incentivó la ejecución y desarrollo de arquitecturas de cálculo paralelo. Entre estos desarrollos se incluyen algoritmos tales como el Message Passing Interface (MPI). Se define así:

..un protocolo de comunicación entre computadoras. Es el estándar para la comunicación entre los nodos que ejecutan un programa en un sistema de memoria distribuida. Las implementaciones en MPI consisten en un conjunto de librerías de rutinas que pueden ser utilizadas en programas escritos en los lenguajes de programación tales como C++, Fortran y Ada. La ventaja de MPI sobre otras librerías de paso de mensajes, es que los programas que utilizan la librería son portables (dado que MPI ha sido implementado para casi toda arquitectura de memoria distribuida), y rápidos, (porque cada implementación de la librería ha sido optimizada para el hardware en la cual se ejecuta) (Wikipedia, 2006).

Otro de los algoritmos desarrollados fue la máquina virtual paralela que “es una librería para el cómputo paralelo en un sistema distribuido de computadoras. Está diseñado para permitir que una red de computadoras heterogénea comparta sus recursos de cómputo (como el procesador y la memoria RAM) con el fin de aprovechar esto para disminuir el tiempo de ejecución de un programa al distribuir la carga de trabajo en varias computadoras” (Wikipedia 2006).

El concepto fundamental, detrás de la informática distribuida, consiste en unir cientos o miles de computadoras para que formen una sola unidad de poder computacional, capaz de procesar gran cantidad de información. La principal ventaja de la informática distribuida consiste en que estas computadoras no deben estar en la misma ubicación geográfica, puede formar parte de este grid cualquier computador que tenga acceso a una conexión de red. Este grid puede llegar a ser tan potente como un supercomputador y recientemente empresas como Oracle, IBM, Hewlett-Packard y Sun Microsystems están comercializando esta tecnología. De esta forma, las empresas pueden enviar a estos grids cantidad de información para ser procesada y pagan únicamente por el poder computacional consumido, lo cual evita que inviertan en la adquisición de gran cantidad de hardware.

Al inicio se pensó que la tecnología de grid computing era un ensanchamiento de la computación paralela; sin embargo, en la práctica los grids se utilizaron como una plataforma de integración de recursos, ejecución de tareas y procesamiento de información independiente de la infraestructura de hardware y software.

En la década de los ochenta²⁰, investigadores de distintas disciplinas empezaron a trabajar sobre los grandes problemas de las ciencias y la ingeniería, lo cual requería una infraestructura computacional gigantesca; además, se desarrolló un modelo de trabajo

²⁰ De Roure, David (2002). *The evolution of the grid*. University of Southampton, Reino Unido, 2002.

multidisciplinario, que involucraba a científicos repartidos alrededor del mundo, quienes necesitaban distribuir información con sus colegas y a la vez procesarla.

En los años noventa, Estados Unidos desarrolló el proyecto US Gigabit²¹ que incluía la interconexión de redes metropolitanas distribuidas y la ejecución de aplicaciones descentralizadas. Esta red fue creada con la intención de investigar la viabilidad de conectar recursos distribuidos y utilizarlos para atacar los problemas de procesamiento que enfrentaban los científicos.

En 1995²², se desarrolló el proyecto I-WAY que incluyó la unión de 17 redes distribuidas geográficamente. En él, se desarrollaron alrededor de sesenta aplicaciones y lo que se considera la primera infraestructura de grid computing, la cual incluía software para administrar el acceso, la seguridad y la coordinación de recursos. Después de I-WAY se desplegaron proyectos tales como Globus, Condor y NetSolve²³, los cuales contribuyeron al desarrollo del grid computing y que serán estudiados más adelante.

El concepto fundamental detrás de la informática distribuida es combinar el poder de procesamiento de las computadoras²⁴. Por ejemplo, Minoli (2005) plantea que si una

²¹ Berman Fran, Fox Geoffrey y Hey Tony (2003). *The Grid. Past, present, future*. Editorial John Wiley & Sons, Ltd, 2003.

²² Idem

²³ Buyya Rajkumar, Stockinger Heinz, Giddy Jonathan, Abramson David. (2002). *Economic models for resource management and scheduling in Grid computing*. Monash University, Melbourne, Australia y CERN, European Organization for Nuclear Research. 2002.

²⁴ Minoli (2005) define la informática distribuida como “un ambiente virtual de computación distribuida. En este ambiente se habilita que en tiempo de ejecución se seleccione, comparta recursos autónomos que están geográficamente distribuidos, esta selección se realiza con base en disponibilidad, capacidad, desempeño y costo de estos recursos, y además se toma en cuenta las restricciones de plazo de ejecución de las tareas y del presupuesto disponible”, página 1. Esta definición permite algunos de los factores que serán considerados en esta investigación.

empresa cuenta con 10.000 computadoras, las cuales tienen un poder computacional de 333 MIPS (miles de instrucciones por segundo) aproximadamente, si se combina todo este poder se tendrá un computador virtual con poder de procesamiento de 3 terabytes (10^{12} bytes) de operaciones de punto flotante por segundo (TFLOPS, por sus siglas en inglés). Según estudios de IBM, mencionados por Minoli (2005), los servidores están ociosos el 40% del tiempo y la mayoría de los computadores personales no hacen ninguna operación de procesamiento pesada el 95% del tiempo; de esta forma, la informática distribuida es una forma de aprovechar ese tiempo ocioso de los computadores.

El procesamiento de grandes cantidades de información es una necesidad creciente. Según Minoli (2005), en el año 2006, la mayoría de los proyectos científicos producirán varios petabytes (un petabyte es 10^{15} bytes) de datos que deben ser procesados. Para tener un punto de comparación, considere una computadora que posee un disco duro de 100 gigabytes, un petabyte equivaldría a 10.000 de estas computadoras. Por lo tanto, la tecnología de grid computing surge como una solución para procesar esta gran cantidad de información y es accesible desde el punto de vista económico.

La aplicación inicial de la tecnología de informática distribuida se enfocó principalmente en el campo científico, en el cual se utilizó para resolver problemas que implicaban gran cantidad de cálculos. Sin embargo, conforme la tecnología ha evolucionado, ésta puede ser aplicada para aumentar el poder computacional de las empresas. Además, se puede utilizar el modelo de informática distribuida para administrar -de forma más eficiente- los recursos de hardware y de software. Por ejemplo, si una empresa cuenta con un número limitado de licencias de un software determinado, con el uso de la informática distribuida se pueden administrar dichos recursos, ya que las cargas de información enviadas a ese software pueden ser coordinadas de forma más eficiente y las licencias del software son utilizadas al máximo.

Cuando se piensa en la informática distribuida, la forma más natural de explicarla es comparándola con los servicios de electricidad²⁵: cuando una empresa necesita electricidad la consume de su proveedor, en unas ocasiones necesita más o menos electricidad, pero al final la empresa paga únicamente por la electricidad consumida. Este es el mismo concepto de la informática distribuida. Las empresas necesitan poder computacional para procesar su información y en lugar de comprar gran cantidad de infraestructura -bajo el concepto de informática distribuida- las empresas consumen poder computacional a través de la red y al final del consumo le pagan a los dueños de los recursos computacionales por la cantidad consumida.

Conforme evolucionó la tecnología y surgió la red Internet, surge la idea de utilizar el potencial de todos los computadores conectados a red. Éste fue el principio de la tecnología de la informática distribuida o grid computing, la cual aprovecha los ciclos de procesamiento que no están siendo utilizados por los computadores.

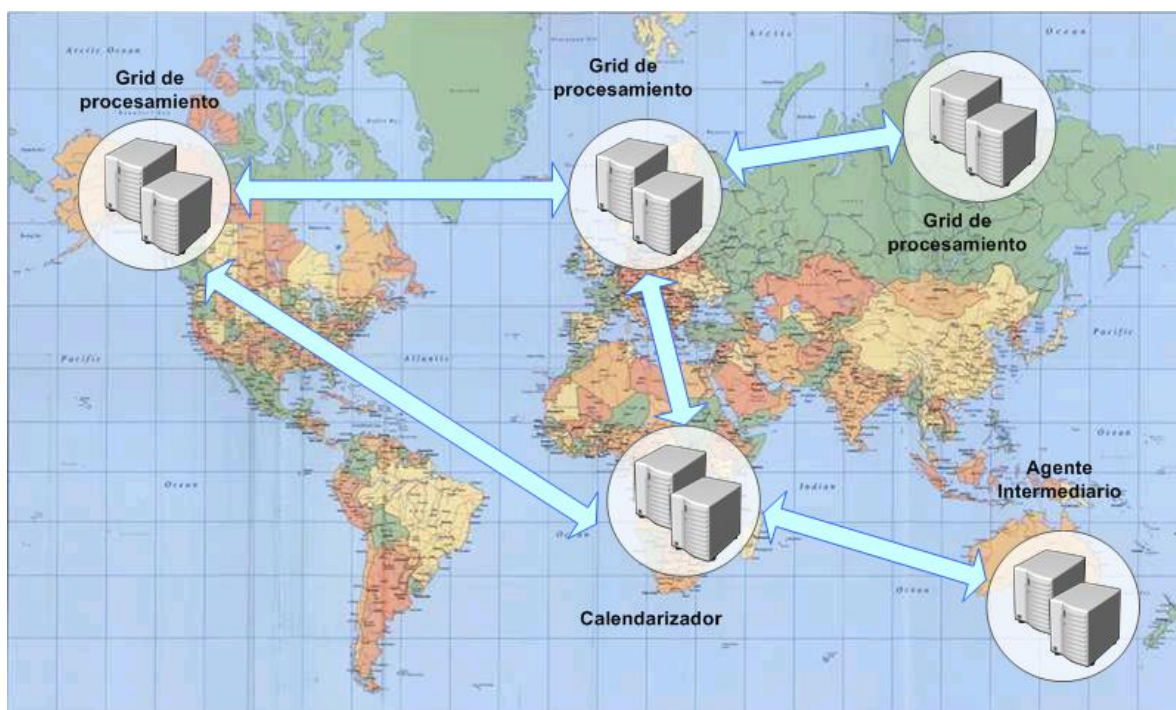
El mecanismo básico de la informática distribuida incluye una aplicación que se instala en los computadores y esta aplicación detecta cuando se encuentra en estado inactivo; en ese preciso momento empieza a ejecutar las tareas de procesamiento de información que le fueron asignadas desde un controlador central.

Los grids se han enfocado en la unión de los recursos de las computadoras para resolver problemas de investigación tales como simulaciones, diseño de medicamentos, modelos ecológicos, entre otros. En la Figura N° 3, se puede ver el esquema general de un sistema de informática distribuida, el cual contiene los siguientes componentes:

²⁵ Buyya Rajkumar (2002), *Economic-based Distributed Resource Management and Scheduling for Grid Computing*. School of computer science and software engineering Monash University, Melbourne Australia, 2002.

- Grid de procesamiento. Contiene un conjunto de computadoras enlazadas entre sí que se utilizan para procesar la información del usuario.
- Calendarizador. Se encarga de asignar tareas a los grids de procesamiento.
- Agente intermediario. Escoge el grid o los grids por utilizar; para esto toma como entrada los requerimientos especificados por el usuario, por ejemplo: límite de tiempo dentro del cual se deben obtener los resultados.

Figura N° 3 Vista general de un sistema de informática distribuida



Fuente: Elaboración propia.

Minoli (2005) ha rastreado el origen del concepto de informática distribuida hasta el año 1987 en el Reporte Especial de Bellcore SR-NPL-000790 en la sección “Network for a Computing Utility”. En este reporte, se define grid computing de la siguiente forma: “El servicio propuesto provee un mecanismo completo para hacer posible el concepto de grid

computing. Este incluye lo siguiente: (1) la red física sobre la cual la información puede viajar y la interface mediante la cual una estación de trabajo invitada puede participar en el aprovisionamiento de ciclos de máquina y mediante el cual los consumidores del servicio pueden enviar trabajos; (2) mecanismos de distribución de carga que se utilizan para invocar los servidores necesarios para completar un trabajo; (3) un mecanismo de seguridad confiable; (4) un mecanismo de contabilidad para invocar el sistema de facturación; (5) un directorio detallado de los servidores”

Según Zhang y Chung (2002), investigadores de IBM, la informática distribuida “es una colección de recursos computacionales distribuidos que están disponibles a través de una red local o global, y se muestran al usuario o aplicación como un único gran sistema computacional virtual. La visión es crear organizaciones virtuales dinámicas que comparten recursos entre personas e instituciones de forma coordinada. El grid computing es un enfoque de la computación distribuida que se expande no únicamente en cuanto a la localidad geográfica sino que también a las organizaciones, arquitectura de las computadoras y límites del software, para que se pueda proveer poder computacional ilimitado, colaboración y acceso a información para cualquiera que se conecte a grid. Internet se enfoca en hacer que las computadoras se comuniquen entre ellas, en cambio el grid computing hace que las computadoras trabajen juntas. La informática distribuida ayudará a convertir a Internet en una verdadera plataforma de computación, que combina la calidad de servicio que la computación empresarial con la habilidad de compartir recursos heterogéneos compartidos – todo desde aplicaciones, datos, almacenamiento y servidores”.

El Globus Alliance (2006), la cual es una organización cuyo objetivo principal es la aplicación práctica de los conceptos de grid en el campo científico y de la ingeniería, define al grid computing como: “una infraestructura que hace posible el uso integrado y colaborativo de computadoras de alto rendimiento, redes, bases de datos e instrumentos científicos que pertenecen y son administrados por múltiples organizaciones. Las aplicaciones de grid generalmente involucran gran cantidad de datos y de cálculos, y que

necesitan compartir recursos de forma segura a través de las fronteras de las organizaciones.”

Ian Foster y C. Kesselman (1999) conciben que un grid computacional es una infraestructura de hardware y software que provee acceso consistente, confiable y de bajo costo a instalaciones de alto poder computacional. Está relacionado con el uso coordinado y compartido de recursos y de resolución de problemas dentro de organizaciones virtuales dinámicas y multi-institucionales. El concepto principal en este modelo reside en la habilidad de negociar contratos, los cuales permitan compartir recursos entre un conjunto de participantes (productores y consumidores) y, de esa forma, se puedan utilizar dichos recursos para algún propósito determinado, que en la mayoría de los casos es el procesamiento de información.

Estos autores consideran que el compartir recursos no se limita a la transferencia de archivos, sino a tener acceso directo a computadoras, software, datos y otros recursos tecnológicos requeridos por estrategias de solución de problemas de ciencias y de ingeniería mediante colaboración. El uso compartido está regido por reglas que definen cuáles recursos están disponibles, quién los puede utilizar y bajo qué condiciones. Foster y Kesselman definen que el conjunto de personas u organizaciones que se rigen bajo esas reglas conforman organizaciones virtuales.

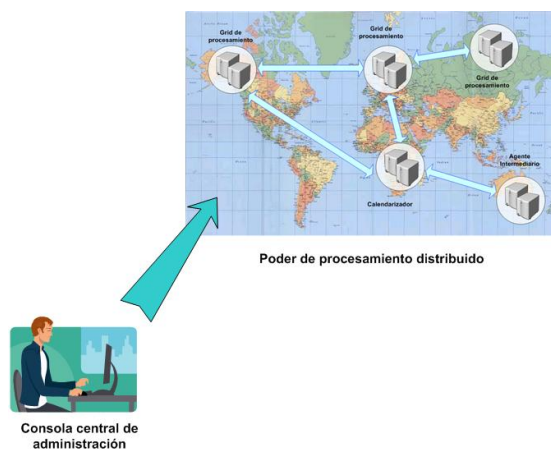
Los autores en mención indican que antes del desarrollo de la informática distribuida una aplicación de negocios tenía asignada una plataforma de hardware que incluía servidores y espacio de almacenamiento dedicado, dicha plataforma estaba restringida para una única aplicación. Debido a la naturaleza de este modelo de distribución de recursos, era prácticamente imposible que dos aplicaciones de negocios pudiesen compartir los recursos asignados. Un problema mayor se presenta debido a que es muy difícil predecir la carga de trabajo que va procesada en los servidores en distintos momentos del tiempo, ya que generalmente las aplicaciones presentan estacionalidad. Por ejemplo, los sistemas de

contabilidad necesitan gran cantidad de procesamiento al final del mes, cuando se realiza el cierre contable y durante el resto del tiempo la carga no es tan elevada, o sea: los recursos son subutilizados, lo cual implica una pérdida. Esta naturaleza estacional de las aplicaciones implica que se deben sobredimensionar los recursos de hardware, lo cual eleva los costos de implementación de los proyectos de tecnología de la información.

El desarrollo del grid computing permite que la inversión de tecnología no sea tan elevada, pues se invierte en la plataforma de hardware, la cual soporta la operación diaria y cuando se presentan periodos de alto consumo de procesamiento, es posible que se adquiera ese procesamiento de un proveedor. De esta forma, la empresa únicamente paga lo que consume y no invierte en activos de hardware subutilizados la mayoría del tiempo.

En la Figura N° 4, se muestra cómo los administradores de la infraestructura puede modificar las políticas de uso de los recursos de forma tal que atiendan las cargas de trabajo adecuadamente. Entre las empresas que ofrecen el servicio de computación por demanda, se encuentran Oracle Corporation, IBM y Sun Microsystems.

Figura N° 4 Los administradores pueden ajustar los recursos de hardware para atender las cargas de información



Fuente: Elaboración propia.

Con el uso de la informática distribuida, una organización puede integrar -de forma transparente- recursos distribuidos geográficamente. Estos recursos son de distinta naturaleza, su unión permite brindar servicios que soportan aplicaciones y procesamiento de datos de forma eficiente, sin necesidad de contar con supercomputadores, lo cual reduce los costos asociados. Según Sun Microsystems (2003), Transamerica Life Canada logró reducir hasta en un cincuenta por ciento sus costos de infraestructura y se mejoró la eficiencia operacional, todo gracias a la implementación de la tecnología de grid computing. En este mismo reporte se indica que las organizaciones que invirtieron en esa tecnología esperan un retorno de la inversión entre 36 y 117 por ciento gracias a la reducción de costos.

Bednarz y Dubie (2003) sostienen que el uso del grid computing está en aumento. Mencionan un estudio de Gartner, en el cual se indica que en año 2003 un 15% de las corporaciones adoptarán la tecnología de procesamiento distribuido y que el mercado para el grid computing pasará de \$8.6 billones de dólares estadounidenses hasta \$25 billones para el año 2007. Para esa época, un 30% de las empresas contarán con algún tipo de tecnología informática distribuida.

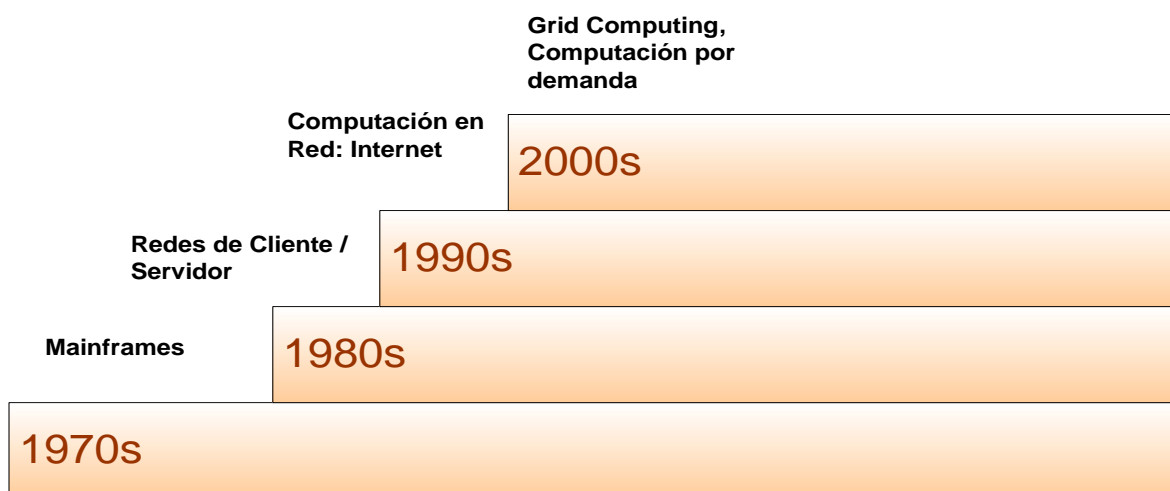
Minoli (2005) plantea que las empresas invierten en servicios de tecnologías de información aproximadamente un 6% de sus ganancias anuales. Por ejemplo, una empresa del Fortune 500 con ingresos anuales de 10 billones de dólares estadounidenses podría gastar hasta 600 millones de dólares anuales en tecnología de la información. El uso de la tecnología de grid computing se espera que provea ahorros de hasta un 15% en el periodo 2007-2008 y elevarse hasta en el 30% en los años 2010-2012.

El concepto de la informática distribuida puede aplicarse a las empresas y corporaciones. Específicamente, el Insight Research Corporation -en su informe Grid Computing: Vertical Market Perspective (2005)- indica que los análisis más recientes de tecnología de

información coinciden en que una de las tecnologías que está surgiendo es el Grid Computing y será la base de la cuarta ola de tecnología.

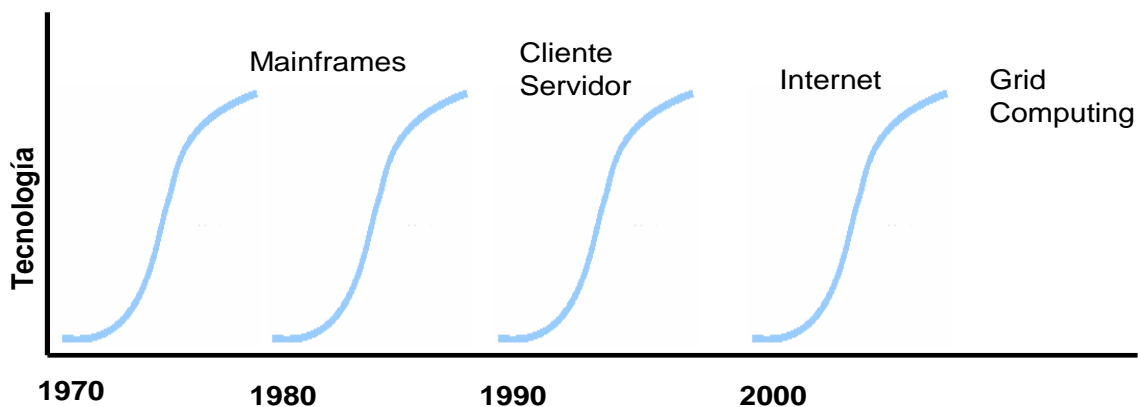
La evolución de la tecnología de información se puede apreciar en la Figura N° 5 y Figura N° 7. En la Figura N° 5, se puede observar que en los años 70s y 80s la tecnología de la computación se centraba en los mainframes o computadores centrales poderos. Posteriormente, durante de la década de los 90s, surgió el concepto de redes cliente servidor, en el cual los usuarios utilizaban computadores capaces de realizar procesamiento localmente y para acceder recursos compartidos, se comunicaban con una computadora central conocida como servidor. A partir del año 2000, el uso de las redes globales -tales como Internet- se generalizó y es gracias a los avances en interconexión mundial de redes que fue posible desarrollar y aplicar la tecnología de grid computing en la investigación y en el soporte de aplicaciones comerciales.

Figura N° 5 Grid Computing como parte de la evolución de la tecnología



Fuente: The Insight Research Corporation.

Figura N° 6 Olas de evolución de la tecnología

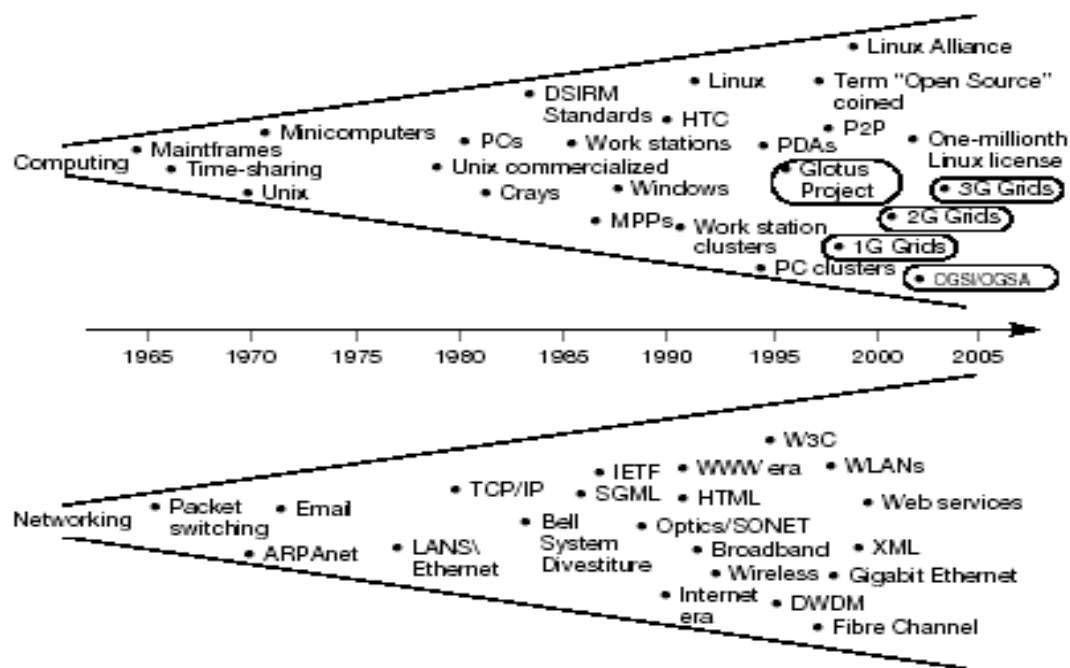


Fuente: Elaboración propia, con base en apreciaciones The Insight Research Corporation.

Según The Insight Research Corporation, la cuarta ola de la tecnología -ver Figura N° 6- incluye conceptos tales como grid computing, computación por demanda, computación utilitaria, tecnología de la información orgánica, computación adaptativa e Internet Computing.

En la Figura N° 7, se aprecia tanto la evolución de la computación como de las redes. La primera inicia con el desarrollo de los mainframes y los minicomputadores, a principios de los 80s se introduce el concepto de computador personas y a finales de la misma década se populariza el uso de sistemas operativos tales como Windows; alrededor de 1995, inicia el desarrollo de la tecnología de computación distribuida apoyada por proyectos tales como el Globus Project. Actualmente, existen iniciativas tales como el 3G Grids y el OGSA que permiten la implementación de grids computacionales.

Figura N° 7 Evolución de la computación y las comunicaciones

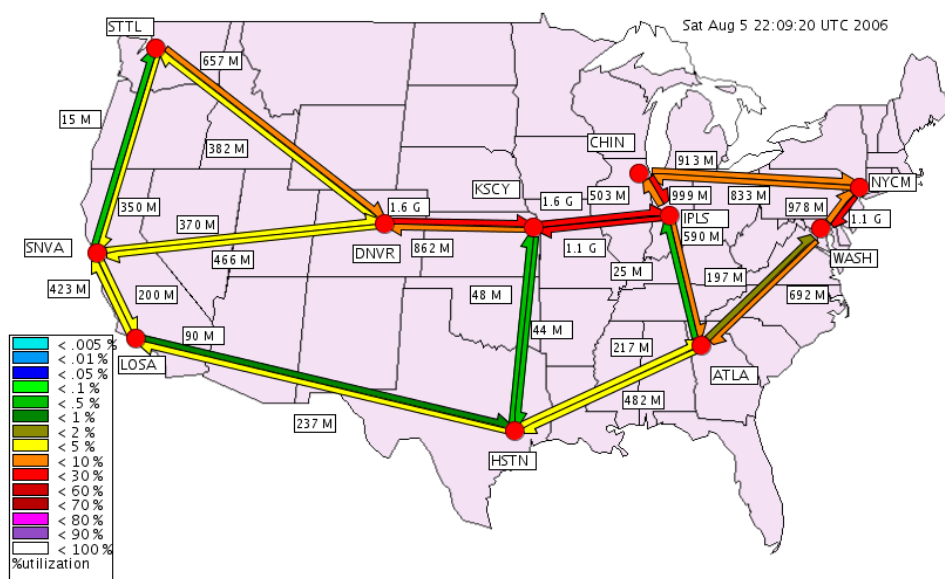


Fuente: Minoli (2005)

El paradigma de la informática distribuida permite que recursos de procesamiento heterogéneos de todos los tipos puedan ser compartidos a través de redes de alta velocidad, con la capacidad de ser reasignados dinámicamente a las aplicaciones, procesos y usuarios, dependiendo del comportamiento de la oferta y la demanda.

Según la Insight Research Corporation (2005), existen alrededor de 100 grids académicos o de investigación en el mundo. En la Figura N° 8 y Figura N° 9, se muestra la red Abilene desarrollada en los Estados Unidos en el año 2002, cuyo objetivo es impulsar el desarrollo científico en este país, por medio de una red distribuida.

Figura N° 8 Mapa de la Red Abilene



Fuente: http://weathermap.gnoc.iu.edu/abilene_jpg.html Fuente: Datos del 05 de agosto del 2006 a las 04:15pm -6GMT.

Figura N° 9 Puntos de conexión de la red Abilene

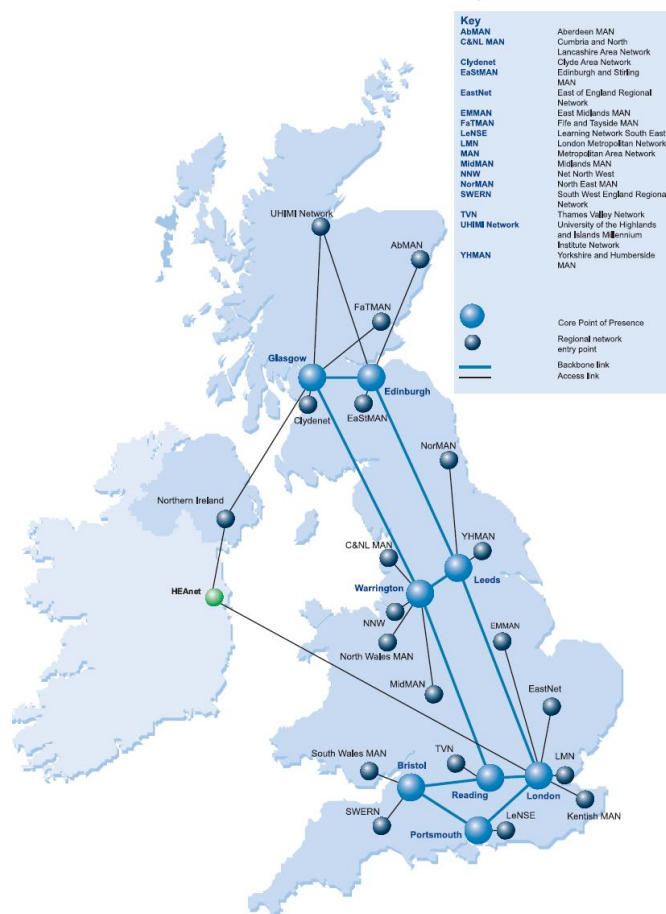


Fuente: <http://abilene.internet2.edu>

Igualmente, en Europa existen iniciativas en torno al grid computing. Por ejemplo, en la Figura N° 10 se muestra la red SuperJanet 4 desarrollada en el año 2002 como una forma

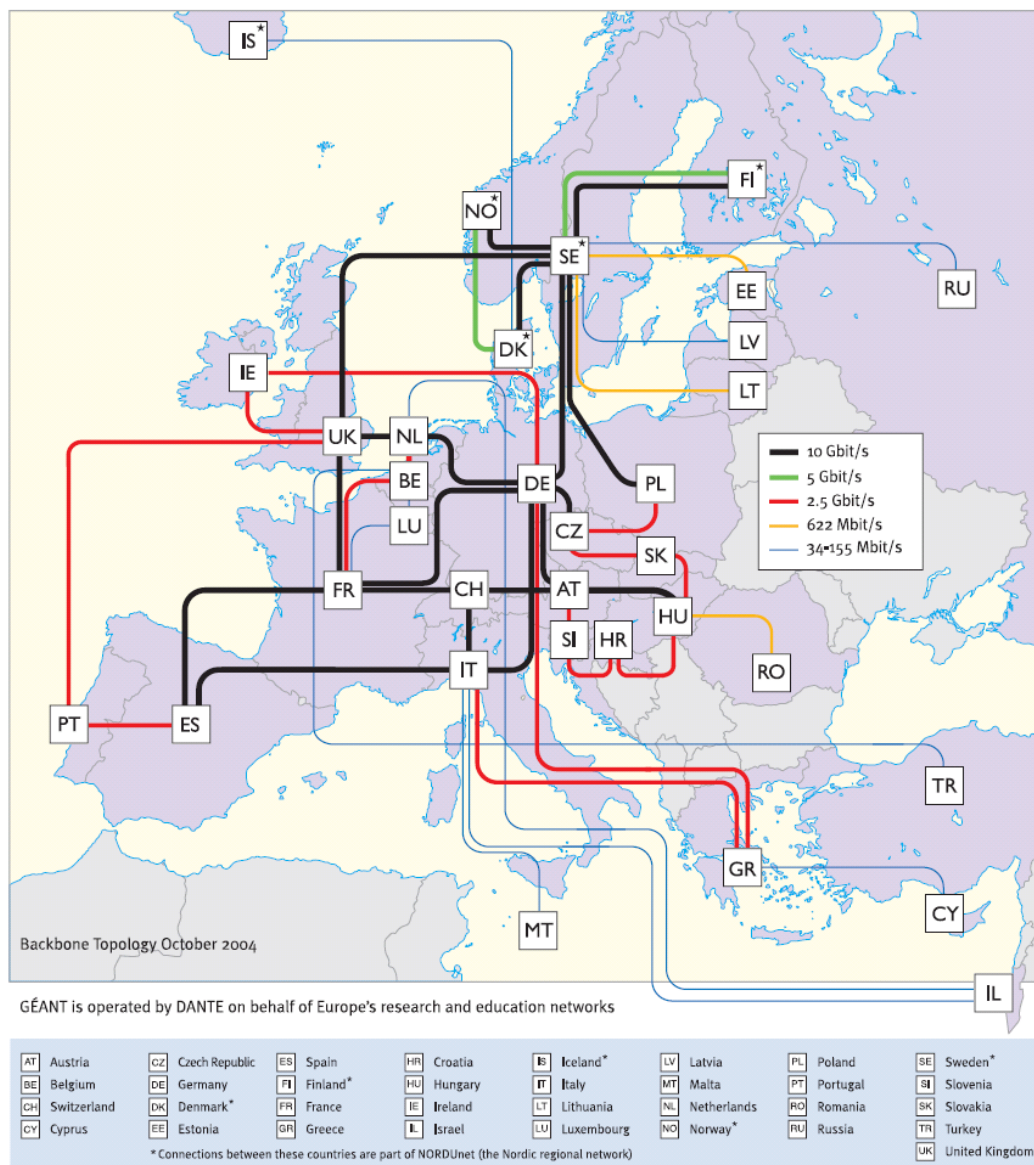
de impulsar el grid computing en esa región. También existen iniciativas a nivel europeo. Entre ellas resalta la European Backbone Research Network que incluye a varios países y se muestra en la Figura N° 11.

Figura N° 10 Topología de la United Kingdom National Backbone Research and Education Network



Fuente: <http://www.ja.net/about/topology/janetbackbonekey.pdf>

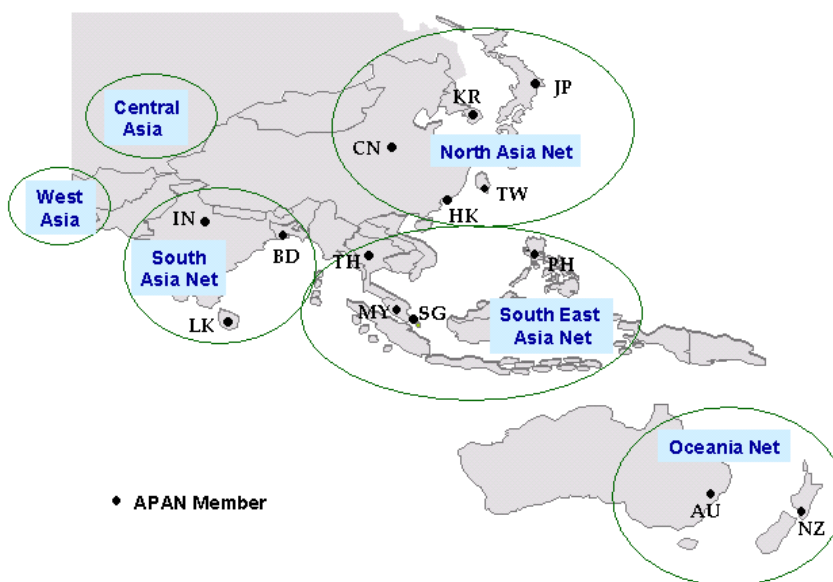
Figura N° 11 European Backbone Research Network GEANT



Fuente: Berman (2003)

Las iniciativas y desarrollos de grid computing han sobrepasado las fronteras y los continentes. Por ejemplo, en Asia se desarrolló la Asia Pacific Advanced Network (APAN)- Figura N° 12- y redes internacionales que unen países alrededor del mundo y se muestran en la Figura N° 13 y Figura N° 14.

Figura N° 12 APAN, Asian Network



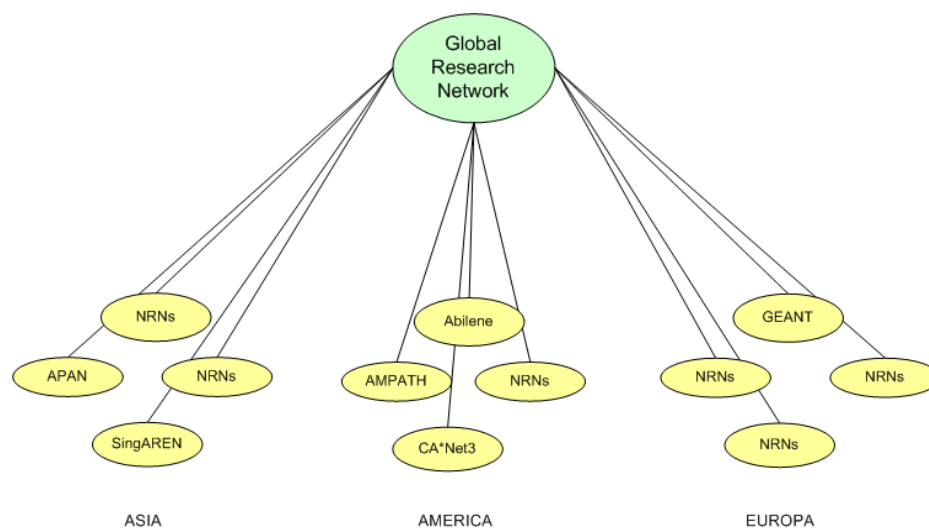
Fuente: <http://www.apan.net/documents/presentation.html>

Figura N° 13 Redes Internacionales de Grid Computing



Fuente: Elaboración Propia.

Figura N° 14 Global Terabit Research Network



Fuente: Elaboración propia.

Figura N° 15 Vista física del Global Terabit Research Network



Fuente: <http://archive.dante.net/conference/globalsummit2002>

1. Ventajas de Grid Computing

La computación distribuida es un sistema en el cual los recursos computacionales se comparten a través de las redes. Esta tecnología utiliza estándares para comunicar recursos computacionales heterogéneos. Mediante la aplicación de estos estándares se posibilita el seleccionar, compartir y agregar recursos que pertenecen a distintos dominios administrativos y residen en distintas áreas geográficas. Los recursos son compartidos con base en su disponibilidad, capacidad, costo y calidad del servicio. IBM Corporation en su artículo IBM Grid Computing for Economic Development, indica que las empresas y gobiernos necesitan enfocarse en el desarrollo económico, para lo cual deberían centrarse en tecnologías no tradicionales, en las cuales la distribución geográfica no sea una limitante.

La comparación de los enfoques tradicionales y colaborativo se muestra en la Tabla N° 1 Desarrollo económico tradicional y colaborativo”.

Tabla N° 1 Desarrollo económico tradicional y colaborativo

Desarrollo económico tradicional	Desarrollo económico colaborativo
Se enfoca en las industrias tradicionales Manufactura Distribución Transporte	Se enfoca en las industrias de capital intelectual: Investigación, Tecnologías y Servicios.
Sensible a los medios de transporte, lugar de ubicación y fuerza laboral	Sensible a la generación de ideas, colaboración y capital de inversión.
Enfrente gran competencia proveniente de lugares con bajos costos de mano de obra.	Menos susceptible a la globalización.
El gobierno coopera con la selección de ubicaciones geográficas adecuadas, infraestructura física y reducciones de	El gobierno contribuye en la creación de redes de colaboración y acceso a recursos críticos.

Desarrollo económico tradicional	Desarrollo económico colaborativo
impuestos	

Fuente: IBM Corporation.

Según el enfoque de IBM, la tecnología de grid computing contribuye al desarrollo económico de las comunidades en los siguientes aspectos:

- Se reúne a las organizaciones locales, líderes de negocio, gobierno e instituciones de educación para que trabajen en la solución de problemas de manera cooperativa y colaborativa.
- Es un medio de bajo costo que permite a los gobiernos locales estimular el desarrollo económico, pues se atrae a personas con perfiles altamente calificados y se mantiene el crecimiento de las empresas.
- Les provee, a las empresas locales, acceso a sistemas computacionales sofisticados que de no ser por la informática distribuida serían inaccesibles debido a los costos económicos asociados. Además, si se cuenta con estos sistemas de computación, será posible atraer inversión en segmentos de la industria sofisticados.
- Se utiliza la infraestructura existente de las universidades locales y de las oficinas del gobierno y mediante el grid computing se hace flexible, ya que se pueden agregar más socios de infraestructura cuando sea necesario.

Con el fin de evaluar el valor potencial de la tecnología de grid computing en las comunidades, IBM Global Services ofrece un proyecto piloto en el cual se consideran los aspectos necesarios para incentivar el desarrollo económico local por medio de grid computing. Como resultado de este estudio, se obtiene un caso de negocio que identifica los puntos más importantes sustentadores del desarrollo económico y se brinda recomendaciones con respecto a cómo el grid computing puede contribuir en los esfuerzos de la comunidad. Los pasos que incluye dicha evaluación son los siguientes:

- Valor de las capacidades del grid para las industrias de la región.
 - Grid de colaboración.
 - Grid de datos compartidos.
 - Grid de procesamiento de cálculos, por ejemplo SETI@Home.
 - Grid de integración.
- Disponibilidad de recursos valiosos para las industrias locales.
- Se enfoca en construir redes de colaboración y brinda soporte para los negocios en crecimiento.
- Identifica el valor del grid computing para comunidad comparado con otras alternativas de inversión.

Según IBM Corporation, Buyya (2002)²⁶ y Foster (1999)²⁷, el uso de la tecnología de informática distribuida le brinda a las organizaciones las siguientes ventajas:

- Consolidación y mejoramiento de la eficiencia del poder computacional, el manejo de datos y de los recursos de almacenamiento.
- La flexibilidad para enfrentar las necesidades de procesamiento de información. Grid computing permite que una empresa administre eficientemente sus recursos, ya que agregar o quitar computadoras al grid depende la capacidad de procesamiento necesario en un momento en particular.
- Es un servicio de alta calidad a un bajo costo, porque se aprovechan todos los recursos de hardware disponibles. Debido a que dentro de un grid los computadores pueden ser heterogéneos, hace posible que se utilice el poder de computadoras ubicadas dentro de la empresa.

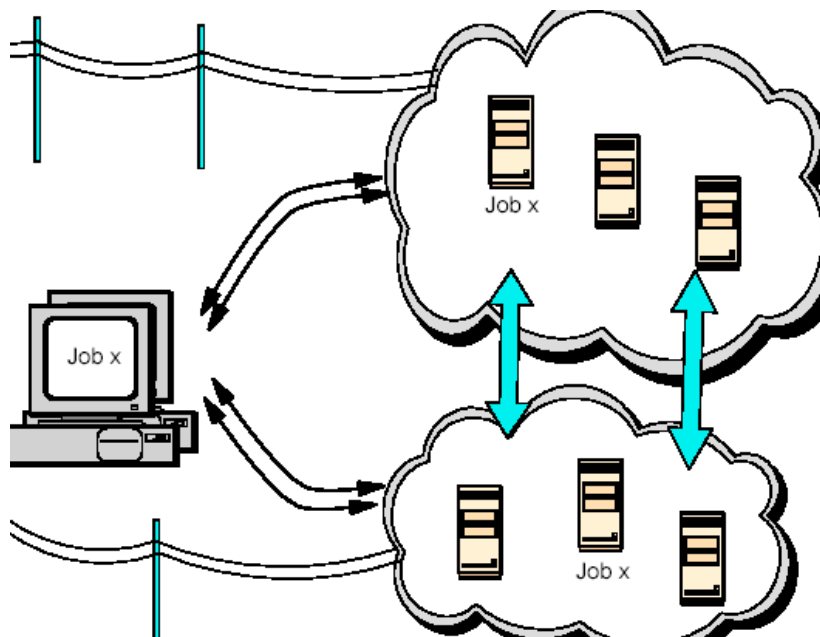
²⁶ Buyya Rajkumar, Stockinger Heinz, Giddy Jonathan, Abramson David. (2002). *Economic models for resource management and scheduling in Grid computing*. Monash University, Melbourne, Australia y CERN, European Organization for Nuclear Research. 2002.

²⁷ Ian Foster and C. Kesselman. (1999). *The Grid: Blueprint for a Future Computing Infrastructure*, Morgan Kaufmann Publishers, USA, 1999.

- Hace posible la creación de organizaciones virtuales que comparten aplicaciones y datos.
- Reducción del costo total de la propiedad (TCO).
- Incrementa la productividad y reduce los costos. Esto se debe a que las empresas no tienen que invertir en supercomputadoras, las cuales –normalmente- cuestan varios millones de dólares.
- Incrementa la productividad y la colaboración.
- Permite la resolución de problemas que antes era imposible debido a restricciones de poder de procesamiento de información.
- Permite que departamentos y negocios dispersos creen organizaciones virtuales con el fin de compartir datos y recursos.
- La infraestructura tecnológica de la informática distribuida es flexible de forma tal que se adapta a las necesidades operacionales de la empresa.
- Se responde de forma rápida y eficiente ante las fluctuaciones de demanda del consumidor; de esta manera, cuando se requieren más recursos de procesamiento se obtienen del grid sin necesidad de realizar grandes inversiones en infraestructura propia.
- Incrementa la productividad porque se posee acceso ilimitado a recursos de procesamiento, datos y almacenamiento. En pocas palabras, la empresa los utiliza cuando los necesita.
- Ayuda al uso adecuado de las inversiones de capital, pues se produce un uso óptimo de las facilidades computacionales.
- Se reducen los errores de inversión en activos, pues -muchas veces- el sobredimensionamiento de los equipos es elevado, lo cual provoca que las inversiones en infraestructura computacional sean más altas de lo necesario.
- La infraestructura de procesamiento de información posee una administración de la carga controlada.
- Se obtiene la capacidad necesaria para ejecutar aplicaciones de alta demanda de procesamiento.

- Se incrementa el acceso a datos y la colaboración:
 - Los datos pueden ser federados y estar distribuidos globalmente.
 - Brinda soporte a grandes equipos multidisciplinarios.
 - Hace posible la colaboración entre organizaciones y entre empresas.
- Provee una infraestructura altamente confiable:
 - Se produce un balance de las cargas de trabajo.
 - Se promueve la comunidad de los negocios.
 - Hace posible la recuperación de fallas más rápidamente; por ejemplo, se puede contar con conexiones a varios grids de procesamiento. De esta forma, si una conexión falla, los trabajos pueden ejecutarse en el otro, tal y como se muestra en la Figura N° 16.

Figura N° 16 Características redundantes de la informática distribuida



Fuente: IBM Corporation, Introduction to Grid Computing with Globus.

2. Clasificación de los modelos de informática distribuida

El Insight Research Institute (2005) ha clasificado los modelos de informática distribuida según el tipo de recursos compartidos. Dicha clasificación se muestra en la Tabla N° 2.

Tabla N° 2 Segmentación del mercado de informática distribuida según tipo de recurso

Modelo	Descripción
Grids de computación	Brinda acceso compartido a un conjunto de sistemas computacionales de alto desempeño.
Grids de datos	Brinda acceso compartido a un conjunto de sistemas de bases de datos y archivos.
Grids de instrumentación	Brinda acceso compartido a instrumental científico.
Grids de aplicación	Brinda acceso compartido a aplicaciones.
Grids de búsqueda de recursos ociosos (scavenging grids)	Se utilizan para encontrar y utilizar ciclos de procesamiento en computadoras que no están siendo empleadas para que sean usados en tareas de alto procesamiento. Generalmente, este tipo de grid se implementa de forma tal que no interfiera con las operaciones del usuario final de la computadora.

Fuente: Insight Research Institute.

De la misma forma, la informática distribuida puede clasificarse según el tipo de organización en la cual residen los componentes del gris. Dicha clasificación se detalla en

la Tabla N° 3, en la cual se puntualizan los principales modelos de grid computing según el tipo de organización:

- Grid corporativo.
- Grid con socios.
- Grid de servicios.

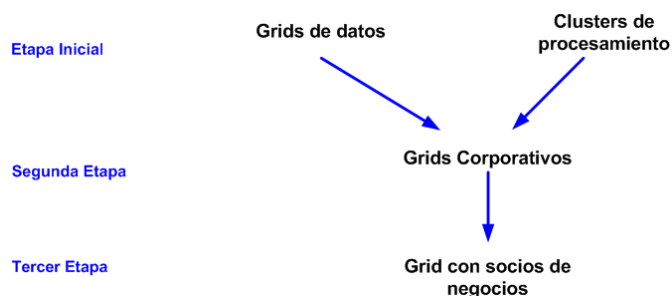
Tabla N° 3 Segmentación del mercado de informática distribuida según tipo de organización

Modelo	Descripción
Grid Corporativo	Recursos privados compartidos en una misma corporación. Es el más sencillo de todos. El grid está formado por pocas computadoras unidas entre sí. Todo el hardware es idéntico y cuenta con la misma versión del sistema operativo. La administración es más sencilla y podría clasificarse como un cluster de computación.
Grid con socios	Se permite que los socios de negocio tengan acceso a los recursos computacionales de una corporación.
Grid de servicios	Se brinda un servicio de acceso público a recursos computacionales a través de Internet.

Fuente: Insight Research Institute.

En la Figura N° 17, se muestra la evolución natural del uso de la tecnología de informática distribuida. En la etapa inicial, las organizaciones cuentan con facilidades para procesar datos y hacer cálculos; trabajan de forma aislada dentro de los distintos sectores de la empresa. Posteriormente, se logran consolidar los recursos para formar un grid corporativo, al cual se pueden conectar todos los departamentos de la empresa y utilizan los recursos bajo un modelo de procesamiento por demanda. En la etapa final, se logra conectar y unificar los recursos corporativos y de los socios de negocios. Así, se aprovechan los recursos de todas las partes de la cadena de valor del negocio.

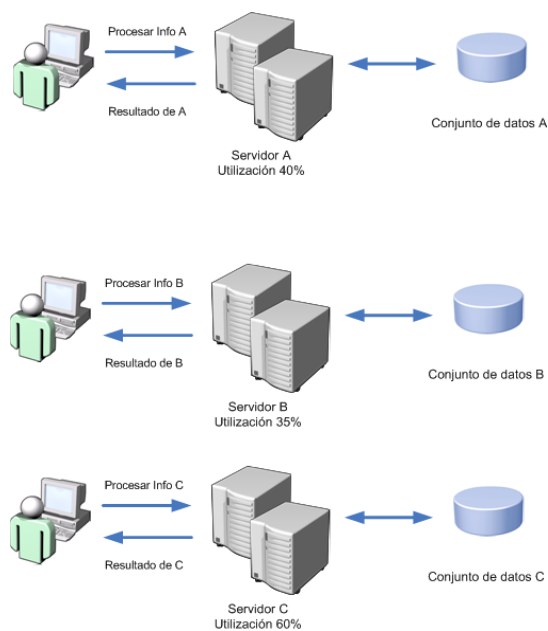
Figura N° 17 Evolución de la tecnología de informática distribuida.



Fuente: Platform Computing.

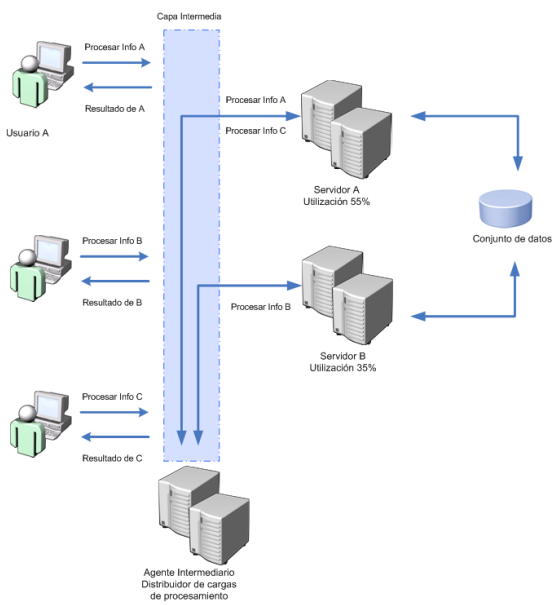
En la Figura N° 18, Figura N° 19 y Figura N° 20, se pueden comparar los distintos enfoques para la implementación de la informática distribuida. Se inicia con una implementación tradicional para después pasar a una de un grid de procesamiento local. Finalmente, se muestra el esquema de una implementación de un grid remoto.

Figura N° 18 Infraestructura computacional estándar



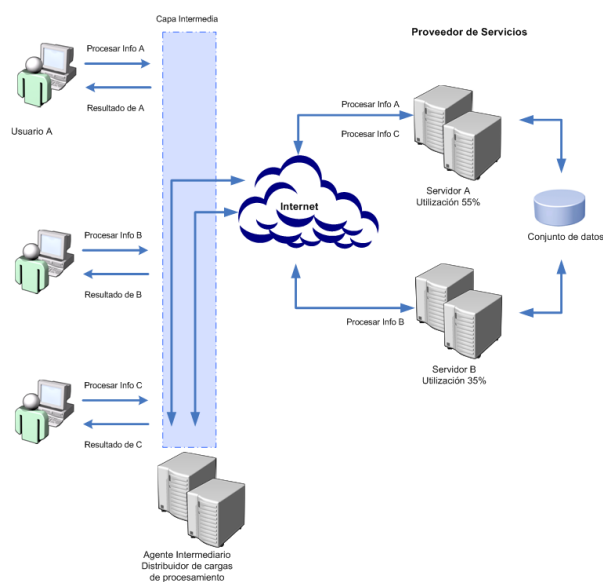
Fuente: Elaboración Propia.

Figura N° 19 Implementación local de un grid



Fuente: Elaboración propia.

Figura N° 20 Implementación remota de un grid

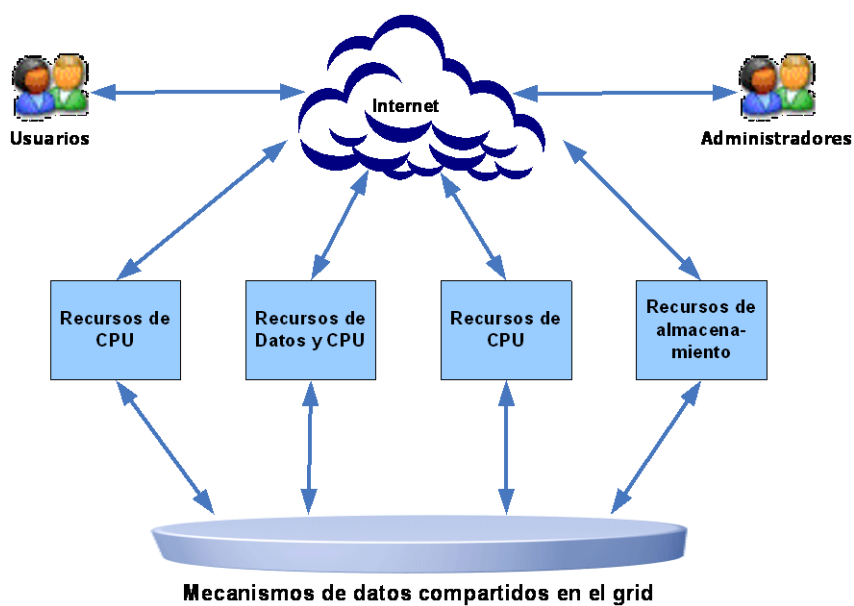


Fuente: Elaboración propia.

Actualmente, el tipo de grid que se ha desarrollado más es el corporativo -ver Figura N° 21-, pero se espera que los grids con socios -Figura N° 22- y de servicios -ver Figura N° 23 y Figura N° 24- aumenten su presencia en el mercado.

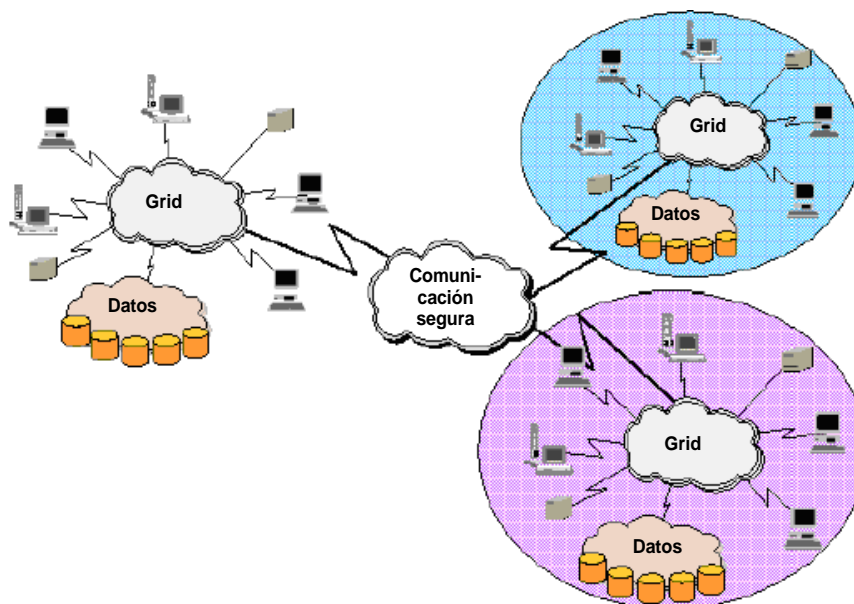
En la Figura N° 25, se observa un ejemplo de modelo de InterGrid, el cual permite que varios proyectos de informática compartan recursos e información, de esa forma todos los participantes cuenta con mayor poder computacional.

Figura N° 21 Ejemplo de grid corporativo simple



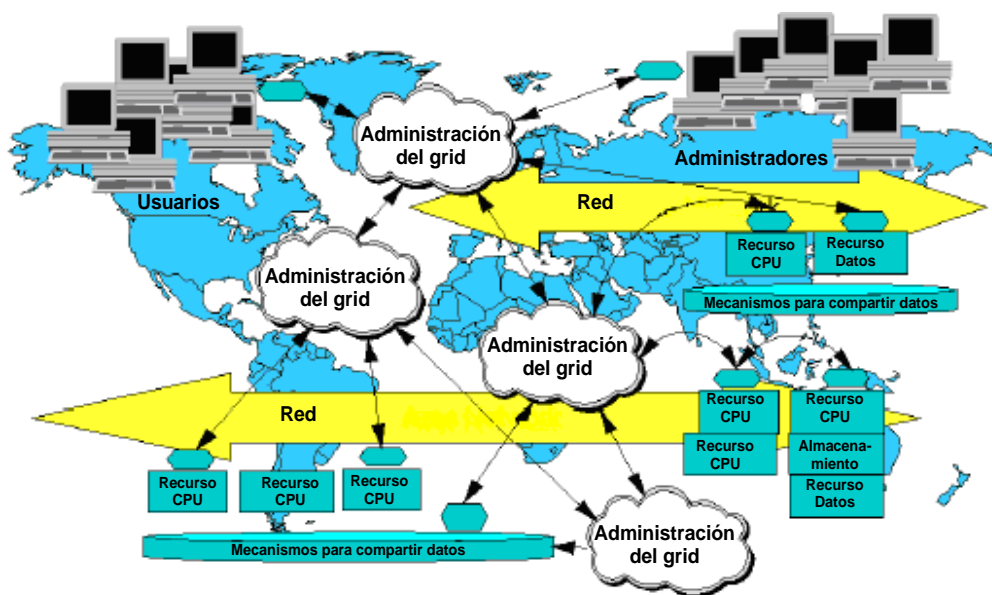
Fuente: IBM Corporation.

Figura N° 22 Los grids con socios pueden comunicar dos o más empresas



Fuente: IBM Corporation.

Figura N° 23 Grid geográficamente distribuido



Fuente: IBM Corporation.

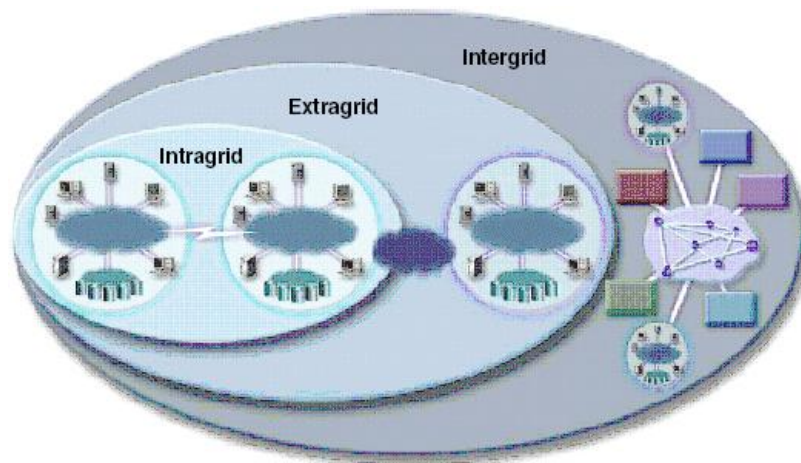
Para explotar al máximo el potencial del mercado, los proveedores deben enfocarse en brindar servicios y soluciones a problemas de industrias específicas. De esta forma, se hace énfasis en un mercado vertical de informática distribuida y no en un mercado horizontal, los cuales se enfocan en grid genéricos.

El estudio del Insight Research Institute (2005) identifica los siguientes mercados verticales:

- Aplicaciones de ciencias físicas: simulación de terremotos, aplicaciones de observación planetaria, física de partículas de alta energía.
- Aplicaciones de ciencias de la vida: simulación de diagnóstico del cáncer, imágenes de alta resolución para las neurociencias.

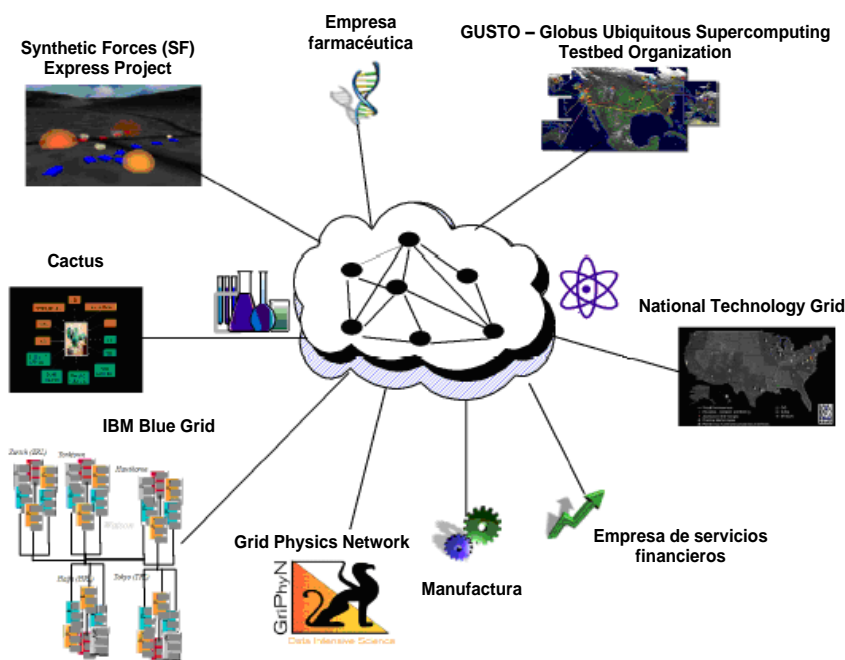
- Aplicaciones comerciales:
 - Aplicaciones farmacéuticas, biomédicas y biotecnológicas: investigación farmacéutica, análisis de proteínas.
 - Aplicaciones de ingeniería y automatización del diseño: diseño de partes para aviones, diseño de circuitos de computadoras, distribución de imágenes aéreas y satelitales.
- Aplicaciones de servicios financieros: aplicaciones de manejo de inversiones bancarias, para el modelaje de seguros de vida, de administración del riesgo.
- Aplicaciones de recursos humanos: soluciones corporativas de manejo de datos centralizados, aplicaciones de servicios de información.
- Aplicaciones de consumo masivo: juegos en línea.

Figura N° 24 Relación de grid corporativo (Intragrid), grid con socios (Extragrid) y grid de servicios (Intergrid)



Fuente: IBM Corporation.

Figura N° 25 Ejemplo de Intergrid que combina los recursos de varias corporaciones



Fuente: IBM Corporation.

3. Aspectos que deben tomar en cuenta los gerentes de informática

Al igual que cualquiera nueva tecnología, las empresas son cautas en cuanto a su implementación. Los gerentes se preguntan de qué forma puede aplicarse esa tecnología para mejorar el negocio, cuáles son las experiencias en este campo. Minoli (2005) brinda una guía, la cual incluye los aspectos clave que debe conocer un gerente respecto a la informática distribuida. Estos se mencionan a continuación:

- ¿Qué es grid computing y cuáles son sus puntos más importantes?
 - Beneficios del grid y cuál es el estado de la tecnología.
 - Motivos por los cuales se debe considerar los grids computacionales.
 - Breve reseña histórica del grid computing.
 - ¿Está lista la tecnología para ser implementada en las empresas comerciales?
 - Proveedores de la tecnología e implementadores pioneros.

- Retos.
 - Futuro y rumbo de la tecnología.
- ¿Cuáles son los componentes de una infraestructura de grid computing?
 - Portales e interfaces de usuario.
 - Seguridad de los usuarios.
 - Agente intermediario (broker).
 - Función de calendarización.
 - Función de administración de datos.
 - Administración de tareas y de recursos.
- ¿Existe algún estándar que dé soporte al grid computing?
 - ¿Qué es OGSA/OGSI?
 - Implementaciones de OGSI
 - Servicios OGSA
 - Creación y administración de organizaciones virtuales
 - Grupos de servicios y descubrimiento de servicios
 - Orquestación y flujos de trabajo
 - Transacciones
 - Servicios de medición
 - Servicios de contabilidad
 - Servicios de facturación y pago
- Consideraciones de seguridad
 - ¿Se puede confiar en el grid?
 - ¿Cuáles son los principales problemas de instalación y administración del grid?
 - ¿Cuáles son los sistemas económicos del grid?
 - El servicio de pago
- ¿De qué manera se unen todos los servicios?
 - Infraestructura de comunicaciones y de red

Una vez que se hayan identificado los principales puntos de la tecnología de grid computing, será posible que se realice su factibilidad para aplicarla en una empresa determinada.

Zhang y Chung (2002) identifican al grid computing como la evolución hacia el próximo nivel después de Internet, pues será posible utilizar aplicaciones a través de la red y permitirán que se integren las operaciones de las empresas poseedoras de oficinas a nivel global.

En el campo de los negocios Chetty y Buyya (2002), identifican que los beneficios del grid computing se pueden transformar en ventajas competitivas dentro del mercado. Entre ellas se encuentran:

- Posibilidad de compartir recursos.
- Provee un acceso transparente a los recursos remotos.
- Hace un uso efectivo de los recursos computacionales.
- Permite que se unan los recursos de distintas ubicaciones geográficas para cumplir las necesidades de demanda de procesamiento.
- Reduce la cantidad de servidores necesarios, alrededor de un 25% a 75%.
- Brinda acceso a software y bases de datos remotas.
- Reduce el tiempo de ejecución de aplicaciones de procesamiento de alto consumo.
- Se puede distribuir la carga entre distintas plataformas de software y hardware.
- Provee tolerancia a fallos.
- Toma ventaja de las diferencias de zonas horarias y de la diversidad. Por ejemplo, durante horas pico, los usuarios pueden utilizar recursos ubicados en otras zonas horarias donde no es hora pico.
- Permite la creación de un centro de datos virtual.

- Provee la flexibilidad de cumplir con alzas en la demanda que no habían sido previstas, debido a que se pueden alquilar recursos externos en lugar de adquirirlos, con lo cual se reducen los costos de operación.

C. Comparación entre el grid eléctrico y el grid computacional

Chetty y Buyya (2002) indican que para realizar una comparación entre las redes de transmisión eléctrica o “power grid” es vital entender cómo se compone la infraestructura de este tipo de redes.

Desde un punto de vista general, las fuentes de alimentación para una red eléctrica son variadas (petróleo, carbón, gas, nuclear, plantas hidroeléctricas y geotermales, entre otras) y heterogéneas. Sin embargo, desde un punto de vista básico, estas distintas fuentes proveen el insumo necesario para producir energía eléctrica.

El sistema de redes eléctricas está conformado por varios buses de transmisión o puntos de intersección, los cuales están interconectados por medio de líneas de transmisión. Una vez que se cuenta con energía eléctrica, se utilizan generadores para enviarla a través de los buses de transmisión.

En la Figura N° 26, se muestra un mapa de la red eléctrica de Estados Unidos de América. Se puede observar la distribución de los distintos puntos de concentración y generación de electricidad y los buses de transmisión de electricidad.

Figura N° 26 Mapa de la red eléctrica de Estados Unidos de América



Fuente: Departamento de energía de EEUU.

Las redes de transmisión eléctrica, muchas veces, se extienden más allá de las fronteras de los países, cada uno de los cuales cuenta con sus propios modelos organizacionales y políticas económicas. Por ejemplo, el proyecto SIEPAC²⁸ tiene como objetivo reforzar la red eléctrica de América Central: Guatemala, El Salvador, Honduras, Nicaragua, Costa Rica y Panamá. Por razones de economía de escala y de uso eficiente de los recursos, la mayoría de los sistemas eléctricos están interconectados para formar *power grids* (redes de energía). Los componentes de estas redes son conocidos como *power pools*²⁹ (conjuntos de energía), cada uno de ellos posee su propia tecnología y son independientes desde el punto de vista económico. Sin embargo, los miembros han establecido una serie de políticas comunes en cuanto a la generación de electricidad y los precios de la energía.

La Figura N° 27 muestra los componentes típicos de una red de transmisión eléctrica: transmisión, subtransmisión y distribución. El sistema de transmisión administra grandes

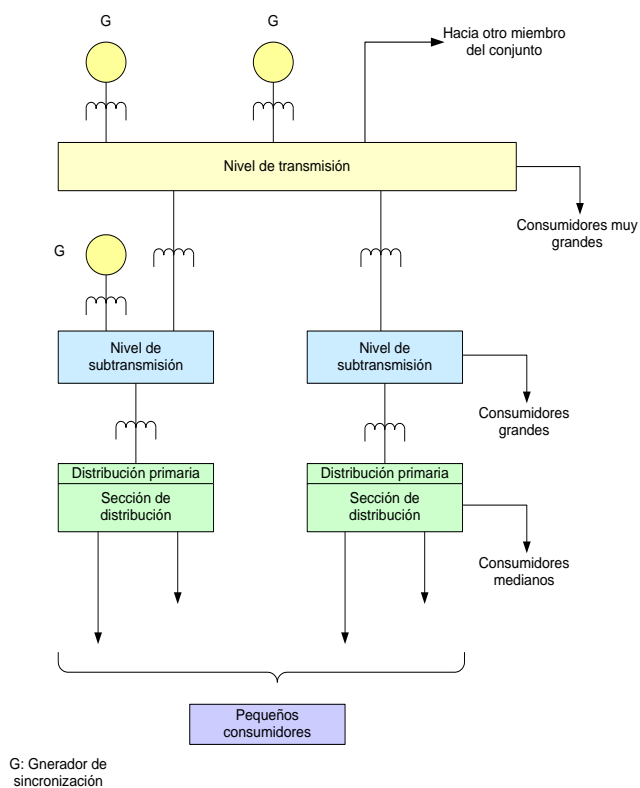
²⁸ Mayor información en <http://www.eprsiepac.com>

²⁹ Entidades en las cuales los productores de electricidad pueden combinar su producción y vender la energía resultante de la unión como un todo. Fuente: <http://www.siliconvalleypower.com/info/?doc=glossary>

cantidades de energía y se encarga de interconectar a todas las estaciones generadoras y todos los puntos de carga del sistema. Por medio de este sistema la energía puede enviarse en la dirección deseada, de forma tal que se obtengan los mayores beneficios económicos o que sea la mejor solución técnica.

El sistema de subtransmisión se encarga de transmitir la energía a altos voltajes a un área geográfica determinada. Por último, el sistema de distribución se encarga de enviar la energía hacia los consumidores finales.

Figura N° 27 Sistemas de transmisión de red eléctrica



Fuente: Chetty M. y Buyya Rajkumar (2002).

En un sistema eléctrico se cuenta con recursos tales como vapor, energía hidroeléctrica o generadores nucleares. En un sistema de informática distribuida, se cuenta con distintos

tipos de recursos computacionales: clusters, supercomputadoras, estaciones de trabajo, espacio de almacenamiento, entre otros. Están integrados por medio de un grid de forma tal que para el usuario es una fuente uniforme de poder computacional. De igual forma que los *power pools* están interconectados, los grids de poder computacional se hallan conectados a través de Internet por medio de puntos de red y enrutadores que transmiten los datos de una red a otra, con el fin de proveer servicios tales como transferencia de información entre los consumidores finales y el grid de poder computacional.

En la red eléctrica, las tasas de transmisión se miden en voltios; sin embargo, en el grid de poder computacional se hace por medio del ancho de banda disponible -o sea- la cantidad de datos que pueden ser transmitidos a través de la red. Los sistemas de transmisión de energía de la red eléctrica encuentran su análogo en las conexiones de transmisión de datos sobre fibra óptica y conexiones ATM -cuando se trata de grandes cantidades de datos- y conexiones T1 y Ethernet (en el caso de cantidades de información medianas y pequeñas).

De forma similar que en la red eléctrica, se conectan cargas heterogéneas -dispositivos mecánicas, instrumentos eléctricos- en el grid de poder computacional; la carga también puede ser heterogénea, por ejemplo: procesamiento de gráficos y multimedia, sistemas de resolución paralela de problemas, procesamiento de señales digitales, entre otros.

En la red de transmisión eléctrica, el consumidor final no sabe cuál es el origen de la energía que está utilizando; sin embargo, en el grid de poder computacional se tiene evidencia de los recursos que están siendo utilizados por una carga de trabajo en particular. Chetty y Buyya (2002) proponen una analogía entre las redes eléctricas y los grids de procesamiento, el detalle de dicha comparación se observa en la Tabla N° 4.

Tabla N° 4 Comparación entre la red de transmisión eléctrica y la informática distribuida. Fuente: Chetty y Buyya (2002)

Parámetro	Red de poder eléctrico	Red de poder computacional
Recursos	Heterogéneos tales como termales, hidroeléctricos, eólicos, solares, nuclear, entre otros.	Heterogéneos tales como computadoras personales, estaciones de trabajo, clusters, los cuales poseen distintos sistemas operativos y sistemas de administración.
Red	Líneas de transmisión y cables subterráneos. Sistemas sofisticados de protección de líneas.	Se utiliza Internet para conectar los recursos distribuidos y las cargas de trabajo.
Entidades Análogas	<ul style="list-style-type: none"> ▪ Bus ▪ Transmisión de energía ▪ Voltaje ▪ Sistemas de transmisión colectiva (230 kV-760 kV) ▪ Subtransmisión (25 kV-150kV) ▪ Distribución (120/240V, 25kV) ▪ Cable ▪ Energía (KW-hora) ▪ Capacidad de almacenamiento por medio de baterías 	<ul style="list-style-type: none"> ▪ Nodo ▪ Transmisión computacional ▪ Ancho de banda ▪ Sistemas de transmisión colectiva Fibra Óptica (2.4 Gbits por segundo) ▪ Ethernet (45 Mbits por segundo) ▪ Modem, ISDN (56-128 Kbits por segundo) ▪ Cable ▪ Poder computacional (ciclos por segundo) ▪ Cualquier dispositivo de almacenamiento

Parámetro	Red de poder eléctrico	Red de poder computacional
		(megabytes)
Estación de generación	Conocida como estación de poder	Conocida como Recurso del Grid (computadoras, fuentes de datos, bases de datos)
Tipo de carga (con base en el tipo de utilización)	Dispositivos heterogéneos que utilizan energía eléctrica; por ejemplo: televisores, ventiladores, entre otros.	Aplicaciones heterogéneas que necesitan poder de procesamiento; por ejemplo: multimedia, resolución de problemas científicos o de ingeniería.
Frecuencia de operación	Frecuencia uniforme 50 o 60 Hertz	No es uniforme, depende del poder de procesamiento de la computadora.
Interfaz de acceso	Acceso directo por medio de toma corrientes o transformadores industriales	Se utiliza una interfaz para acceder a los distintos recursos utilizados.
Facilidad de uso	Muy simple: conecte y utilice.	Muy complejo, pero se espera que conforme la tecnología se consolide.
Dispositivos para regular distintos niveles de energía: voltaje, ancho de banda, velocidad de CPU.	Transformadores: se encargan de cambiar de un voltaje a otro.	Los agentes intermediarios realizan la selección de los recursos que cumplen los requerimientos del usuario y que pueden ejecutar, adecuadamente, las tareas asignadas
Agregación de	Cuando una carga requiere más	Cuando una aplicación necesita

Parámetro	Red de poder eléctrico	Red de poder computacional
recursos	poder del que se puede proveer localmente, se suministra poder adicional a través del grid. Los centros económicos de distribución utilizan algoritmos de calendarización y estudios de flujos de carga para administrar mecanismos que lleven a cabo esta acción.	más poder computacional del que puede proveer un recurso, el grid computacional permite agregar recursos para ejecutar componentes de la aplicación en paralelo, como Nimrod-G (Buyya 2006): proveen esta capacidad.
Confiabilidad	Las líneas de transmisión importantes son duplicadas. Se utilizan sistemas sofisticados de protección de las estaciones generadoras, líneas de transmisión y de los equipos de generación.	Los recursos del grid pueden fallar en cualquier momento. Por lo tanto, es tarea de los agentes intermediarios administrar estas fallas en tiempo de ejecución.
Escalabilidad	La escalabilidad es crucial para mantener los generadores en sincronía. Se utilizan algoritmos de control para asegurar esta estabilidad.	Depende de la política de administración de recursos. Si el recurso es compartido, entonces la capacidad disponible para un usuario puede cambiar en un intervalo de tiempo.
Capacidad de transmisión	La capacidad máxima de estas líneas se basa en sus límites termales.	La capacidad máxima se basa en el ancho de banda del medio de transporte.
Almacenamiento	Se utilizan baterías de poca capacidad de almacenamiento.	No se puede almacenar el poder computacional.
Contabilidad	Se utilizan mecanismos de	Los sistemas de administración

Parámetro	Red de poder eléctrico	Red de poder computacional
automática	medición de energía.	utilizan mecanismos de medición locales. Además, los agentes intermediarios pueden realizar esta medición
Interconexión de los sistemas de poder	Los sistemas eléctricos regionales se interconectan por medio de líneas de transmisión.	Este concepto todavía no existe.
Operación desregulada	Se presenta una operación exitosa en países con suficiente capacidad de generación.	Conforme la tecnología madure y las empresas empiecen a utilizarla, su operación será exitosa.
Operación regulada	Un centro de despacho de carga administra la operación óptima del sistema.	Existe gran potencial a fin de utilizar mecanismos de precios establecidos por el mercado para que ayude a regular la oferta y demanda de los recursos.
Reguladores	La administración la realizan asociaciones de generadores de electricidad o el gobierno.	Actualmente, no existe este mecanismo.
Cuerpo de estándares	Existen estándares mundialmente extendidos.	Existen iniciativas para crear un estándar para la industria.

Fuente: Chetty y Buyya (2002).

D. Componentes de grid computing

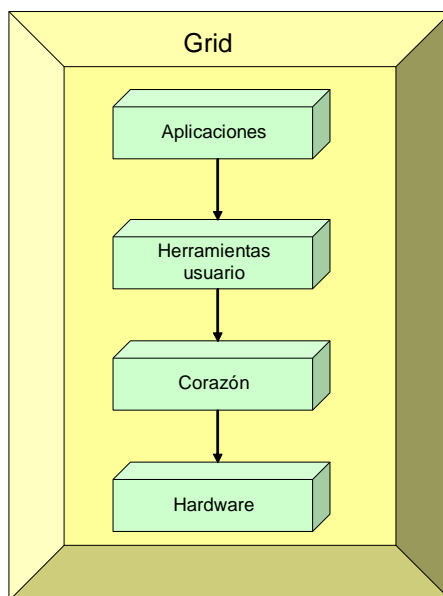
En este apartado se introducen los principales componentes que se hallan en la arquitectura de grid computing. En general, las características del grid proveen a los usuarios un ambiente computacional para ejecutar sus tareas. Estas características son las siguientes:

- **Heterogeneidad.** Un grid posee múltiples recursos disponibles en ambientes heterogéneos, ya que cada sistema puede estar compuesto por distintos tipos de hardware.
- **Escalabilidad.** Un grid puede formarse por pocos computadores o por miles de ellos. Esto conlleva un posible problema de desempeño, pues las aplicaciones que necesitan gran cantidad de poder computacional deben tomar en cuenta los tiempos de latencia y anchos de banda limitados, porque los miembros del grid están distribuidos geográficamente.
- **Distribución geográfica.** Los computadores pertenecientes al grid están geográficamente distribuidos y pertenecen a distintas empresas.
- **Adaptabilidad.** Debido a que los recursos son dinámicos, ellos no están siempre disponibles; por eso, las aplicaciones tienen que adaptarse a los cambios de la disponibilidad de los recursos.

Un grid posee distintos componentes -ver Figura N° 28- los cuales permiten que los recursos estén disponibles para las aplicaciones de los usuarios. Estos componentes son los siguientes:

- Hardware del grid. Sus recursos distribuidos geográficamente pueden ser accedidos desde cualquier lugar. Comprenden computadoras, dispositivos de almacenamiento, instrumentos científicos, bases de datos, entre otros.
- Corazón del grid. Se encarga de brindar servicios tales como administración de procesos, control de recursos y almacenamiento. También reserva y negocia los recursos del grid.
- Herramientas de usuario final. Incluye las aplicaciones utilizadas por los usuarios para desarrollar aplicaciones que se ejecutan en el grid.
- Aplicaciones del grid. Son las aplicaciones ejecutadas dentro del grid.

Figura N° 28 Componentes principales de una infraestructura de grid computing



Fuente: Elaboración propia.

Desde un punto de vista general, los componentes principales de grid computing incluyen lo siguiente:

- Administración de recursos: El grid debe conocer cuales son los recursos disponibles, de esa manera podrá asignar las tareas de procesamiento de forma adecuada.
- Administración de datos: Los datos deben ser transportados, aceptados, divididos y procesados.
- Servicios de administración: Tanto usuarios como aplicaciones deben tener la posibilidad de realizar consultas sobre el estado del grid.
- Administración de la seguridad: Los recursos del grid únicamente deben ser utilizados por los usuarios y aplicaciones autorizados.

Estos componentes pueden apreciarse en la Figura N° 29.

Figura N° 29 Componentes de la infraestructura administrativa del grid computing



Fuente: Elaboración propia.

E. Origen de los modelos económicos en la informática distribuida.

La tecnología grid hace posible que se resuelvan problemas, los cuales requieren un nivel elevado de procesamiento computacional, tarea que sería imposible solventar con una sola computadora.

La implementación de un grid necesita el desarrollo de distintos servicios, entre ellos: seguridad, información, administración de recursos, calidad de servicio, algoritmos de negociación³⁰, selección de recursos, calendarización de tareas y administración de la ejecución de ellas.

En esta tesis, nos concentraremos en la definición de un modelo económico para la administración de recursos dentro del grid y, de esta forma, brindar calidad de servicio a los usuarios.

El desarrollo de las tecnologías Peer-to-Peer y la informática distribuida las han posicionado como un modelo prometedor y candidato a convertirse en la nueva generación de plataformas de procesamiento de grandes cantidades de información. Este desarrollo permite la creación de Corporaciones Virtuales (Buyya 2004), estas corporaciones se caracterizan por compartir recursos que están distribuidos en distintos países alrededor del mundo.

Los componentes de este tipo de red de computadores están distribuidos geográficamente y su administración es altamente compleja, debido a que se cuenta con distintos tipos de

³⁰ Se refiere al proceso de negociación que se lleva a cabo entre el consumidor de servicios de computación y los proveedores de los mismos. Durante dicha negociación se pueden definir términos tales como el periodo durante el cual se pueden utilizar los servicios y el mecanismo de precios aplicado a un servicio en particular.

plataformas (UNIX, Windows, Linux, Macintosh) y pertenecen a diferentes organizaciones, las cuales aplican diversas políticas de administración de recursos.

Con respecto a las políticas, cabe resaltar que cada organización establece las que rigen la disponibilidad de los recursos: cantidad de procesamiento disponible, horas en las cuales se pueden atender y procesar cargas de información. Igualmente, existen políticas con respecto a los modelos de costos aplicados.

Debido a la naturaleza de este tipo de ambiente, muchos autores- entre ellos Stockinger, Abramson y Giggy (2004)- han sugerido la aplicación de reglas generales de administración basadas en modelos económicos que permitan determinar cómo se deben asignar los recursos y regir las condiciones de oferta y demanda.

El objetivo principal de estas reglas generales de administración de trabajo es regular y optimizar las políticas que impactan sobre el comportamiento de la oferta y demanda. Estas políticas están dictadas por los objetivos de cada consumidor y las funciones son utilizadas por los servicios de negociación y distribución del grid.

En la vida real, existe distintos modelos de mercado que se utilizan para establecer los precios de las mercancías o productos. Estos toman como base la oferta y demanda para determinar el precio dado al consumidor final. Entre estos modelos encontramos los económicos competitivos, como las subastas; los de precio fijo, se aplica el mismo costo para todos los consumidores; tiempo de uso, toma en consideración horas de alta y baja demanda; entre otros.

Tal y como se ha mencionado anteriormente, la infraestructura de la computación distribuida está conformada por recursos heterogéneos. Entre ellos se hallan computadores personales, estaciones de trabajo, clusters y supercomputadoras, los cuales combinados

brindan suficiente poder computacional para ejecutar aplicaciones de distinto tipo, por ejemplo, sistemas militares, científicos, comerciales, médicos, entre otros.

Cada una de estas aplicaciones necesita distintos requerimientos, tales como ciclos de CPU, memoria, capacidad de red, entre otros.

En la informática distribuida se tienen distintos roles; se caracterizan por poseer distintas metas, requerimientos, objetivos y estrategias. Entre estos roles se encuentran:

- Usuario final. Persona o software que tiene necesidad de procesar información.
- Consumidor. Es el software que se encarga de generar, recolectar y enviar la información que debe ser procesada por el grid.
- Productor. También conocido como dueño de los recursos. Este rol posee una gran cantidad de recursos productores de poder computacional que lo ponen a disposición de los consumidores. De esta forma, los interesados envían cargas de información que deben ser procesadas dentro de los recursos del productor.

Estos roles presentan una característica común: la distribución geográfica. Los consumidores pueden estar ubicados en cualquier punto del planeta que posea una conexión con Internet, debido a esto los usuarios y los recursos están ubicados en distintas zonas horarias lo cual dificulta la coordinación del uso de los recursos.

Para solventar los problemas de administración de los recursos distribuidos, tradicionalmente se ha recurrido a enfoques que se basan en la centralización de políticas de uso y distribución. Para que estas políticas sean ejecutadas apropiadamente, necesitan información completa del estado de los recursos y de las necesidades de los consumidores, lo cual es muy difícil de obtener y mantener actualizado. Además, en este tipo de enfoques se necesita que exista una política de administración de recursos común en todos los conjuntos de grids y en cada uno de sus componentes.

Una alternativa a estos enfoques es descentralizar las políticas de uso de recursos; sin embargo, se debe llegar a un consenso global sobre la interpretación y aplicación de dichas políticas. Por tanto, el problema de administración y asignación de recursos continúa siendo compleja. Ferguson y Yemini (1996) plantean que es imposible definir una política estática de administración de recursos que aplique a todo el sistema grid.

R. Buyya y D. Abramson (2000) sugieren el uso de un paradigma basado en la economía para administrar la asignación de recursos en un entorno de informática distribuida. Este enfoque provee las bases para la administración exitosa de recursos en un ambiente de recursos distribuidos. Los autores sostienen que los modelos económicos suministran algoritmos, políticas y herramientas para regir los recursos compartidos en un sistema de informática distribuida. Estos modelos pueden ser creados con base en precios o intercambios (trueque). En el primer caso, los recursos tienen un precio que se establece con base en la demanda, oferta y el valor de los recursos; en el segundo, todos los participantes tienen que poseer recursos e intercambiarlos con otros participantes.

Un sistema de administración de recursos, cuya base sea la economía, debe proveer un modelo que permita llevar a cabo las metas de los proveedores de servicios (dueños de los recursos dentro del grid) y de los consumidores. En este modelo se debe incluir las necesidades de los consumidores (demanda) y un mecanismo de asignación o calendarización de recursos, por medio del cual se creen estrategias que permitan escoger a los proveedores adecuados para suplir las demandas de los consumidores. Igualmente, se hace necesario establecer parámetros dentro del modelo, que admitan crear políticas de establecimiento de precios y así se incremente la utilización del sistema y permita a los proveedores de servicios ser más competitivos.

En un modelo perfecto, resulta imprescindible que el mercado sea competitivo, para lo cual se requieren mecanismos que ayuden a mantener el precio en equilibrio -o sea- el punto en que la oferta de un recurso es igual a su demanda.

1. Uso de los modelos económicos en la informática distribuida

En el caso de la computación distribuida, los modelos económicos ofrecen mecanismos para regir los recursos compartidos. Los modelos económicos permiten que se descentralice el control de los recursos y están basados en algoritmos matemáticos, los cuales pueden ser aplicados en los problemas de coordinación del uso de los recursos compartidos.

El aspecto de descentralización existe en la economía, pues está compuesta por agentes que intentan lograr sus propias metas, estos pueden ser clasificados en consumidores y proveedores. Los consumidores intentan optimizar sus estrategias para obtener los recursos que necesitan y en algunos casos no conocen el desempeño de todos los recursos existentes en la economía. Un proveedor fija su fin en satisfacer sus metas individuales -o sea- las ganancias, las cuales son derivadas de proveer recursos a los consumidores por una tarifa dada.

Generalmente, en la economía, el dinero y el precio son utilizados como un mecanismo de regulación de los intercambios entre los consumidores y los productores. Los primeros poseen dinero para comprar recursos y los segundos tienen una cantidad limitada de recursos para proveer. En el caso más sencillo, el precio que el productor establece para los recursos está dado por la oferta y la demanda.

Si se relaciona la economía con la informática, el rol de los consumidores lo toman las aplicaciones tales como transacciones, trabajos de procesamiento, transferencia de datos entre aplicaciones o redes, mecanismos de comunicación, entre otros. Los proveedores son los recursos computacionales que pueden atender a los consumidores: tiempo de procesamiento, memoria, disco, ancho de banda y sistemas de comunicación. El control de acceso a los recursos de los productores está regido, en primera instancia, por un precio; de ahí que se haga posible aplicar los modelos económicos de mercado al problema de control de acceso y venta de recursos en la informática distribuida.

2. Participantes en una economía de informática distribuida

La aplicación de modelos económicos en un ambiente de informática distribuida implica que existen al menos dos papeles dentro de este mercado: los proveedores de recursos, que tienen el papel de los productores (ProdR), y los consumidores de recursos o consumidor final (ConF). Este mercado -según Buyya, Abramson y Giddy (2001)- debe suministrar la infraestructura necesaria para su funcionamiento e incluye aspectos tales como seguridad, información digital, acceso remoto a los recursos que poseen los productores; además, mecanismos para que los consumidores y los productores puedan negociar tiempos y tarifas.

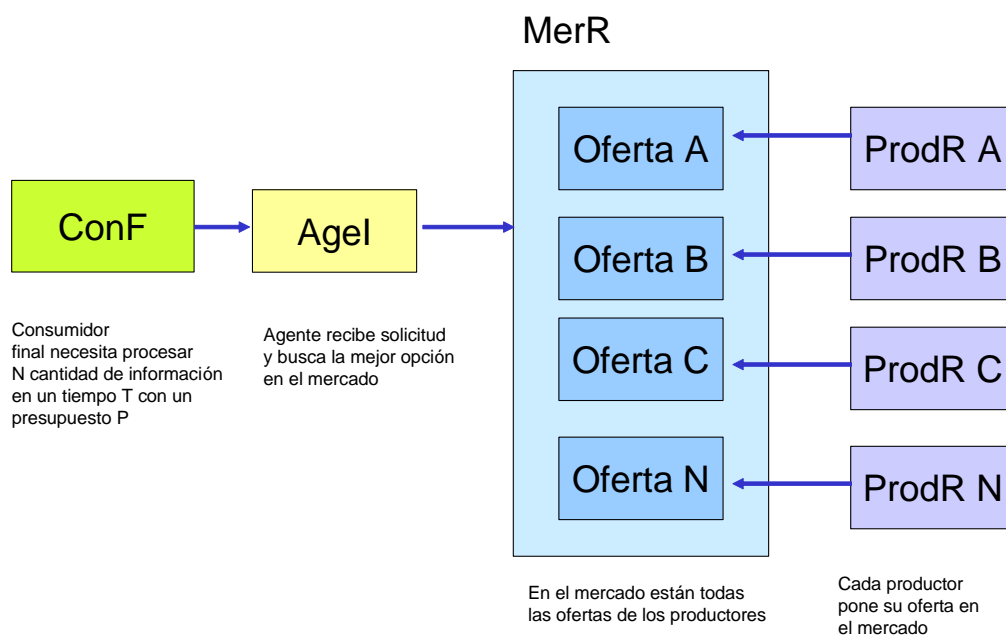
En este mercado (MerR), se hace necesario introducir un nuevo rol: el del agente intermediario (AgeI). Este se encarga de tomar las necesidades de los consumidores (cantidad de información, tiempos máximos para procesar la información, presupuesto) para negociar con los productores (costo por hora, costo por unidad procesada, costo por cantidad de nodos requeridos) en el ambiente computacional. Se le conoce como calendarizador.

El papel principal del agente intermediario es escoger la mejor opción para los consumidores y la elección realizada depende del modelo económico que se esté utilizando en el mercado. La interacción de los participantes en un ambiente de computación distribuida se muestra en la Figura N° 30.

Al igual que los mercados económicos convencionales, los consumidores representan la demanda y los productos representan a la oferta y los dos roles interactúan en un mercado. En él, los mecanismos económicos establecen cómo se suple la demanda de cada uno de los consumidores; aparte de las estrategias de los productores para satisfacer la demanda con su oferta.

De la misma forma, el comportamiento de los consumidores determina cómo se comporta el mercado y –principalmente– influyen sobre el establecimiento de los precios de las ofertas en el mercado.

Figura N° 30. Participantes en un mercado de informática distribuida



Fuente: Elaboración propia, con base en estudios realizados.

F. Modelo económico para la administración de recursos

Las plataformas de grid hacen posible que se utilicen fuentes de datos y computadoras heterogéneas para resolver problemas de gran escala en áreas tales como la ciencia, ingeniería y comercio. Sin embargo, la administración de los recursos y la calendarización de las tareas en este tipo de ambientes es un problema complejo principalmente debido a que los recursos están distribuidos y pertenecen a organizaciones con diferentes políticas de

uso y modelos de costos, los cuales varían dependiendo de los patrones de carga y disponibilidad (Buyya 2000).

Buyya (2000) y Ferguson (1996) determinaron que los productores de recursos y los consumidores tienen diferentes métricas de rendimiento, objetivos, estrategias y requerimientos. Para trabajar en este ambiente, estos autores han propuesto la creación de modelos económicos de administración de recursos; así se podrá regular la oferta y demanda de los recursos disponibles. Estos modelos permiten que los dueños de los recursos los pongan a disposición en el grid y motiva a los usuarios a tomar en cuenta aspectos tales como tiempo de procesamiento y costo computacional dependiendo de sus requerimientos de calidad de servicio. En esta tesis, se propone un modelo para evaluar la eficiencia de los modelos económicos propuestos, de forma tal, que se puedan comparar y elegir el más eficiente desde un punto de vista económico.

Los servicios de administración de recursos deben tomar en cuenta los requerimientos de los consumidores y los suplidores y, además, se tienen que adaptar continuamente a los cambios de disponibilidad de los recursos. Otros factores que afectan al modelo de calendarización son la autonomía de los sitios, la heterogeneidad de los computadores, las políticas de los proveedores, el control de los medios de transporte (red), la comercialización de servicios y la calidad del servicio.

Varios sistemas de grid computing, por ejemplo Globus -desarrollado por K. Czajkowski, I. Foster, N. Karonis, C. Kesselman, S. Martin, W. Smith, y S. Tuecke (1998)- han trabajado en modelos que toman en cuenta la mayoría de los puntos mencionados anteriormente; sin embargo, han dejado por fuera la comercialización de recursos y la calidad de servicio. Por lo tanto, el modelo propuesto en esta tesis tomará en cuenta esos dos puntos y se enfocará en el desarrollo de un modelo de evaluación de economía computacional genérica; de esta forma se logrará clasificar diferentes estrategias de calendarización utilizando distintos modelos económicos.

La idea de aplicar modelos económicos para la administración de recursos en sistemas distribuidos no es nueva, ya ha sido aplicada en sistemas distribuidos existentes. Por ejemplo: Popcorn, propuesto por N. Nisan, S. London, O. Regev, and N. Camiel (1987); Java Market, desarrollado por Y. Amir, B. Awerbuch y R. Sean Borgstrom (1998); Xenoservers, modelado por D. Reed, I. Pratt, P. Menage, S. Early, and N. Stratford (1999); Mariposa, propuesto por M. Stonebraker, R. Devine, M. Kornacker, W. Litwin, A. Pfeffer, A. Sah, and C. Staelin (1994) y Jaws, creado por S. Lalis and A. Karipidis (2000). Sin embargo, estos sistemas poseen modelos difícilmente generalizables o extensibles y su principal inconveniente radica en que los usuarios deben crear aplicaciones compatibles con el modelo, por lo tanto su aplicación ha sido limitada. En esta tesis se pretende crear un modelo económico adaptable a distintos ambiente y fácilmente extensible y adaptable, de forma que se pueda evaluar la eficiencia sin necesidad de cambiar los servicios ofrecidos.

En un ambiente de grid computing basado en modelos económicos, la administración de recursos debe poseer herramientas, las cuales permitan que los consumidores y los proveedores puedan establecer sus necesidades. Además, estos mecanismos tienen que hacer más fácil el logro de las metas de los proveedores y de los consumidores. Para tener claro los requerimientos del modelo, es necesario analizar las exigencias de los consumidores y de los proveedores.

Los consumidores precisan un modelo para solicitar recursos que cumplan con sus necesidades. Debe ser dinámico, así la elección de los proveedores será dinámica, basada en los requerimientos de disponibilidad, capacidad, costo y calidad de servicio establecida por el consumidor.

Por otro lado, los proveedores necesitan que el modelo les permita establecer los precios y mecanismos que maximicen el uso del sistema.

La obtención de acceso y utilización de recursos en forma, hace necesario establecer acuerdos entre las partes. Debido a que la cantidad de recursos es limitada, un problema básico de administración de recursos de un grid es saber quién podrá utilizar un recurso en un momento dado. Este inconveniente puede ser visto como una dificultad de oferta y demanda, para lo cual existen distintas soluciones en la teoría económica.

Desde este punto de vista, el grid puede ser visto como un mercado similar al mercado de valores de las sociedades económicas. En éstos, los proveedores de recursos tienen bienes en una cantidad finita y los consumidores desean tener acceso a ellos. En este contexto, los proveedores establecen un precio que los consumidores deben pagar para tener acceso a los bienes.

En el contexto de grid computing, podemos utilizar mecanismos de la teoría económica para equilibrar el mercado de recursos -o sea: la oferta y la demanda. Estos mecanismos permiten la construcción de sistemas que satisfacen las necesidades de los participantes en forma justa. La definición de mecanismos para llevar a cabo las negociaciones entre consumidores y proveedores permite su automatización, con lo cual se resuelve el problema de administración del grid. Así se crea un conjunto de relaciones y mecanismos reguladores del acceso a los recursos de forma dinámica, tomando como base el estado del mercado. Considérese el siguiente caso: un proveedor de recursos podría escoger entre todos los consumidores solicitantes del recurso a quien tenga la mayor probabilidad de proporcionarle la mejor ganancia; incluso, el proveedor y el consumidor del recurso después de cierto número de transacciones exitosas podrían renegociar el precio del bien.

Además, debido a que la participación de los proveedores dentro de un grid es voluntaria, para motivar su injerencia es necesario que exista un retorno, conocido en la teoría económica como retorno de la inversión.

En un escenario de grid computing, donde se aplica la teoría económica, es posible que un consumidor -quien desea utilizar recursos- delegue la labor de obtención de acceso a un agente. Este, con base en los requerimientos del consumidor se convierte en el encargado de localizar los recursos solicitados y negocia el precio conveniente con el proveedor o agente representante del recurso. A la vez, un proveedor de recursos también necesita escoger, entre todos los solicitantes de recursos, las ofertas más interesantes o que le incrementarán su utilidad.

Los distintos modelos económicos por aplicarse a esta situación se enumeran a continuación: modelo de precios publicados, modelo de regateo, modelo de oferta y demanda, modelo de subastas, modelo de intercambio cooperativo y modelo de monopolio. Cada uno de ellos se estudiará de manera más detallada posteriormente.

G. Estado del arte

A nivel mundial, existen varios proyectos de grid computing. La lista completa de estos puede encontrarse en el Grid Computing Info Centre de Rajkumar Buyya (2005). En la Tabla N° 5, se enumeran los proyectos más destacados hasta el año 2005.

Tabla N° 5 Principales proyectos de grid computing

Proyecto	Organización	Objetivos
NetSolve	Univeridad de Tenesse	Un sistema de programación para acceder bibliotecas y recursos de alto desempeño.

Proyecto	Organización	Objetivos
Unicore	Alemania	Un ambiente para acceder supercomputadoras de forma remota.
Globos	ANL-ISI	Provee un ambiente seguro para acceder recursos computacionales y de almacenamiento remotos.
GRACE	Universidad Monash	Un framework computacional distribuido de grid orientado a servicios.
Nimrod-G	Universidad Monash	Un agente intermediario de recursos de aplicaciones distribuidas.
GridPort	SDSC	Herramientas para la creación de portales computacionales.
Virtual Laboratory	Universidad Monash	Modelos moleculares para la creación de medicamentos.
NEESGrid	NCSA	Simulación y estudio de terremotos
Fusion Grid	Univerdad Princeton	Estudio de la fusión magnética.
IPG	NASA	Simulaciones aeroespaciales

Fuente: Grid Computing Info Centre de Rajkumar Buyya (2005).

Los grid se pueden utilizar para resolver problemas en campos distintos tales como química, biología, desarrollo de medicamentos, tomografía, minería de datos, análisis financiero, simulaciones nucleares, predicción del clima, entre otros.

Minoli (2005) también enumera los esfuerzos existentes de grid computing y se resume en las siguientes tablas.

Tabla N° 6 Consorcios de grid computing

Institución
Asia Pacific Grid
Australian Grid Forum
Content Alliance: About Content Peering
Distributed.net
eGrid: European Grid Computing Initiative
EuroTools SIG on Metacomputing
Global Grid Forum
Global Grid Forum (GGF)
Grid Computing Info Centre
GridForum Korea
IEEE Task Force on Cluster Computing
New Productivity Initiative (NPI)
Peer-to-Peer (P2P) Working Group
SETI@home
The Distributed Coalition

Fuente: Minoli (2005).

Tabla N° 7 Iniciativas científicas de grid computing – Aplicaciones

Aplicaciones
Access Grid

Aplicaciones

APEC Cooperation for Earthquake Simulation
Australian Computational Earth Systems Simulator
Australian Virtual Observatory
Cellular Microphysiology
DataGRID—WP9: Earth Observation Science Application
Distributed Proofreaders
DREAM Project: Evolutionary Computing and Agents Applications
EarthSystemGrid
Fusion Collaboratory
Geodise: Aerospace Design Optimisation
Globus Applications
GRid seArch & Categorization Engine (GRACE)
HEPGrid: High Energy Physics and the Grid Network
Italian Grid (GRID.IT) Applications
Japanese BioGrid
Knowledge Grid
Molecular Modelling for Drug Design
NC BioGrid
NEESgrid: Earthquake Engineering Virtual Collaboratory
Neuro Science—Brain Activity Analysis
NLANR Distributed Applications
OpenMolGrid
Particle Physics Data Grid
The International Grid (iGrid)
UK Grid Apps Working Group
US Virtual Observatory

Fuente: Minoli (2005).

Tabla N° 8 Iniciativas científicas de grid computing –Sistemas y aplicaciones P2P

Sistemas y aplicaciones P2P
Bayanihan Computing Group
Cetacean acoustic communication study
DALiWorld
Distributed Particle Accelerator Design
Distributed.net
Evolutionary@Home
FightAIDS@Home
Folderol—Bringing the Human Genome Project to the Desktop
Folding@home
Genome@home
Great Internet Mersenne Prime Search (GIMPS)
Life Mapper
Moneybee: Stock forecasts
SaferMarkets.org—Understanding and Predicting Market Volatility
Server-less Video on Demand
SETI@home: Search for Extraterrestrial Intelligence at Home
XPulsar@home

Fuente: Minoli (2005)

Tabla N° 9 Iniciativas científicas de grid computing – Grid de datos

Iniciativas de grid de datos
Datacentric Grid
DIDC Data Grid Work
EU DataGrid
GridPP
GriPhyN (Grid Physics Network)

Iniciativas de grid de datos

HEPGrid (High Energy Physics and Grid Networks)
 Particle Physics Data Grid (PPDG)
 Virtual Laboratory: Tools for Data Intensive Science on Grid

Fuente: Minoli (2005).

Además, existen iniciativas comerciales de grid computing. Éstas se resumen de la Tabla N° 10 a la Tabla N° 16.

Tabla N° 10 Sistemas intermediarios de grid computing

Sistema
Condor
Cosm P2P Toolkit
Globus Toolkit
Grid Datafarm
Gridbus
GridSim: Toolkit for Grid Resource Modeling and Scheduling
Simultation
Jxta Peer to Peer Network
Legion: A Worldwide Virtual Computer
PUNCH
Simgrid

Fuente: Minoli (2005)

Tabla N° 11 Sistemas de grid

Sistema
Amica
Bayanihan

Sistema
Compute Power Market
CrossGrid
DAMIEN
DIET
Echelon: Agent Based Grid Computing
Global Operating Systems
GridLab
Harness Parallel Virtual Machine Project
JAVELIN: Java-Based Global Computing
Management System for Heterogeneous Networks
MetaNEOS
MILAN: Metacomputing In Large Asynchronous Networks
MOBIDICK
MultiCluster
NeuroGrid
Poland Metacomputing
PUNCH—Network Computing Hub
XtremWeb

Fuente: Minoli (2005).

Tabla N° 12 Calendarizadores de grid

Calendarizador
AppLeS
Computing Centre Software (CCS)
Condor/G
DISCWorld
NetSolve

Nimrod/G Grid Resource Broker
 SILVER Metascheduler
 ST-ORM

Fuente: Minoli (2005).

Tabla N° 13 Portales de grid

Portal
ActiveSheets
Enginframe
G-Monitor
Grid Enabled Desktop Environments
Gridscape
Interactive Control and Debugging of Distribution- IC2D
Lecce GRB Portal
NLANR Grid Portal Development Kit
SDSC GridPort Toolkit
UNICORE—Uniform Interface to Computing Resources

Fuente: Minoli (2005).

Tabla N° 14 Ambientes de programación grid

Ambiente
Albatross: Wide Area Cluster Computing
Environments Cactus Code
GAF3J—Grid Application Framework for Java
GrADS: Grid Application Development Software Project
Jave-based CoG Kit

MetaMPI—Flexible Coupling of Heterogenous MPI Systems
 Nimrod—A tool for distributed parametric modeling
 Ninf
 ProActive PDC
 REDISE—Remote and Distributed Software Engineering
 Virtual Distributed Computing Environment

Fuente: Minoli (2005).

Tabla N° 15 Herramientas de monitoreo y de predicción

Herramienta
NetLogger Weather Service
Forecasting Remos

Fuente: Minoli (2005).

Tabla N° Facilidades de grid y de desarrollo

Facilidad	
Alliance Grid	TeraGrid
Developments Asia Pacific Bioinformatics Network	ThaiGrid
EuroGrid	The Alliance Virtual Machine Room
GrangeNet	The Distributed ASCI Supercomputer (DAS)
G-WAAT	World Wide Grid (WWG)
I-Grid	
Internet Movie Project	
Irish Computational Grid (ICG)	
Kerala Education Grid	
LHC Grid	
Micro Grid	
N*Grid Korea	

NASA Information Power Grid (IPG)	
Nordic Grid	
NPACI: Metasystems	
OurGrid	
Polder Metacomputer	

Fuente: Minoli (2005).

Tabla N° 16 Compañías comerciales de grid y P2P

Compañía	
Avaki	Popular Power
CapCal	Powerllet
Centrata	ProcessTree
DataSynapse	Sharman Networks Kazza
Distributed Science	Sun Gridware Sysnet Solutions
Elepar	Ubero United Devices
EMC/VMWare	Veritas Xcomp
Entropia.com	
Grid Frastructure	
GridSystems	
Groove Networks	
HP	
IBM	
Intel	
Jivalti	
Mind Electric	
Mithral	
Mojo Nation	
NewsToYou.com	

NICE, Italy	
Noemix, Inc.	
Oracle	
Parabon	
Platform Computing	

Fuente: Minoli (2005).

H. Modelos económicos aplicados en problemas computacionales

Si se toma como base los modelos económicos aplicados a los mercados humanos, entonces se pueden aplicar sus principios básicos para regular los recursos de una arquitectura de grid computing.

De forma general, el problema de administración de recursos computacionales puede ser analizado en términos de la economía, específicamente el uso de los mecanismos de comercialización y precio. Al respecto, Buyya (2000) plantea que existen dos puntos importantes que debe contemplar este modelo económico: ¿Cómo se puede maximizar la utilidad de los dueños de los recursos? ¿Cómo pueden los consumidores resolver sus problemas con un costo mínimo?

La creación de la mayoría de los sistemas grid existentes ha sido motivada por el bien común, la investigación, el reconocimiento y la fama, entre otros motivos. Sin embargo, en los últimos años, el uso de grid computing de modo comercial ha sido popular. Por ejemplo: Oracle, Sun, HP, Entropía y Parabon ofrecen sistemas basados en esta tecnología.

Los sistemas de administración de recursos de grid tienen que negociar de forma dinámica los mejores recursos utilizando métricas como precio, desempeño y agenda de calendarización; todo esto con el fin de cumplir con los requerimientos establecidos por el usuario.

La aplicación de modelos económicos en grid computing ofrece ciertos beneficios. Entre ellos encontramos los siguientes:

- Permite regular la oferta y demanda de recursos.
- Permite que los primeros problemas por ejecutar sean los críticos, ya que cuando un usuario tiene un inconveniente no crítico se beneficia ejecutándolo más tarde debido a un modelo de precios reducidos.

1. Características que debe poseer un modelo económico aplicado a la informática distribuida

En los ambientes de grid computing, los productores (dueños de los recursos) y los consumidores (usuarios de los recursos) tienen metas, objetivos, estrategias y patrones de oferta/demanda distintos. Administrar ese ambiente y tratar de optimizar todos los recursos es una tarea casi imposible. Generalmente, se han utilizado políticas centralizadas que necesitan un conjunto de información específica sobre cada nodo y una infraestructura igual en todos los sitios. D. Ferguson, C. Nikolaou, J. Sairamesh y Y. Yemini (1996) plantean que es imposible definir una política estática de administración de recursos aplicable a todo el sistema grid.

Debido a esto, es necesario utilizar un enfoque económico para la administración de recursos dentro de ambientes grid computing. El enfoque económico posee las bases para la administración de la descentralización y heterogeneidad del grid, ya que estos factores

están presentes en las economías humanas. Los modelos económicos de competitividad tienen algoritmos, políticas y herramientas que se aplican en el manejo de recursos (compartir, asignar) de los sistemas de grid computing.

Los modelos pueden basarse en intercambios o en precios. En el modelo de intercambios, todos los participantes deben poseer recursos y negociarlos por medio de intercambios. Por ejemplo, cambiar tiempo de CPU por almacenamiento; cambiar horas de procesamiento. En el modelo basado en precios, los recursos tienen un costo que se basa en factores del sistema económico: oferta, demanda, valor, bienestar.

En los sistemas de grid computing actuales, se han utilizado estrategias convencionales para la administración de los recursos. En ellos, los componentes de calendarización deciden las tareas por ejecutar y en cuáles recursos. Para hacerlo, utilizan funciones de costos programadas dentro de los parámetros centrales del sistema grid. Entre estos sistemas grid se encuentran Legion, NetSolve, Condor y XtremWeb. Se caracterizan por querer mejorar el desempeño del sistema ejecutando las tareas en el tiempo más corto posible y no se preocupan por la aplicación como un todo, no toman en cuenta el precio del recurso a pesar de que el precio de los recursos varía dependiendo del momento de ejecución. Se debe recordar que el usuario quiere realizar sus tareas pero sin pagar el precio máximo: lo ideal es negociar un precio con base en la demanda, valor, prioridad y presupuesto disponible.

A diferencia de los modelos estáticos, en un modelo económico, las decisiones de calendarización de tareas se toman de forma dinámica y utilizan los requerimientos establecidos por el usuario: tiempo máximo de ejecución, hora, presupuesto disponible. El modelo económico se le cobra al usuario por el servicio que consume y no por el hardware que utiliza. El factor principal de un modelo económico se basa en que el precio de los bienes se establece con base en la demanda y oferta: por lo tanto, en un ambiente de grid

computing los usuarios son entes dentro del mercado que compiten con otros usuarios para utilizar los recursos.

Cuando se introduce el concepto de modelo económico dentro de grid computing, se debe tomar en cuenta aspectos tales como comercialización de recursos y la calidad de servicio. Específicamente, el modelo económico de grid computing debe poseer las siguientes características:

- Un modelo para establecer el valor de los recursos.
- El modelo ha de tener mecanismos para regir los precios y su publicación.
- Los protocolos de negociación deben ser contemplados dentro del modelo.
- Tiene que existir un lugar en el cual se puedan localizar las entidades participantes en el grid, para que los usuarios puedan buscar los recursos requeridos.
- La calidad del servicio debe ser tomado en cuenta durante el proceso de calendarización de tareas.
- Se introduce el concepto de un agente intermediario, quien puede ser utilizado por los consumidores para localizar los recursos que necesita. El papel del agente intermediario es tomar los requerimientos del consumidor -tipo de recurso, presupuesto, restricciones de tiempo, prioridades- y encontrar al proveedor que cumpla con las exigencias del consumidor.

2. Modelos de determinación de precios

El esquema de precios más sencillo es una tarifa fija; sin embargo, cuando se involucran parámetros de calidad de servicio este esquema es inadecuado. Se debe a que estos varían dependiendo de las aplicaciones y de cuando se ejecutan los procesos. G. Tesauro y J. Kephart (1998) al igual que J. Sairamesh and J. Kephart (1998) han propuesto varios modelos para establecer los precios en ambientes distribuidos. Se basan en distintos parámetros, entre los cuales se encuentran los siguientes:

- Modelos económicos competitivos, por ejemplo las subastas.
- Modelo de precio fijo, en el cual se aplica el mismo costo para los todos los consumidores.
- Periodo de tiempo de uso y su duración: Considera horas de alta y baja demanda.
- Oferta y demanda.
- Basado en predicciones: Se utiliza un modelo para predecir las respuestas de los competidores.
- Basado en la lealtad de los clientes. Por ejemplo, los programas de acumulación de millas de las líneas aéreas.
- Datos históricos.
- Contratos de nivel de servicio con los proveedores de servicios.
- Basado en cronogramas.
- Basado en compras al por mayor de servicios.
- Con base en acuerdos: Los cuales los sindicatos de comercialización establecen la estructura de precios.
- Dependiendo del área de aplicación se pueden ofrecer precios menores. Por ejemplo, las aplicaciones académicas o de investigación pueden tener precios más bajos que las aplicaciones comerciales.

El doctor Sairamesh (1998) también plantea diferentes estrategias de precios, las cuales- dependiendo de las exigencias de los consumidores- son clasificadas en dos categorías: enfocadas en calidad y en precio.

En el enfoque de calidad, los consumidores requieren poder computacional ilimitado; mientras que en el de precio necesitan poco poder computacional. Cuando se tiene un mercado enfocado en la calidad, todas las estrategias de precios intentan lograr un equilibrio de precios, el cual se predice utilizando un análisis basado en la teoría de juegos.

Sin embargo, cuando se posee un mercado enfocado en el precio, las estrategias de precios se basan en una guerra de estos.

Como puede deducirse, estas estrategias pueden ser aplicadas al grid computing. Únicamente, se debe tomar en cuenta que los proveedores de recursos han de beneficiarse por medio de su eficiente utilización y los consumidores deben tener la posibilidad de analizar el costo beneficioso de las ofertas en el mercado.

Las aplicaciones de los usuarios poseen distintos requerimientos de poder computacional dependiendo de su naturaleza, algunas necesitan mucho uso de procesador y otras de uso intensivo de almacenamiento. Con base en este planteamiento, es preciso establecer sobre cuáles recursos se establecerá una tarifa; estos pueden ser memoria, los procesadores, el espacio en disco duro, entre otros componentes.

3. Modelos económicos aplicados a la informática distribuida

Ferguson, Nikolaou, Sairamesh, y Yemini (1996) establecen los puntos que deben tomarse en cuenta al aplicarse modelos económicos en la resolución de problemas:

- ¿De qué forma los agentes intermediarios realizan la demanda de recursos? Para responder a esta pregunta es necesario establecer un mecanismo que rija las preferencias del agente al elegir entre varios recursos. De la misma manera, se puede establecer una función de sensibilización de utilidad, la cual ayuda a los agentes en la elección de recursos.
- ¿Cómo se ajustan los precios en la economía? En un modelo económico, la eficiencia de asignación de recursos se logra cuando la demanda es igual a la oferta bajo cierto criterio de precios.

- ¿Cuáles son los mecanismos de precios que utilizan los productores? Dependiendo de los mecanismos de precios, los consumidores podrían ser atraídos o no por un recurso determinado.
- ¿De qué forma los productores aseguran un precio a los agentes intermediarios? Este es un aspecto fundamental, ya que los precios podrían variar desde el momento cuando el agente elige un proveedor hasta que empieza a utilizarlo, dependiendo de las condiciones del mercado.
- ¿Cuáles son los mecanismos de comunicación para que los consumidores reserven un recurso de un productor determinado?
- ¿Cuáles son las estrategias que utilizan los agentes intermediarios para elegir el proveedor adecuado?

Buyya, Stockinger, Giddy y Abramson (2004) han planteado que el uso de los modelos económicos es factible en un ambiente de informática distribuida; distintos modelos pueden ser aplicados. En esta sección, se enumeran los modelos sugeridos por estos autores y cómo pueden regir el comportamiento de oferta y demanda en la informática distribuida.

Así, los productores publican sus ofertas en el mercado y los consumidores ingresan sus necesidades o demanda en el mismo mercado. Las especificaciones de oferta y demanda se realizan utilizando un lenguaje de especificación común; por medio de él se establecen los costos, los horarios disponibles para procesamiento, los límites de tiempo en el cual la información debe ser procesada, la cantidad de presupuesto disponible para procesar la información, entre otros parámetros.

a) Modelo de oferta y demanda

En este modelo los productores establecen un precio por sus recursos y facturan a los consumidores con base en la cantidad de recursos consumidos. Las políticas de precios pueden establecer distintos parámetros tales como tarifa fija o tarifa variable. En el caso de

la tarifa variable, cambia según el comportamiento de la demanda: si los recursos ofertados son limitados y la demanda es alta, entonces el precio tiende a subir; en caso contrario, la tarifa tiende a bajar. Sin embargo, se debe establecer un piso hasta el cual baja el precio para que los productores no incurran en pérdidas descontroladas, pues si esto ocurre el mercado podría colapsar.

Se debe considerar que en el caso de una tarifa fija, esta se mantiene invariable sin importar la calidad del servicio. En este caso, la calidad del servicio podría ser afectada si existe mucha demanda y los recursos de los productores son sobre explotados; de forma tal, que se brinda el servicio, pero cae su calidad.

Según Buyya (2005), el esquema de precio en un ambiente de oferta y demanda puede basarse en los siguientes modelos: tarifa fija, tarifa variable, con base en el tiempo de uso de los recursos, tarifa por suscripción, basado en la oferta y demanda -el precio puede variar hasta que se encuentre el punto de equilibrio del mercado.

Cuando los productores publican sus precios, incluyen los costos de producción así como un margen de ganancia; de esta forma, los consumidores eligen cual oferta se ajusta más a sus necesidades: puede ser reducir costos, entre otros. La decisión final del consumidor depende del valor que le asigne a las ofertas, el cual puede ser representado por la siguiente función:

Valor de la oferta = calidad de los recursos ofertados por el productor, costo de los recursos, calidad del servicio, cantidad de demanda sobre los recursos, preferencias por un productor en particular.

Después de que el consumidor o el agente intermediario ha puesto un valor a cada una de las ofertas, se procede a elegir una y se inicia el proceso de envío de información e intercambio de resultados.

Las políticas de precios en este modelo toman en cuenta aspectos tales como uso de ciclos de procesamiento, almacenamiento requerido, uso de la red, entre otros. Un ejemplo, de implementación de este modelo es el proyecto Nimrod desarrollado por Buyya Rajkumar (2002), en el cual se presenta un lenguaje para especificar las ofertas y demandas, las cuales son publicadas en del mercado del gris. Los requerimientos del usuario incluyen, por lo menos, los siguientes parámetros: tiempo máximo para el procesamiento de la información y el presupuesto. Nimrod utiliza un agente intermediario para comunicar a los consumidores y a los productores. El proceso que lleva a cabo este agente es el siguiente:

- Identifica a los productores o proveedores de recursos.
- Identifica los recursos potenciales con base en las ofertas y determina sus precios.
- Selecciona la oferta adecuada con base en los objetivos del consumidor: menores costos o tiempo de procesamiento más rápido. Utiliza heurísticos para asignar cada uno de los trabajos de procesamiento de información con cada uno de los recursos incluidos en la oferta seleccionada.
- Utiliza los recursos y, finalmente, cancela su precio.

b) Publicación de ofertas

El modelo de publicación de ofertas es similar al de oferta y demanda; sin embargo, se caracteriza por publicar ofertas por parte de los productores con el fin de atraer nuevos consumidores o para que los consumidores existentes interactúen con recursos utilizados por ellos con anterioridad.

En este caso, los agentes intermediarios no negocian un precio con los productores; se acogen a las ofertas publicadas, las cuales -en la mayoría de los casos- incluyen precios menores al precio de lista. Sin embargo, estas ofertas –generalmente- incluyen restricciones

en cuanto al horario de uso de los recursos, a la cantidad de recursos disponibles y a la validez de la oferta.

Las tareas dentro del modelo de ofertas incluyen las mismas que se han establecido para el modelo de oferta y demanda y, adicionalmente, se agregan las siguientes:

- Los productores publican sus ofertas en el mercado.
- Los agentes intermediarios revisan las ofertas y determinan si cumplen con los requerimientos de alguno de sus consumidores.
- Al identificarse una oferta que cumple con los requerimientos, se establece una comunicación con el productor para determinar si el recurso puede reservarse.
- De aquí en adelante, se continúa el mismo proceso que en el modelo oferta y demanda.

c) **Modelo de regateo**

En el modelo de regateo, los agentes intermediarios negocian con los productores los términos relacionados con el precio y el tiempo de utilización de los recursos. La meta del agente intermediario es obtener el precio más bajo y utilizar los recursos al máximo tiempo posible. En este caso, tanto los agentes intermediarios como los productores tienen diferentes funciones-objetivo; por tanto, es necesario el inicio de un proceso de negociación hasta que las dos partes queden satisfechas.

El mecanismo básico es el siguiente: el agente intermediario inicia ofreciendo un precio relativamente bajo; por otra parte, el productor realiza una oferta que contempla el precio más alto. En este momento, se inicia el proceso de negociación en el cual se varían los precios hasta que se encuentra un punto de equilibrio o de acuerdo o -más bien- una de las partes decide no seguir negociando y retira su oferta.

Durante el proceso de negociación, el agente intermediario utiliza los requerimientos establecidos por el usuario; por ejemplo, el presupuesto y el plazo en el cual la información debe estar procesada.

En este proceso, los agentes intermediarios podrían tomar una estrategia de buscar los recursos más baratos. Esto podría provocar que se descarten máquinas caras pero rápidas. En este caso, el mercado debería reaccionar y los dueños de los recursos bajar sus precios para reducir la cantidad de recursos ociosos.

El modelo de regateo es utilizado donde el modelo de oferta y demanda no está regido por un conjunto de políticas bien definidas. Esto provoca que los consumidores o agentes intermediarios bajen los precios mediante la presión de la negociación.

d) Modelo de subastas

Este modelo se basa en un patrón negociativo de uno a muchos: un productor -proveedor de servicios dueño de los recursos- negocia con muchos consumidores -compradores de recursos de procesamiento de información- hasta acordar un precio adecuado para el recurso que se está subastando. La negociación no es realizada por el productor directamente; la realiza el subastador, el cual ha recibido un conjunto de directrices que regirán la subasta y pueden ser establecidas tanto por el productor como por los compradores.

En general, los roles presentes en este modelo son los dueños de los recursos, los compradores y los subastadores o intermediarios. En un ambiente de informática distribuida, se utiliza un protocolo para determinar el precio de venta de un recurso. El proceso de subasta consta de los siguientes pasos:

- a. Los productores anuncian sus servicios y esperan las ofertas de compra (bids en inglés).
- b. Los agentes intermediarios hacen sus ofertas (dependiendo del tipo de subasta podrían ver las ofertas hechas por los otros agentes).
- c. Se repite el paso b hasta que se encuentre un ganador o cuando nadie esté dispuesto a ofrecer más por el servicio en subasta y la oferta más alta actual no cumple con los requerimientos de precio mínimo establecido por el dueño del recurso.
- d. El productor brinda el servicio al ganador.
- e. El consumidor utiliza el recurso que ganó durante la subasta.

Thomas Sandholm (1999) establece que las subastas pueden ser abiertas o cerradas. En las abiertas, los consumidores pueden observar las ofertas hechas por los demás y, si lo desean, actualizar su oferta inicial; incluso, el proveedor podría variar el precio que está exigiendo por el recurso. En las subastas cerradas, los consumidores no pueden ver las ofertas hechas por los otros agentes interesados.

Sandholm clasifica los protocolos de las subastas en los siguientes tipos:

- Subasta inglesa (primer precio, puja abierta).
- Subasta de precio sellado.
- Subasta descendente.
- Subasta Vickrey.

En la subasta inglesa, cada participante puede aumentar su oferta cuando lo desee. La subasta finaliza si ningún participante está dispuesto a aumentar su oferta y gana quien realice la oferta más alta. La estrategia del participante se basa en el valor máximo dispuesto a ofrecer, las estimaciones de las ofertas de los otros y las ofertas que se han realizado anteriormente en casos similares. En una subasta inglesa privada, la estrategia del

participante es siempre ofrecer un poco más que la oferta actual más alta, hasta llegar al límite del valor máximo por ofrecer. En una subasta inglesa correlativa, se aumenta el precio a una tasa constante. En algunas ocasiones, los participantes pueden indicar que salen de la subasta, lo cual le brinda más información a los otros sobre el estado de la subasta y con quién están compitiendo por los recursos.

En la subasta de precio sellado, cada participante entrega su oferta sin conocer las otras ofertas: gana aquel que haya ofrecido el precio más alto.

En la subasta descendente, el dueño del recurso o productor inicia solicitando un precio y lo va bajando hasta que uno de los compradores adquiera el recurso al precio actual. En este modelo, el productor busca un mercado y empieza a ofrecer su servicio a un precio un poco más elevado que el promedio del mercado, después lo baja continuamente con la esperanza de que el recurso sea adquirido por un consumidor.

En la subasta Vickrey, cada participante realiza una oferta cerrada sin conocer las otras ofertas. La gana con el valor más alto, pero el precio del servicio está dado por la segunda oferta más alta.

Todos los modelos de subastas presentados tienen la desventaja de que los participantes podrían ponerse de acuerdo para ofrecer menos del valor real del recurso, lo cual implica pérdidas para los dueños de los recursos.

e) Modelo oferta / contratista

Este modelo es uno de los más utilizados entre los procesos que necesitan algún tipo de negociación. El ambiente distribuido fue propuesto por R. Smith y estudiado por Zafeer Alibhai (2006). Se ha utilizado para modelar el proceso de negociación de bienes y

servicios; de forma tal, que ayuda a encontrar al proveedor adecuado con base en los requerimientos del comprador.

En este modelo, el consumidor le indica al agente intermediario sus requerimientos y este agente toma el rol de administrador; los recursos -candidatos para resolver los requerimientos del usuario- son los contratistas potenciales.

El proceso llevado a cabo por el administrador es el siguiente:

- Se anuncian los requerimientos del usuario y se invita a hacer ofertas para cumplir esos requisitos.
- Los proveedores interesados revisan los requerimientos y responden con su oferta.
- El agente intermediario evalúa y premia a la oferta que sea la más apropiada.
- El agente intermediario y el contratista o proveedor se comunican para coordinar el uso del recurso.

El anuncio de los requerimientos incluye la siguiente información: ubicación geográfica del consumidor interesado, requerimientos de operación -tipo de sistema operativo, memoria, espacio en disco, velocidad de la red-, descripción del servicio requerido; opcionalmente, podría incluir el precio que el usuario está dispuesto a pagar por el servicio, detalle de la información que deben contener las ofertas y cuando se dejarán de recibir ofertas.

Desde el punto de vista del contratista o proveedor, el proceso es el siguiente:

- Recibe los anuncios del consumidor.
- Evalúa la capacidad de sus servicios para determinar si puede hacerle frente a los requerimientos del usuario.
- Responde con una oferta.
- Brinda el servicio si su oferta es la ganadora.

En este modelo, si un productor no puede brindar un servicio adecuado, el agente intermediario puede buscar a otro productor. Sin embargo, podría ocurrir que se escoja a un productor no muy veloz, pues durante la etapa de apertura de ofertas el productor más rápido estaba ocupado y no podía atender la tarea en el plazo especificado.

Una de las principales desventajas de este modelo radica en que el administrador no está obligado a informar a los potenciales participantes que una elección ya se ha hecho, lo cual podría provocar que los interesados continúen enviando ofertas innecesarias.

Igualmente, se podría suceder que el administrador no recibe ninguna oferta, ya sea porque todos los potenciales contratistas están ocupados o porque no existe alguno interesado. Para manejar este tipo de casos, el administrador puede solicitar una respuesta rápida a los proveedores de servicios y estos contestar con algunos de los siguientes mensajes: ocupado, elegible, no elegible o no interesado. De esta forma el administrador determinaría si su cartel de requerimientos tiene o no potenciales contratistas. En caso de que ningún proveedor o productor esté interesado, el administrador podría variar los términos del cartel de requerimientos; por ejemplo, se podrían variar el presupuesto o los plazos de tiempo para procesar la información.

f) Modelo de uso compartido proporcional de recursos

En este caso, a cada participante se le asigna un porcentaje de los recursos. Este resulta proporcional al valor de su oferta inicial por el recurso comparado con las ofertas de los participantes. Por ejemplo, si dos consumidores desean utilizar igual recurso al mismo tiempo y el consumidor A ofrece 10 US\$ y el consumidor B ofrece 40 US\$, el consumidor A recibe $1/5$ de acceso al recurso mientras que consumidor B recibe $4/5$.

g) Monopolios

En el caso en que un productor de servicios domine un mercado y sea el único proveedor, se presenta el caso de monopolio. En este ambiente, los consumidores no pueden realizar tipo alguno de presión para modificar los precios del proveedor. Los agentes intermediarios- únicamente- pueden consultar los precios publicados, pero no iniciar negociación alguna.

I. Negocios de Grid Computing en los Estados Unidos de América

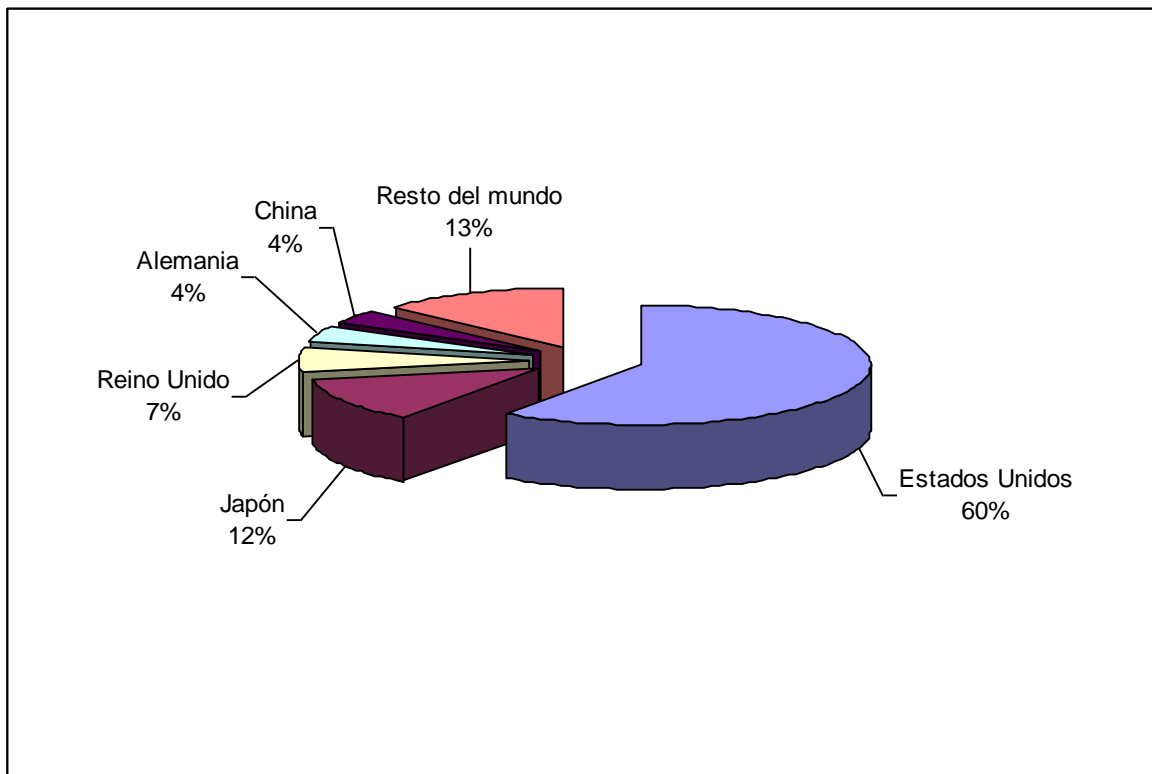
1. Introducción

En este capítulo, se presenta el estado actual de la tecnología de informática distribuida en los Estados Unidos de América. Primero, la distribución de estos proyectos en el ámbito mundial y luego, específicamente, en los Estados Unidos.

Inicialmente, se describen los sistemas de administración de recursos así como empresas comerciales que brindan servicios de informática distribuida. Después, se presenta un estudio sobre el impacto de la informática distribuida sobre la economía de las empresas. En conclusión, se describen las inversiones realizadas en esta tecnología en un sector de la industria de los Estados Unidos de América.

2. Iniciativas de grid computing

Aunque las iniciativas de informática distribuida se están presentando en distintas partes del mundo, la mayoría de los desarrollos se han originado en Estados Unidos de América; cerca del 60% del total de poder computacional mundial se encuentra en ese país, lo cual se puede observar en el Gráfico N° 1. Por este motivo, en esta sección se incluye una selección de los principales proyectos estadounidenses de informática distribuida y que han sido los más sobresalientes.

Gráfico N° 1 Distribución del poder computacional mundial

Fuente: High Performance Computing University of Tennessee, 2004.

Si se realiza una búsqueda en Internet que liste los proyectos de informática distribuida existentes, este inventario contendrá cientos de proyectos. Por lo tanto, es imposible enumerar cada uno de ellos. A continuación, se incluirán aquellos proyectos considerados líderes en el campo de informática distribuida.

En este apartado, se incluye la descripción de productos que permiten construir plataformas de informática distribuida: Data Synapse, Avaki, Entropía, Globus, United Devices y Platform Computin. Además, los programas calendarizadores de tareas: Condor, LoadLeveler y PBS.

En la siguiente tabla, se resumen las principales iniciativas de informática distribuida y el modelo económico que rige su operación.

Tabla N° 17 Sistemas de computación distribuida que utilizan modelos económicos

Sistema	Modelo económico	Plataforma	Observaciones
Mariposa Universidad de Berkeley	Ofertas (Protocolo Contract-Net). Precios basados en la carga y la información histórica.	Base de datos distribuida	Soporta procesamiento de consultas basadas en presupuesto y administración del almacenamiento.
Mungi. Universidad de New South Wales	Mercado detallista. Se alquila el espacio de almacenamiento, el cual aumenta a medida que el espacio disminuye, lo cual fuerza a los usuarios para que liberen el espacio no necesitado.	Servidores de almacenamiento	Suporta objetos de almacenamiento basados en cuentas de banco, de lo cual se deduce el alquiler del almacenamiento utilizado por los objetos.
Nimrod-G. Monash University	Soporta modelos económicos tales como detallistas, mercados mayoristas y utilizan el protocolo Contract-Net para establecer	Un grid de computadores distribuidas (estaciones de trabajo y clusters)	Soporta algoritmos de calendarización basados en restricciones de presupuesto y tiempo de ejecución, para aplicaciones

Sistema	Modelo económico	Plataforma	Observaciones
	los precios.		que se ejecutan en recursos distribuidos dependiente de su costo, poder de procesamiento y disponibilidad y los requerimientos de calidad de servicio de los usuarios.
PopCorn Hebrew University	Subastas. La mayor oferta obtiene el acceso al recurso y transfiere los créditos de la cuenta del comprador hacia la cuenta del vendedor.	Navegadores de Internet	Se necesita utilizar el API de Popcorn con el propósito de especificar un presupuesto para cada uno de sus módulos.
JavaMarket	Basado en calidad de servicio de un mercado computacional	Navegadores de Internet	Una persona puede vender ciclos de CPU cuando entra al portal de sitio y permite la ejecución de tareas.
Enhanced Hebrew University MOSIX	Mercado detallista: el costo de los recursos de cada nodo es conocido.	Clusters de computadoras Linux	Soporta la migración de procesos de forma, tal que se mantenga bajo de costo de ejecución

Sistema	Modelo económico	Plataforma	Observaciones
			de los trabajos.
JaWS University of Crete	Ofertas	Navegadores de Internet	Similar a Popcorn
Xenoservers University of Cambridge	Ofertas: recursos compartidos proporcionalmente.	Computadora sencilla	Permite la ejecución de código no confiable.
D'Agents Dartmouth College	Ofertas: recursos compartidos proporcionalmente.	Computadora sencilla o agentes móviles	La oferta es proporcional al beneficio.
Rexec Anemone UC Berkeley	Ofertas: recursos compartidos proporcionalmente.	Clusters Almacenamiento en red	Los usuarios asignan un valor de utilidad a su aplicación y el sistema ubica los recursos proporcionalmente.
Mojo Nation. Autonomous Zone Industries, CA	Sociedad basada en créditos: los contribuidores ganan créditos cuando comparten almacenamiento y los gastan cuando sea necesario	Almacenamiento en red	Es una red comunitaria de contenido compartido.
Spawn Xerox PARC	Subastas del segundo precio	Red de estaciones de trabajo. Cada una ejecuta una tarea por un tiempo determinado.	Soporta la ejecución de trabajos en paralelo, los cuales pueden formar una jerarquía de procesos

Sistema	Modelo económico	Plataforma	Observaciones
			que se expande y encoje dependiendo del costo de los recursos.
Supercomputing Centre University of Manchester	Mercado detallista basado en un modelo de prioridades: cobran por CPU, memoria, almacenamiento y servicios de soporte.	Supercomputadoras y servidores de almacenamiento	Cualquier aplicación puede utilizar este servicio y su calidad es proporcional a la prioridad del usuario y mecanismos de calendarización.

Fuente: Buyya (2002).

3. Sistemas de administración de recursos

Dependiendo del enfoque con el cual es desarrollado el grid, los Sistemas de Administración de Recursos del Grid (SARG) se clasifican en Grid Computacionales, Grids de Datos y Grids de Servicios. En el año 2002- K. Krauter, R. Buyya y M. Maheswaran definen la taxonomía de los sistemas de administración de recursos, la cual los clasifica según el sistema del grid, organización de las computadoras, recursos y métodos de calendarización.

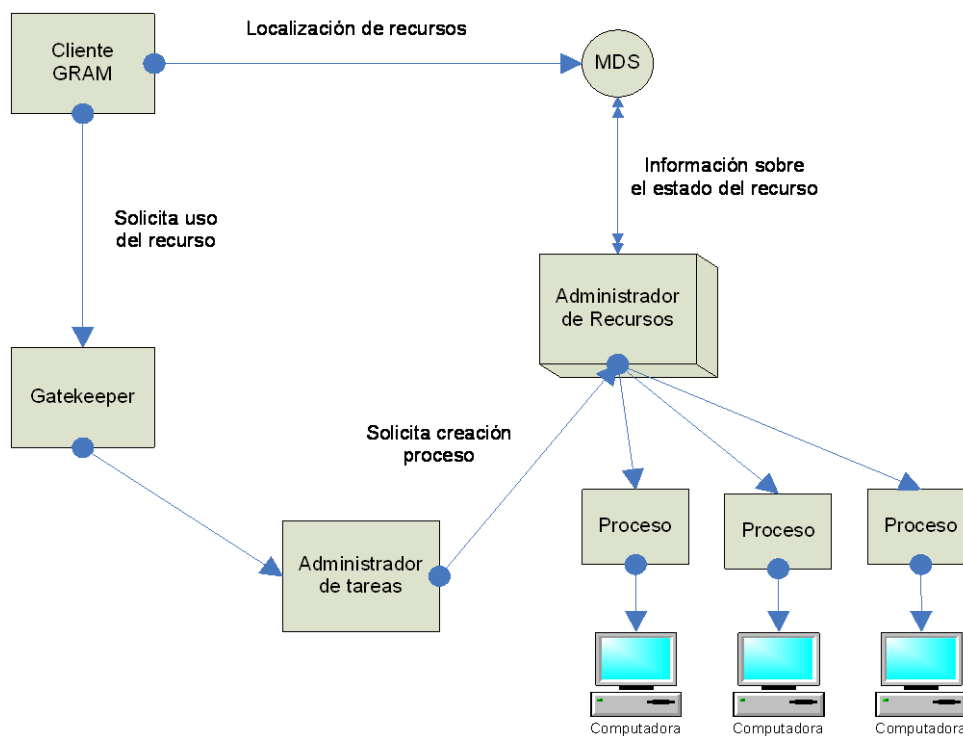
Entre los sistemas actuales de Grid Computing, la solución dada para la concretar quién tiene acceso a los recursos y a qué precio, ha llevado a definir distintos modelos estáticos, los cuales establecen las reglas de acceso a los recursos. En esta sección, se presentan algunos de los modelos estáticos de asignación de recursos existentes.

a) Globus Toolkit

El Globus Toolkit es un conjunto de servicios, el cual facilita la ejecución de aplicaciones en grid. Permiten que se agreguen y administren aplicaciones, que se descubran recursos y se muevan datos en el grid.

La utilización de los servicios de Globus requiere la instalación de una infraestructura de soporte determinada. Cada recurso se administra por un objeto llamado Globus Resource Allocation Manager (GRAM), el cual es responsable de monitorear el estado del recurso. Su funcionamiento se muestra en la siguiente figura.

Figura N° 31 Elementos de un calendarizador de procesos y recursos



Fuente: Buyya (2000).

Las peticiones del consumidor son recibidas por el Gatekeeper -agente que controla el acceso a los recursos- el cual consulta al Globus Security Infrastructure (GSI). Este

servicio permite la autenticación del consumidor dentro del grid. Después, el GRAM verifica que el usuario puede ejecutar el recurso en cuestión. Si el usuario tiene acceso permitido, se crea un administrador de la tarea, el cual es responsable de iniciar y monitorearla.

A pesar de que Globus resuelve satisfactoriamente los aspectos operacionales, no lo realiza con los problemas administrativos del grid. Con su modelo presente, es necesario establecer una negociación manual entre los consumidores y los proveedores: esto limita la escalabilidad del mecanismo.

b) MyGrid

MyGrid es una plataforma de ejecución de aplicaciones paralelas dentro de un grid. Los niveles de aplicación se definen como tareas independientes que no necesitan comunicación entre ellas durante su ejecución.

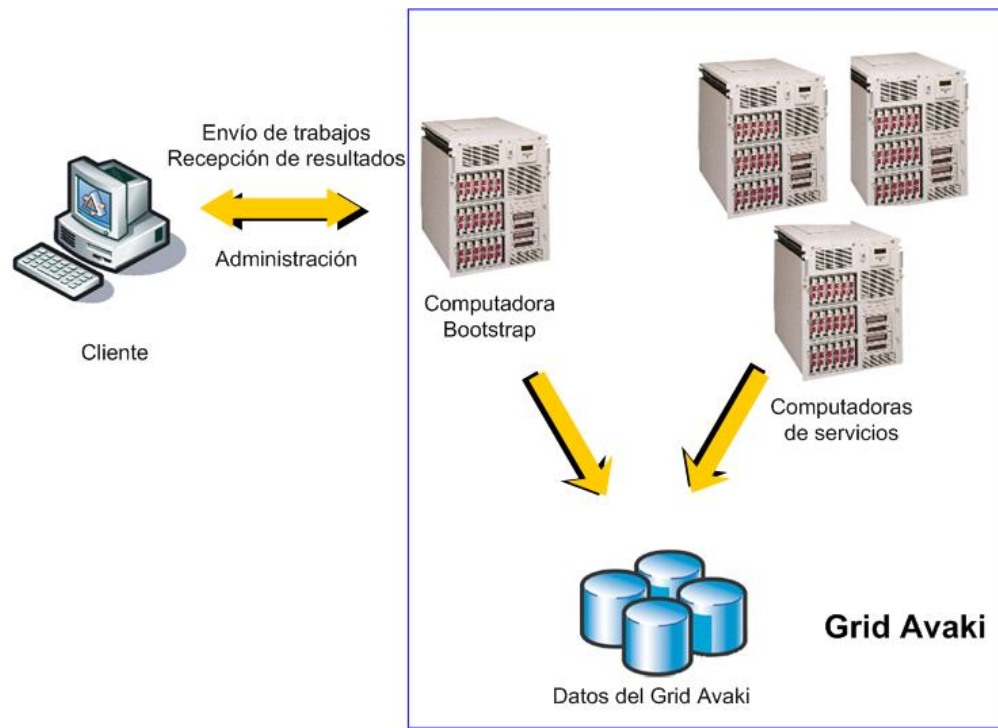
Desde el punto de vista gerencial, es responsabilidad del consumidor obtener el acceso a los recursos que desea utilizar.

c) Avaki

Avaki es un producto de software comercial que permite crear plataformas de informática distribuida de tipo grid de cálculo computacionales y grid de datos. Los sistemas operativos -en los cuales se pueden ejecutar- incluyen Windows, Linux, Tru64, AIX, Solaris e IRIS.

En la Figura N° 32, se muestra la plataforma general de Avaki, la cual incluye un cliente que envía los trabajos hacia el grid y cada uno de los recursos está configurado para atender las tareas recibidas. Los usuarios ingresan en el sistema y envían las tareas por ejecutar.

Figura N° 32 Plataforma Avaki



Fuente: www.avaki.com.

d) DataSynapse

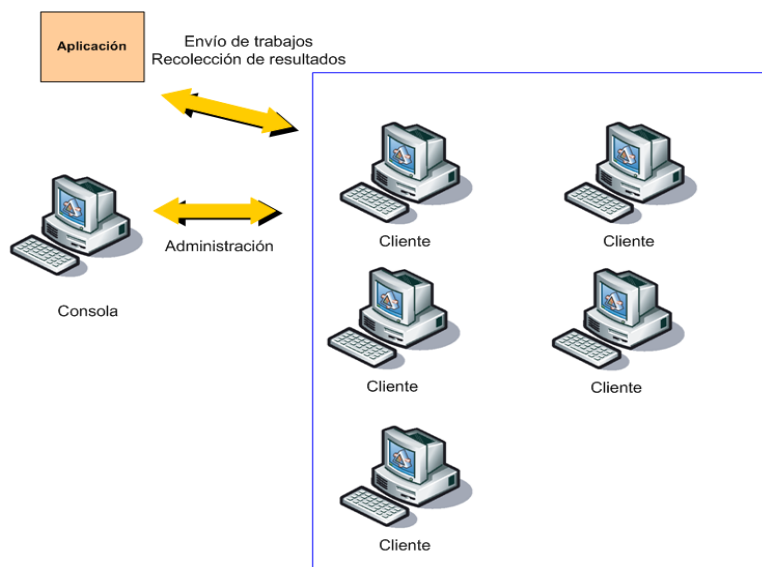
DataSynapse es una empresa privada ubicada en Nueva York. Se dedica a proveer servicios de informática distribuida; el producto que ofrecen es LiveCluster. Las plataformas sobre las cuales puede ejecutarse las tareas incluyen Linux, Solaris y Windows.

La infraestructura general de LiveCluster está compuesta por un servidor, motores y controladores y se observa en la Figura N° 33. Los controladores se encargan de coordinar el traspaso de información de tareas entre las aplicaciones y el núcleo de servidores del grid. Las funciones de calendarización y de administración están a cargo del servidor. Los motores pueden ser instalados en servidores o en estaciones de trabajo, dependiendo del poder computacional que se desee configurar. El papel fundamental de los motores es identificar cuándo un computador está inactivo. Cuando lo hace, le envía nuevas tareas que

deben ser ejecutadas. Este es un modelo no invasivo de procesamiento de información, pues si el motor está ejecutando una tarea y el usuario del computador empieza a utilizarlo, entonces la tarea es suspendida y se envía en su estado actual hacia un computador disponible. De esta forma, el usuario puede utilizar la computadora mientras que el procesamiento calendarizado continúa su ejecución en un computador distinto.

Todo el sistema puede ser administrado a través de Internet, ya que la consola central se ejecuta en un navegador web.

Figura N° 33 Plataforma de DataSynapse



Fuente: www.datasynapse.com

e) Entropía

Entropía es una empresa que comercializa el producto DCGrid, el cual aprovecha los ciclos de procesamiento no utilizados en computadores de la plataforma Windows. Este sistema está integrado por los siguientes componentes: clientes, administrador y calendarizador (ver Figura N° 34). El cliente se instala en las computadoras deseadas como parte del grid.

El mecanismo de ejecución de tareas es no intrusivo, de esta manera una computadora trabaja con una tarea a la vez, sin interrumpir las actividades del usuario. Si por alguna razón la estación de trabajo de usuario está consumiendo muchos recursos -por ejemplo CPU, disco o memoria- la tarea es automáticamente pausada y asignada a un cliente que tenga los recursos disponibles.

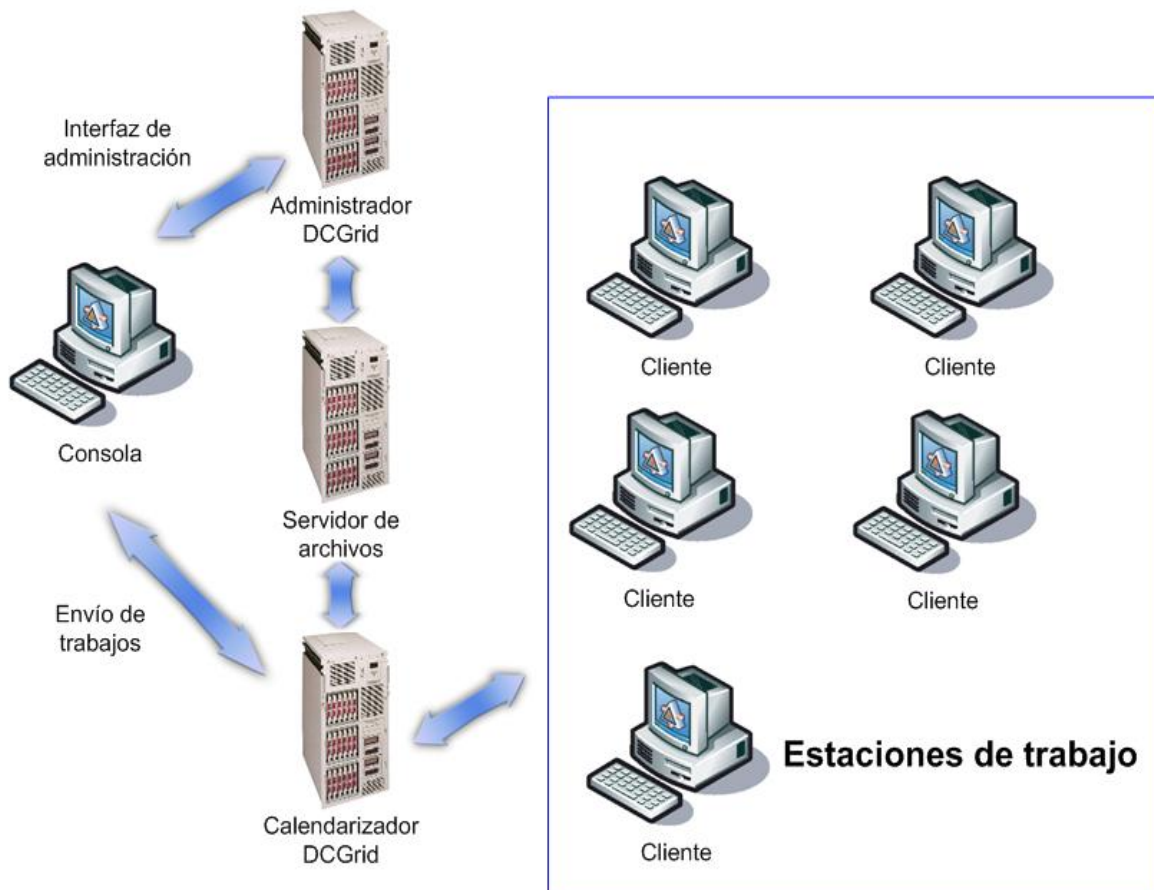
Para aumentar la seguridad de las aplicaciones, cada cliente de Entropía cuenta con un espacio aislado de seguridad o “sandbox”. Este espacio es utilizado para ejecutar las tareas asignadas por el calendarizador, de esta forma las tareas de procesamiento no tienen acceso a los recursos privados de la estación de trabajo.

DCGrid funciona únicamente en plataformas Windows y no requiere que se modifique el código fuente de las aplicaciones por ejecutarse en el grid. El tipo de aplicaciones que explotan mejor las características de DCGrid son aquellas que tienen altos requerimientos de procesamiento y cuya ejecución puede ser en paralelo.

f) Platform Computing

Esta empresa canadiense se dedica a brindar soluciones de informática distribuida. Entre ellas se encuentran LSF, ActiveCluster y MultiCluster (ver Figura N° 35). Además, distribuye una versión de Globus Toolkit y brinda soporte a las empresas para que lo implementen y utilicen adecuadamente.

Figura N° 34 Plataforma basada en Entropía



Fuente: www.entropia.com

Los productos de Platform Computing se pueden ejecutar en los siguientes sistemas operativos: AIX, IRIX, Tru64, HP-UX, Solares, Windows y Linux. Esto le da la ventaja de utilizar recursos que están en plataformas heterogéneas, los cuales se pueden agrupar sin importar si pertenecen a una empresa u otra. De esta manera, las cargas de procesamiento de información pueden ser distribuidas equilibradamente.

La infraestructura de LSF tiene los siguientes componentes:

- Computadora Maestro.
- Computadora Servidor.

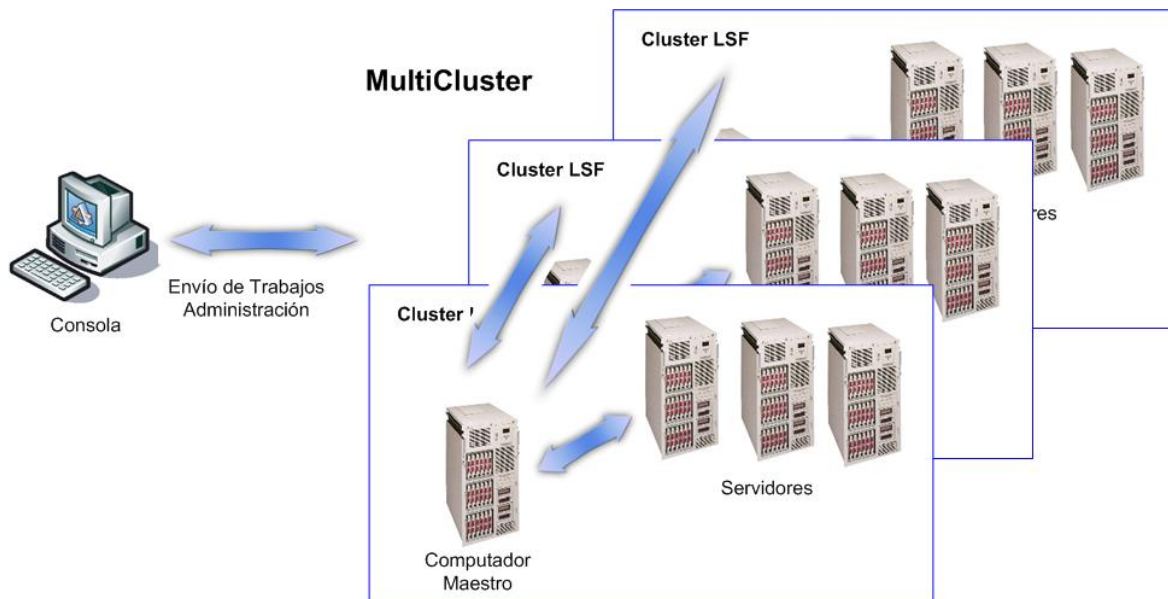
- Computadora Cliente.

El mecanismo de distribución de carga es ejecutado por las computadoras maestro y son responsables de asignar las tareas a las computadoras pertenecientes al grid que controlan. Es posible que exista más de un maestro por grid; por lo tanto, la distribución de carga debe ser coordinada entre todas las computadoras maestro. Las computadoras de tipo servidor se encargan de ejecutar las tareas asignadas por las computadoras maestro; las computadoras cliente se encargan de enviar sus solicitudes de procesamiento de información a las computadoras maestro.

El producto ActiveCluster es una extensión de LSF que trabaja en ambientes Windows. Se encarga de ejecutar las tareas asignadas por los calendarizadores de LSF. El mecanismo de ejecución de ActiveCluster se basa en la ejecución no invasiva de tareas.

MultiCluster se utiliza para enlazar varios clusters LSF, de forma tal que se pueden formar grandes redes de procesamiento de datos. Con el uso de MultiCluster, se hace posible que se unifiquen las iniciativas de distintas partes de la organización. Una posible aplicación de MultiCluster es la siguiente: si un departamento posee un cluster y por determinado momento requiere más poder de procesamiento, por medio de MultiCluster puede conectarse a un cluster de otro departamento y ejecutar las tareas que necesita. De la misma forma, si un departamento posee computadoras con poder de procesamiento que no está utilizando, le puede vender o alquilar este poder a otros departamentos que lo necesiten.

Figura N° 35 Infraestructura basada en Platform Computing



Fuente: www.platform.com

g) United Devices

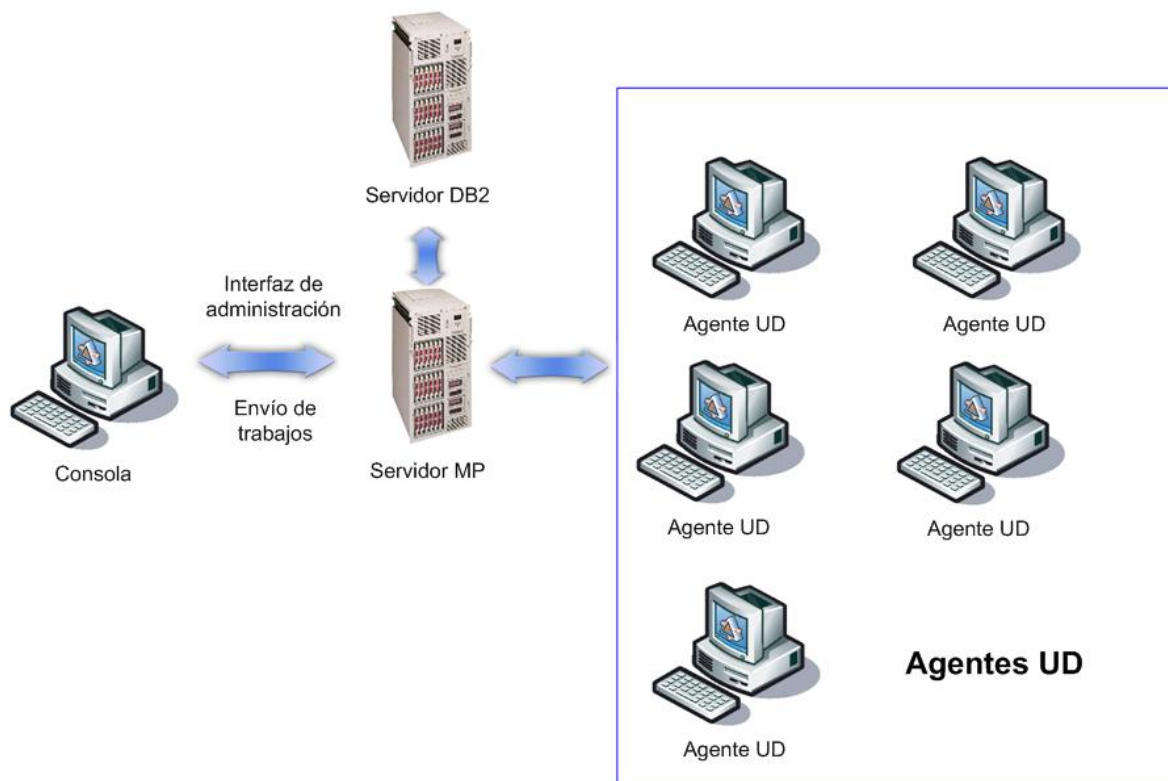
United Devices es una empresa con operaciones centrales en Austin Texas; el producto que ofrecen es MetaProcessor Platform (Grid MP), este se encarga de utilizar los ciclos de CPU no utilizados por las computadoras miembros del grid.

En la Figura N° 36, se observa los componentes de la infraestructura de Grid MP, la cual está compuesta por un servidor (MP Server) y por un cliente (MP Client); toda la plataforma puede ser administrada a través de Internet por medio de una consola central.

El MP Server debe ejecutarse en el sistema operativo Red Hat Linux; sin embargo, la base de datos puede llevarse a cabo en cualquier plataforma donde se pueda ejecutar DB2. El software del cliente puede ejecutarse en Microsoft Windows 98, Millenium, NT, 2000, XP y 2003 y en Linux. La tarea del servidor es calendarizar los trabajos en los clientes, recolectar los resultados y administrar todos los recursos. La base de datos se utiliza para

guardar estadísticas sobre la plataforma, las cuales incluyen tareas ejecutadas, usuarios, agentes instalados en los clientes, entre otros.

Figura N° 36 Infraestructura basada en United Devices



Fuente: www.ud.com

Entre los clientes de United Devices se destacan los siguientes:

- British Telecom.
- Arcelor.
- Schlumberger.
- J&J PRD.
- IBM.
- CInergy/Duke Energy.

- Novartis.
- Top 5 Auto Manufacturers.
- Dassault Aviation.
- CETIM.
- Dr. Reddy's Laboratories.
- Philips.
- University of Zurich.
- CCLRC (Council for the Central Laboratories of the Research Councils).
- U.S. Department of Defense.
- Children's Memorial Research Center.
- American Diabetes Association.
- Oxford University.
- University of Texas.
- National Physics Laboratory (UK).
- Sumisho Electronics.
- Texas A&M University.
- University of Florida.
- Corus Automotive.
- GlaxoSmithKline.
- Gaz de France.
- Sanofi~Aventis.
- EADS (European Automotive Defense and Space Co).
- C-DAC (Center for Advanced Computing).
- University of Georgia.
- Georgia State University.
- Framatome.
- Japan Biological Information Research Center (JBIRC).
- Attorney General's Office (confidential location).

- University of Texas Medical Branch.
- Rapid Prototypes.
- Purdue University.
- Humber College.
- Landmark Graphics.
- Rockefeller University.
- Singapore Computer Systems (SCS).
- University of Basel.
- National Foundation for Cancer Research.
- USAMRIID.
- Galderma.

4. Comentarios sobre los sistemas existentes

Actualmente, existe gran cantidad de proyectos que implementan diferentes enfoques de grid computing. Shapin y Buyya (2000) han establecido esos distintos enfoques, los cuales se pueden clasificar en tres grandes áreas: Grid Computacional, Grid de Calendarización y Grid de Datos. El único proyecto de Grid de datos existente hasta la fecha es el CERN Data Grid.

Los distintos tipos de grid son combinados para crear aplicaciones en grid, por ejemplo, Globus y Legion. Sin embargo, se puede identificar que los sistemas de grid computing existentes tienen un enfoque de administración de recursos centralizado, o sea, la forma en que se asignan recursos depende de las políticas de cada sitio participante del grid. Por lo tanto, se hace necesario un modelo de administración de recursos no centralizado, que soporte la asignación y calendarización de recursos con base en los requerimientos de calidad de servicio del usuario.

5. El impacto económico de la informática distribuida

El reporte "Grid Computing - Projected Impact on North Carolina's Economy and Broadband Use through 2010" publicado por el Dr. Robert B. Cohen, sostiene que la economía de Carolina de Norte obtendrá \$10 billones de dólares estadounidenses adicionales y se dará un crecimiento de 24,000 trabajos: todo esto gracias a la aplicación de la informática distribuida. Este estado posee alrededor de seis millones de habitantes y es comparable a la población de cualquier país de Latinoamérica o de Europa, por lo que los hallazgos del Dr. Cohen pueden ser utilizados como un ejemplo aplicable al resto del mundo y se podría determinar de esta manera el impacto económico que tiene la aplicación de la informática distribuida.

De acuerdo con el estudio del Dr. Cohen, la utilización de los clusters de alto rendimiento y de la informática distribuida se traduce en ganancias de productividad y se proyecta que para el año 2010 se tendrían los siguientes beneficios:

- Entradas adicionales por \$10.1 billones de dólares estadounidenses.
- Aumento de 1.5% de productividad.
- Ingresos personales adicionales de \$7.2 billones de dólares estadounidenses.
- Alrededor de 24,000 puestos de trabajo, que son el resultado de un aumento de 55,700 trabajos creados debido al crecimiento de la industria menos 31,700 trabajos perdidos debido a la adopción de las nuevas tecnologías.

La aplicación de la tecnología de informática distribuida implica que se deben realizar cambios estructurales en la economía de Carolina del Norte:

- Primero. El impacto de la tecnología se verá en múltiples sectores, lo cual contribuye a movimientos en la mezcla de industrias locales. Las inversiones realizadas por las industrias pioneras en grid computing contribuirá en las ganancias

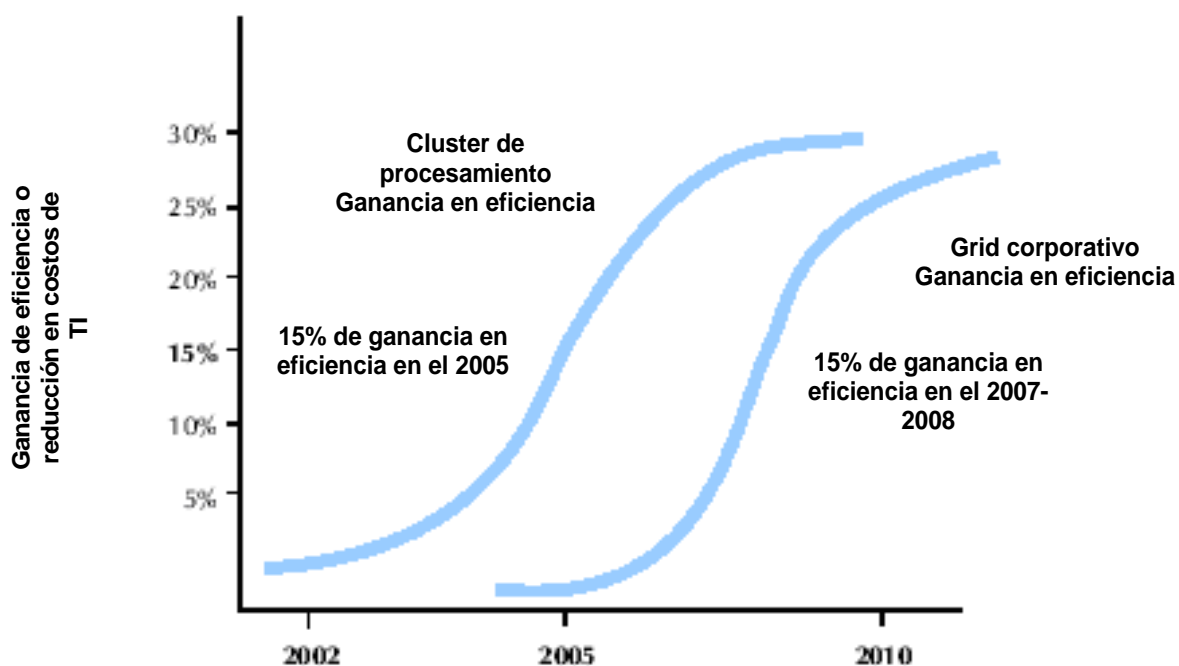
de otros sectores que dan soporte directo a la tecnología computacional. Por ejemplo, se incrementará la inversión en software en US \$1.13 y en computadoras en US \$ 681 millones.

- Segundo. Se incrementarán los ingresos de los proveedores de servicios de telecomunicaciones, puesto que para interconectar recursos distribuidos geográficamente es necesario contar con redes de comunicación de alto desempeño y seguras.
- Tercero. El incremento en productividad de los empleados reducirá los costos de las empresas, con lo cual se podrán realizar mayores inversiones en investigación y desarrollo. Esto, a largo plazo, aumentará la competitividad de Carolina del Norte. En el Gráfico N° 2, se muestra que se puede aumentar hasta un 15% de la eficiencia debido a la adopción de la informática distribuida.
- Cuarto. Los precios de los bienes y servicios incrementarán a un ritmo más lento en aquellas economías en donde las empresas aumentan su productividad, lo cual las hace más competitivas.
- Quinto. Las prácticas del negocio cambiarán drásticamente en ciertas industrias, pues cuando se adopta una nueva tecnología existen grandes ahorros en tiempo y recursos. Sin embargo, a medida que las empresas utilizan recientes tecnologías, empiezan a considerar cómo las nuevas capacidades ofrecidas por las computadoras pueden traer nuevos beneficios: algunas empresas transforman sus procesos para desarrollar productos de forma más eficiente. Por ejemplo, las industrias automotrices y farmacéuticas han comenzado a probar nuevos productos utilizando técnicas de modelaje más sofisticadas y bases de datos altamente detalladas. Las compañías farmacéuticas han empezado a utilizar computadoras más sofisticadas para analizar la forma como ciertas proteínas reaccionan en el cuerpo humano, con lo cual se reduce la cantidad de experimentos.

Según el Dr. Cohen, las industrias que adopten la tecnología de la informática distribuida se beneficiarán con reducciones en costos de mano de obra y de otro tipo (Ver Gráfico N° 3);

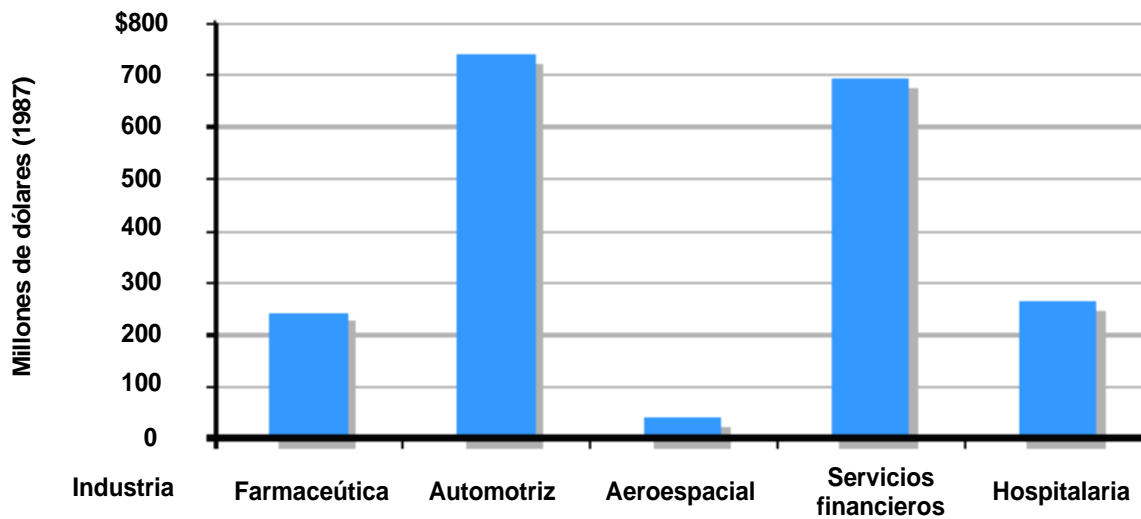
estos ahorros incrementan la productividad de mano de obra. Se proyecta que se obtendrán acrecentamientos en las ganancias de las industrias; por ejemplo: US\$ 242 millones en la industria farmacéutica, US\$ 736 millones en la industria automovilística, US\$ 36 en la industria aeroespacial, \$US 690 en servicios financieros y \$US 262 millones en el sector hospitalario (Ver Gráfico N° 3, Gráfico N° 4, Gráfico N° 5 y Gráfico N° 6).

Gráfico N° 2 Ganancias en eficiencia gracias a la implementación de grid computing



Fuente: Cohen (2003).

Gráfico N° 3 Cambio proyectado en las ganancias netas de las industrias en el año 2010



Fuente: Cohen (2003).

La economía de Carolina del Norte se verá beneficiada debido a la inversión que se realice en clusters y en la informática distribuida. Gracias a las empresas que adopten la tecnología inicialmente, provocará que se den beneficios en otros sectores. Las áreas de impacto se aprecian en el Cuadro N° 1. El detalle de los beneficios de las empresas que adopten la tecnología se muestra en el Cuadro N° 2 y en el Gráfico N° 4.

Cuadro N° 1 Impacto de la informática distribuida en la industria

Year	Areas of Impacts Over Baseline Forecast																			
	Pharmaceuticals				Autos				Aerospace				Financial Services				Health Care Services			
	'02	'05	'08	'10	'02	'05	'08	'10	'02	'05	'08	'10	'02	'05	'08	'10	'02	'05	'08	'10
<i>Change in Output — cumulative percentage increases over expected</i>	0.5	13	28	36	0.5	10	30	40	0	4	15	30	0.5	13	28	36	0.5	8	20	35
<i>Output/Employee — change by this year</i>	0	10	20	25	0	10	22	28	0	3	10	20	0	10	20	25	0	5	14	25
<i>Cost Savings — usually due to outsourcing — cumulative</i>	0	4	6	7	0	4	8	12	0	3	7	15	0	4	6	7	0	4	6	10
<i>Outsourcing increase in percent — cumulative</i>	0	2	5	7	0	4	8	12	0	6	10	15	0	2	5	7	0	2	5	8
<i>Productivity Increases — cumulative</i>	0	3-6	8-12	15-25	0	3-6	9-15	15-25	0	2-4	6-12	10-20	0	4-6	8-12	15-25	0	3-6	6-12	12-20
<i>Broadband Demand Changes (bandwidth terms)</i>	4	24	75	115	2	18	50	90	2	18	50	90	4	24	75	115	4	24	75	115
<i>Broadband Demand (changes in dollar terms)</i>	0	10	30	50	0	8	25	40	0	8	25	40	0	10	30	50	0	10	30	50

These estimates combine the previous analysis of leading firms in specific sectors with the figures showing the penetration of Grids and Web services in each industry.

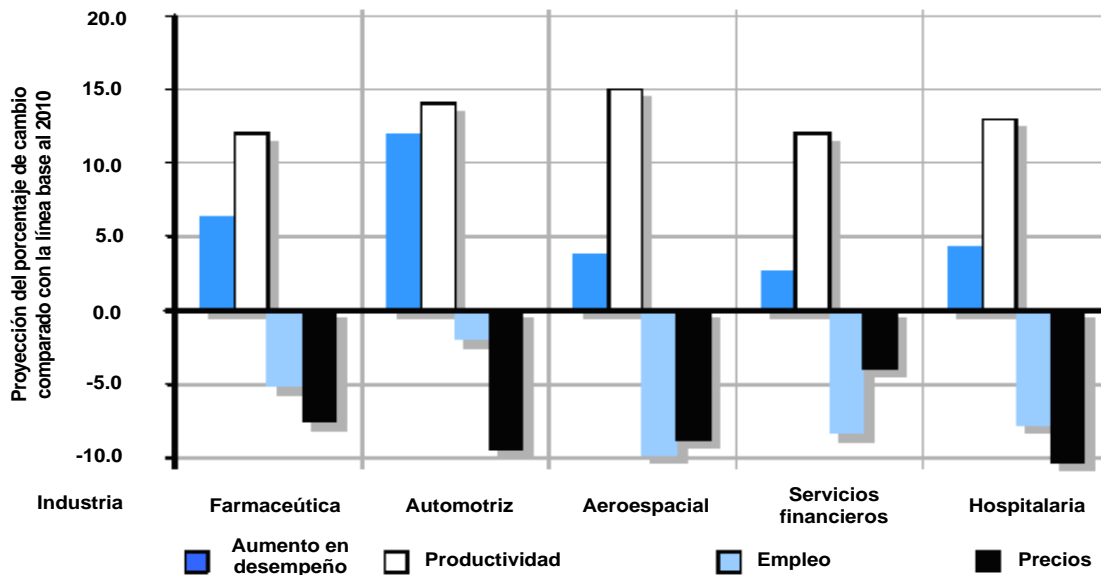
Fuente: Grid Computing - Projected Impact on North Carolina's Economy and Broadband Use through 2010.

Cuadro N° 2 Cambios en los ingresos según industria que adopte inicialmente la tecnología

Industria	Cambios en Ingresos (millones de dólares)	Cambios en ingresos	Cambios en productividad	Cambios en precios	Estimación de empleos 2002/2010	Cambio en empleo
Industria farmacéutica	\$242	6,30%	12%	-7,60%	15.3/16.2	-5,10%
Autos	\$736	11,90%	14%	-9,50%	11.3/12.9	-2,00%
Industria aeroespacial	\$36	3,70%	15%	-8,80%	4.0/3.7	-9,80%
Sector financiero	\$690	2,60%	12%	-4,00%	159/175	-8,40%
Hospitales	\$262	4,30%	13%	-10,30%	104/122	-7,70%
Total	\$1,966				294/329	-25.516

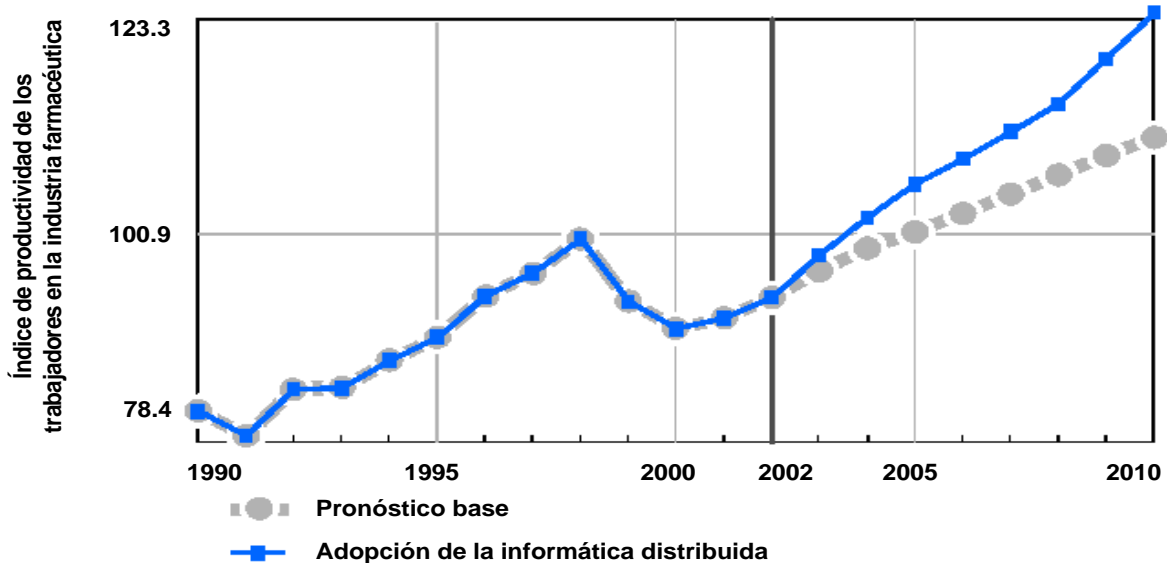
Fuente: Grid Computing - Projected Impact on North Carolina's Economy and Broadband Use through 2010.

Gráfico N° 4 Cambios en los ingresos, productividad, empleo y precios gracias a la adopción de la informática distribuida, según industria



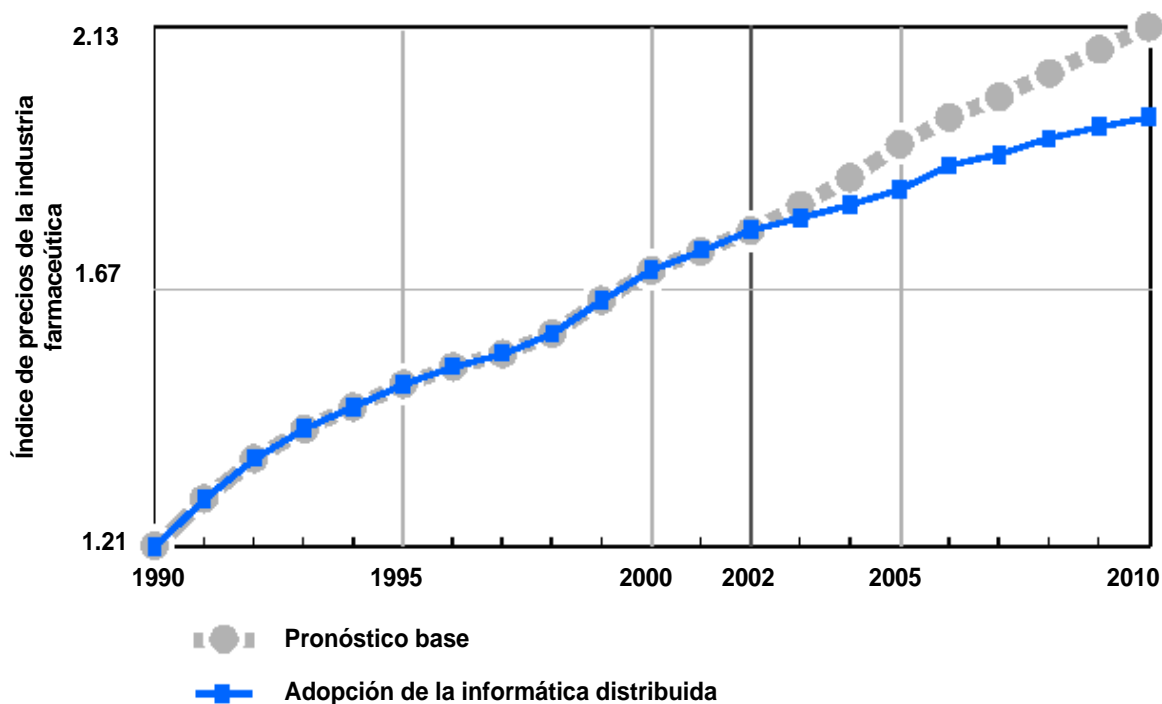
Fuente: Cohen (2003).

Gráfico N° 5 Cambio de productividad laboral en la industria farmacéutica



Fuente: Cohen (2003).

Gráfico N° 6 Cambio en precios en la industria farmacéutica



Fuente: Cohen (2003).

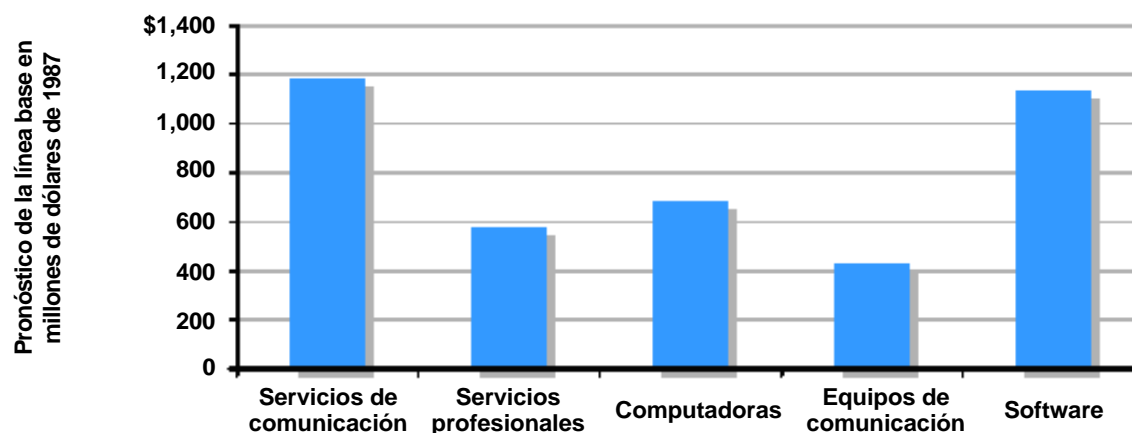
El sector beneficiado directamente es la industria de implementación de tecnologías de información; específicamente, el reporte proyecta que las inversiones de software serán de \$1.13 US billones, las de computadoras rondarán los \$681 US millones; los servicios profesionales, \$575 US millones y las inversiones en equipos de comunicación serán \$432 US millones. Este gasto contribuirá en el desarrollo de la economía de otras industrias; por ejemplo, en cuanto al mobiliario de oficina se espera su aumento en un 19%. El detalle del aporte a otras industrias se observa en el Cuadro N° 3. El estudio estima que el impacto total en la economía será de ganancias de US \$10.9 millones menos US \$800 en pérdidas en otros sectores, tal y como se observa en el Cuadro N° 4.

Cuadro N° 3 Cambios en los ingresos según industria

	Ranking	Cambio en los ingresos	Cambios en Ingresos (millones de dólares)
Equipos de oficina	1	19,00%	36,8
Vehículos automotores	2	11,90%	736,4
Computadoras	3	11,30%	681,4
Equipo de comunicaciones	4	8,40%	431,6
Servicios de comunicación	5	7,90%	1.182,50
Computación y procesamiento de datos	6	7,30%	1.131,30
Maquinaria metalúrgica	7	6,90%	87
Medicamentos	8	6,30%	242
Muebles	9	6,20%	630,1
Partes de motor	10	5,10%	319,3
Servicios profesionales	11	4,70%	575,2
Hospitales privados	12	4,30%	262,3
Industria aeroespacial	13	3,70%	38
Instrumental médico	14	3,70%	45
Películas	15	3,10%	143,5
Componentes electrónicos	16	2,90%	364,6
Otros instrumentos	17	2,80%	34,1
Finanzas y seguros	18	2,60%	689,9
Casas de cuidado	19	2,40%	39,8
Electrónica industrial	20	2,40%	33,7
Cambios totales			7704,5

Fuente: Grid Computing - Projected Impact on North Carolina's Economy and Broadband Use through 2010.

Gráfico N° 7 Cambios en las ganancias netas de las industrias de Carolina de Norte gracias a las inversiones en informática distribuida



Fuente: Cohen (2003).

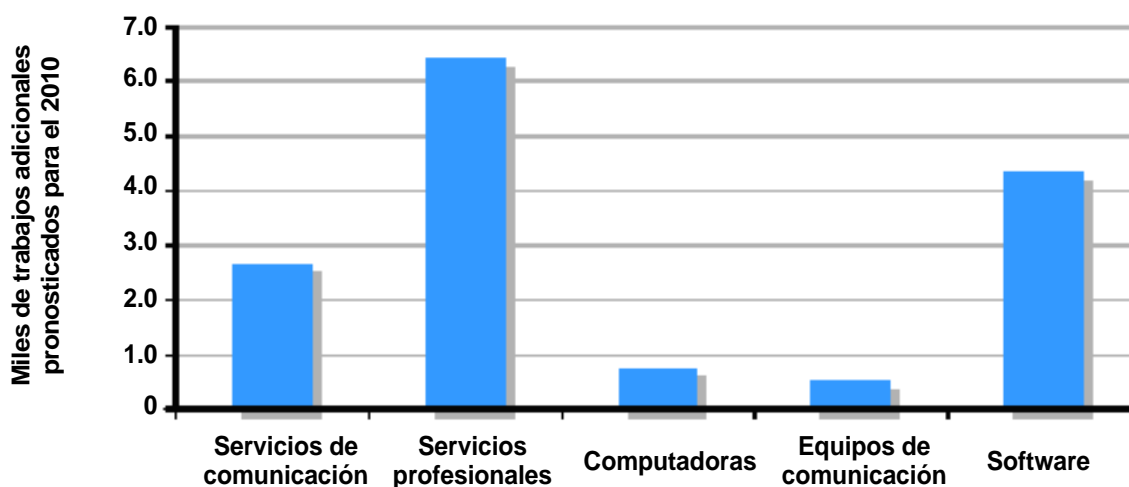
Cuadro N° 4 Cambios en los ingresos según industria

	Ranking	Cambio en los ingresos	Cambios en Ingresos (millones de dólares)
Industrias de gobierno	1	-2,90%	-128,6
Motores y turbinas	2	-2,00%	-17,6
Otros servicios de negocios	3	-1,30%	-165,2
Servicios automovilísticos	4	-1,20%	-89,3
Publicidad	5	-1,10%	-70,7
Equipo de búsqueda y navegación	6	-1,00%	-5,1
Transporte aéreo	7	-0,90%	-48,1
Impresión y publicación	8	-0,90%	-39,3
Comidas enlatadas y congeladas	9	-0,80%	-13,7
Productos plásticos	10	-0,80%	-46,6
Productos de hule	11	-0,70%	-15,7
Servicios de transporte	12	-0,60%	-5,7
Tránsito de camiones y pasajeros	13	-0,60%	-55,7
Bebidas alcohólicas	14	-0,60%	-5,9
Servicios sanitarios	15	-0,60%	-9
Cambios totales			-716,2

Fuente: Grid Computing - Projected Impact on North Carolina's Economy and Broadband Use through 2010.

El incremento de productividad en la mano de obra debido a la aplicación de la informática distribuida, según el Dr. Cohen, reduce los costos y aumenta la eficiencia. De esta forma, se liberan recursos de capital para invertir en la construcción de nuevas plantas, en la compra de equipo y la capacitación y entrenamiento de los empleados; además, se pueden inyectar recursos a labores de investigación y desarrollo.

Gráfico N° 8 Ganancias en empleo gracias a las inversiones en informática distribuida



Fuente: Cohen (2003).

Se espera que las políticas de las empresas cambien debido a la adopción de la informática distribuida y cuando se haga se verán beneficiadas por los grandes ahorros en tiempo y recursos. Dichos ahorros dependen del modelo de negocio que adopten las empresas. (Ejemplos del impacto de la inversión en esta tecnología se observan en la Tabla N° 18.) Para que este ahorro se dé, será necesario que las cadenas de proveedores estén alineados tecnológicamente, pues a medida que las grandes empresas actualizan su tecnología, exigirán que sus proveedores sean capaces de interactuar con ellos y que se puedan aprovechar los recursos distribuidos de manera geográfica. Este será un reto principalmente para aquellas empresas que están ubicadas en lugares con difícil acceso a conexiones de Internet o de banda ancha.

6. Impacto en las empresas

El reporte concluye que para Carolina del Norte la tecnología de informática distribuida puede lograr un cambio cualitativo y cuantitativo en la forma como las empresas realizan sus negocios. Esto se puede apreciar en la Tabla N° 18, en la cual se ilustran ejemplos de cómo ha cambiado el modelo de operación de las empresas debido a la adopción de la informática distribuida.

Sin embargo, deja claro que es necesario que se desarrolle la infraestructura necesaria para dar soporte a la tecnología; así como se deben de desarrollar las destrezas adecuadas en el personal que diseñará y atenderá los modelos de operación basados en la informática distribuida.

Tabla N° 18 Propósito e impacto en el negocio de la informática distribuida

Tipo de grid	Propósito e impacto en el negocio
Grid de datos	<p>Propósito</p> <p>Conectar bases de datos ubicadas en distintas ubicaciones geográficas de la empresa.</p> <p>Impacto en el negocio</p> <p>Ahorros significativos en la búsqueda de información</p> <p>Inversiones realizadas</p> <p>Grandes inversiones realizadas en enlaces de banda ancha que conecten a todos los centros de datos de las distintas localidades geográficas.</p> <p>Ejemplos</p> <p>Eli Lilly. Grid de datos entre Sphinx Labs en Carolina del norte y otros laboratorios de Investigación y desarrollo.</p>

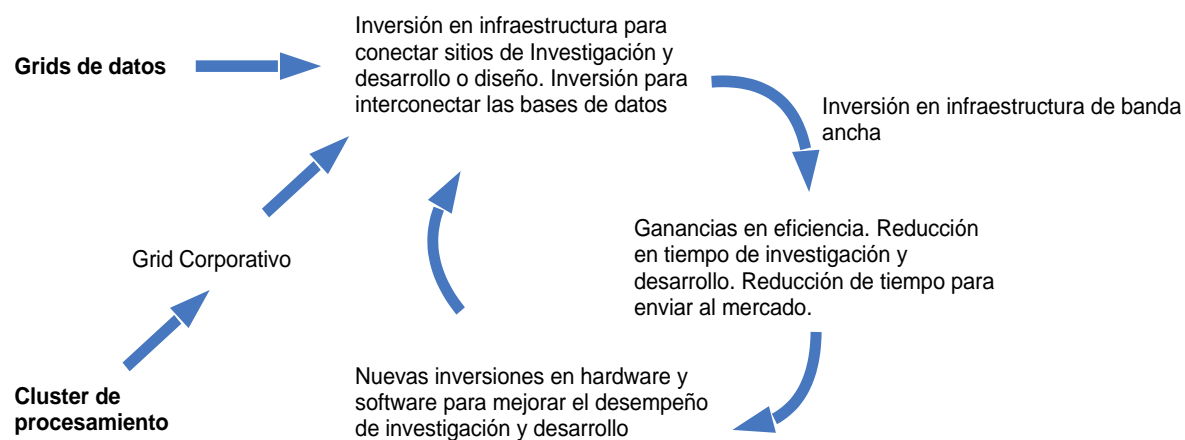
Tipo de grid	Propósito e impacto en el negocio
	<p>Banco de América. Enlaces entre los centros de datos para que soporten su sistema de cheques accesible vía Internet.</p>
<p>Cluster de procesamiento</p>	<p>Propósito Poder de procesamiento. Unir computadoras para lograr alta velocidad de procesamiento.</p> <p>Impacto en el negocio Ahorros en tiempo de procesamiento. Agrega eficiencia ya que se logra mayores logros. Ahorros en costos de investigación y en diseño.</p> <p>Inversiones realizadas No existe impacto en inversiones de banda ancha hasta que se evoluciona hacia en Grid Corporativo.</p> <p>Ejemplos Pratt & Whitney ahorraron 50 por ciento en tiempo de ingeniería utilizando Platform Computing. Novartis aceleró la identificación de medicamentos en un factor de diez veces más. Universidad de Oxford. Se hizo una revisión de billones de posibles compuestos de medicamentos en 24 días.</p>
<p>Grid Corporativo</p>	<p>Propósito Poder de procesamiento más transporte de cargas de información. Enlaza centros de investigación y desarrollo en distintas zonas geográficas.</p> <p>Impacto en el negocio Eficiencia debida a poder de procesamiento más el acceso a datos. Se presentan ahorros en investigación y desarrollo y se reduce el tiempo de</p>

Tipo de grid	Propósito e impacto en el negocio
	<p>puesta en el mercado.</p> <p>Inversiones realizadas</p> <p>Inversiones en enlaces de banda ancha, pueden requerirse altas velocidades debido a la cantidad de información que debe ser transmitida.</p> <p>Ejemplos</p> <p>AstraZeneca. Suiza, Estados Unidos y el Reino Unido</p> <p>General Motors, Daimler Chrysler, Ford. Enlaces entre los grupos de ingeniería y de diseño, la mayoría en Europa.</p>
<p>Grid con socios</p>	<p>Propósito</p> <p>Poder de procesamiento más transporte, para una o más empresas.</p> <p>Impacto en el negocio</p> <p>Ahorros en tiempo de diseño y de investigación y desarrollo, reducción en el tiempo de puesta en el mercado.</p> <p>Permite la colaboración entre socios de negocios, especialmente aquellos involucrados en la cadena de abastecimiento.</p> <p>Inversiones realizadas</p> <p>Inversión en enlaces de banda ancha seguros y de alto desempeño entre dos o más empresas.</p> <p>Ejemplos</p> <p>General Motors, Daimler Chrysler y Ford Motor Company</p>

Fuente: Grid Computing - Projected Impact on North Carolina's Economy and Broadband Use through 2010.

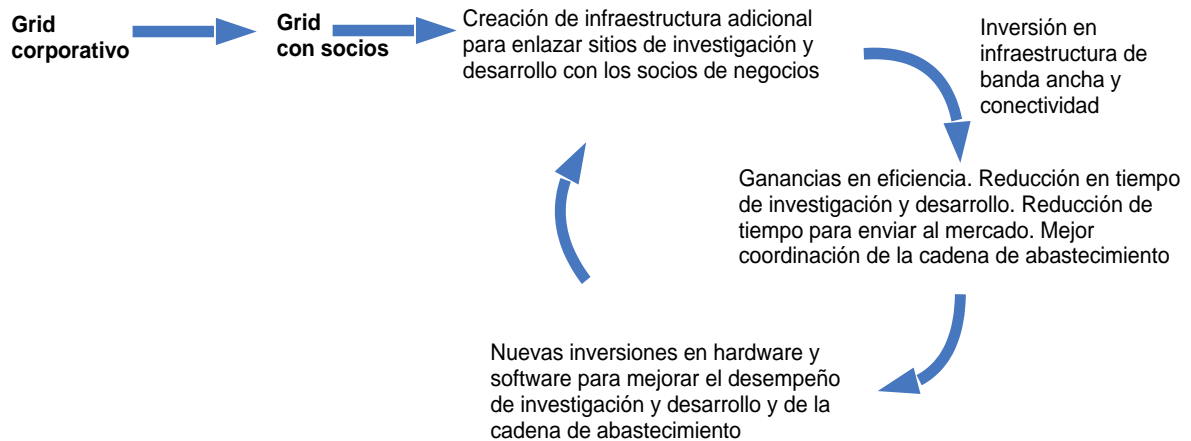
Los beneficios económicos de la aplicación de la tecnología son el resultado de su adopción, la cual se da en dos fases como se muestra en la Figura N° 37. La primera consiste en interconectar los recursos internos de la empresa, sean estos locales o remotos. Una vez que esta primera fase está consolidada, el paso siguiente es la integración con los socios de negocios -principalmente aquellos que tienen un papel importante en la cadena de abastecimiento. La Figura N° 38 muestra la implementación de la segunda fase de la tecnología, en ella se incluyen a los asociados de negocios estratégicos con el fin de acelerar los procesos de investigación y desarrollo y, sobre todo, agilizar la cadena de abastecimiento.

Figura N° 37 Primera fase de adopción de la tecnología



Fuente: Grid Computing - Projected Impact on North Carolina's Economy and Broadband Use through 2010.

Figura N° 38 Segunda fase de adopción de la tecnología



Fuente: Grid Computing - Projected Impact on North Carolina's Economy and Broadband Use through 2010.

Entre los cambios en los procesos de las empresas se encuentran los siguientes:

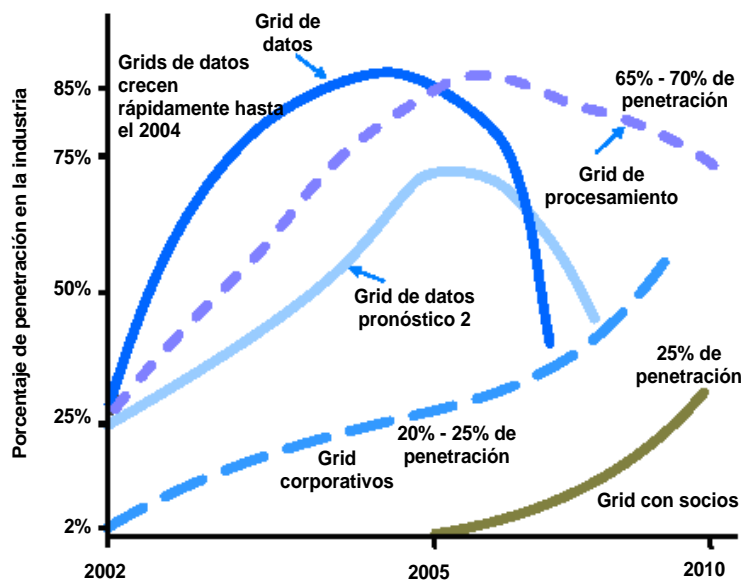
- En la cadena de abastecimiento: La adopción de los grids por parte de los distribuidores y fabricantes tendrá diferente impacto en las empresas proveedoras y redes de distribución. Conforme se aumenta la tecnología en las empresas manufactureras, éstas provocan que las empresas interactuantes con ellas se modernicen para que el proceso total de producción sea más eficiente y rápido.
- En las industrias de software y hardware: Las empresas de software, almacenamiento de información y procesamiento de datos que dan soporte a las empresas que utilizan grids, así como los proveedores de servicios de telecomunicación se verán afectados, pues tendrán que conocer la forma cómo los negocios utilizan la nueva tecnología y –principalmente- deberán dominar las nuevas tecnologías necesarias para soportar el nuevo modelo de negocios.
- Olas de adopción: Algunas empresas retrasarán la adopción de la tecnología de informática distribuida hasta enterarse de que sus competidores los aventajan. Por lo tanto, la demanda de hardware y software que soporten la nueva tecnología vendrá

en olas. Por ejemplo, un sector en el cual puede darse este caso es en los bancos de segundo nivel. Se darán cuenta de que la pérdida de sus clientes se debe al ofrecimiento de mejores servicios al usuario por parte de los bancos rivales, como servicios financieros de bajo costo a través de Internet. Otro sector que podría ser involucrado es el farmacéutico; sería el caso de empresas grandes, las cuales de forma tardía se den cuenta que si no adoptan rápidamente la nueva tecnología, estarán en desventaja en la carrera por encontrar nuevos medicamentos para comercializar.

Estas olas de adopción se harán en fases. En el Gráfico N° 9, se muestra la adopción de la tecnología en la industria farmacéutica. Al inicio, se tiene entre un 20% y 25% de penetración hasta llegar a un 65% a 75% una vez que la implementación de la informática distribuida sea consolidada. El Gráfico N° 10 muestra que para el año 2010 se espera que la penetración en la industria automovilística sea de aproximadamente un 85%.

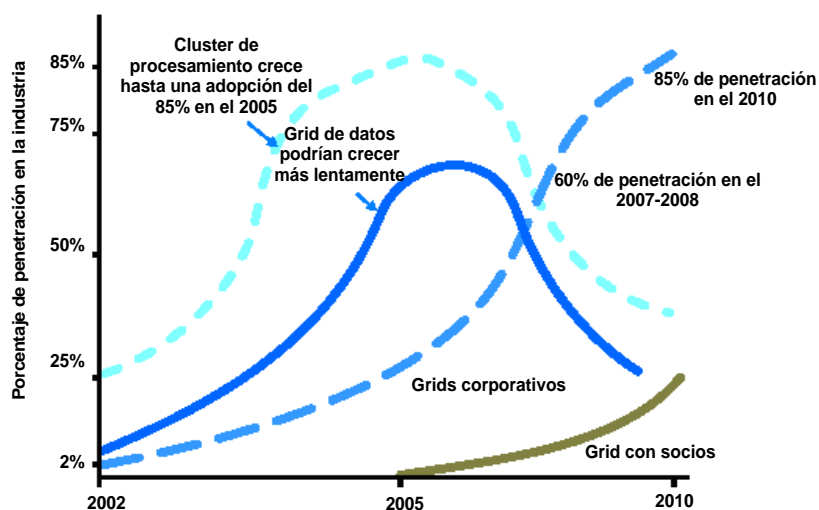
- Nuevos enlaces comprador-proveedor y necesidades de habilidades: Cuando las empresas adoptan grid computing y forman grids con asociados de negocio se inicia una relación comprador-proveedor. Por ejemplo, las empresas de investigación trabajarán estrechamente con las empresas farmacéuticas. El surgimiento de estas redes de colaboración hará necesario que las empresas cuenten con las habilidades de personal precisas para hacer frente a todas las necesidades que surgen durante la administración de redes complejas de colaboración con base en la informática distribuida.

Gráfico N° 9 Grids y servicios distribuidos en la industria farmacéutica



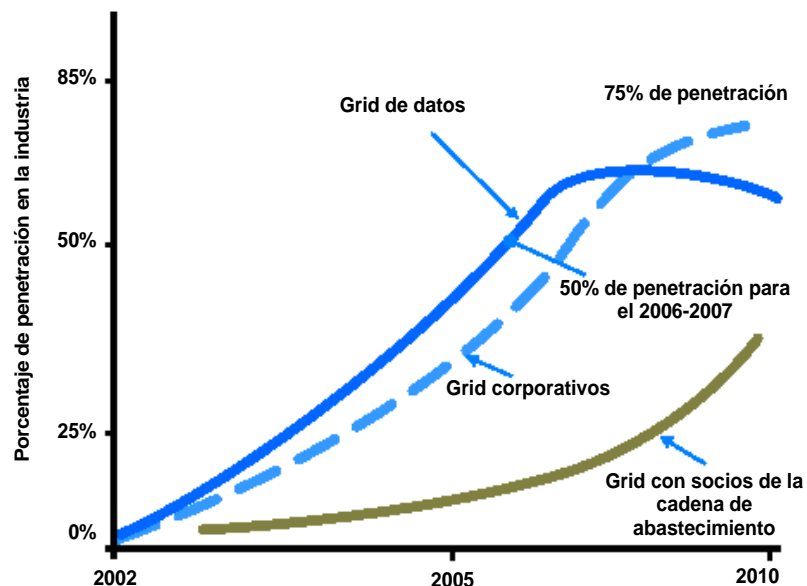
Fuente: Grid Computing - Projected Impact on North Carolina's Economy and Broadband Use through 2010.

Gráfico N° 10 Grids y servicios distribuidos en la industria automovilística



Fuente: Grid Computing - Projected Impact on North Carolina's Economy and Broadband Use through 2010.

Gráfico N° 11 Grids y servicios distribuidos en la industria de servicios financieros



Fuente: Grid Computing - Projected Impact on North Carolina's Economy and Broadband Use through 2010.

El estudio del Dr. Cohen indica que la dinámica de crecimiento de Carolina del Norte podría cambiar de una basada en tecnología de la revolución informática tradicional (computadoras y software) hacia una revolución de conectividad (redes, acceso rápido a bases de datos y conexiones rápidas entre las empresas). Este estudio concluye:

- Las empresas que adopten la informática distribuida obtendrán grandes beneficios en cuanto a ganancias de eficiencia en los próximos cinco u ocho años, con lo cual reducirán su estructura de costos, mejorarán su productividad, reducirán los precios de sus productos y serán más competitivas. Las industrias en las cuales estas ganancias podrían ser enfocadas incluyen la automotriz y aeroespacial, servicios financieros, sector farmacéutico y hospitalario. La inversión realizada por estos sectores provocará crecimiento en los sectores de computación, software, equipo de comunicación y servicios profesionales.

- El análisis indica que el estado de Carolina del Norte en el año podría crecer hasta US \$10 billones en el sector industrial, US \$7 billones en los ingresos personales como resultado de las inversiones en informática distribuida. El detalle de estas inversiones se muestra en el Cuadro N° 5. Estas inversiones también estimularán el crecimiento de otras industrias relacionadas, tales como equipo de oficina, equipo hospitalario e industrias de proveedores.
- La implementación de la informática distribuida hará surgir la necesidad de personal altamente calificado que pueda hacer frente a la implementación de redes complejas de colaboración distribuidas geográficamente.

En resumen, la implementación y uso de la informática distribuida involucra cambios cualitativos en la forma en que las empresas llevan a cabo sus operaciones y beneficiará a las empresas debido a que estarán a la vanguardia de la tecnología y las hará más competitivas.

Cuadro N° 5 Inversiones en informática distribuida en millones de US\$

	Año (millones de US\$)		
	2003-05	2006-07	2008-10
Industria farmacéutica			
Computadoras	\$200	\$463	\$763
Equipo de comunicaciones	\$100	\$250	\$450
Software	\$200	\$650	\$1,350
Servicios de comunicación	\$300	\$675	\$1,175
Servicios profesionales	\$200	\$463	\$763
Industria aeroespacial			
Computadoras	\$40	\$128	\$353
Equipo de comunicaciones	\$20	\$70	\$220
Software	\$40	\$190	\$715
Servicios de comunicación	\$60	\$185	\$560
Servicios profesionales	\$40	\$128	\$353
Industria automotriz			
Computadoras	\$150	\$400	\$925
Equipo de comunicaciones	\$75	\$375	\$775
Software	\$150	\$750	\$2,150
Servicios de comunicación	\$225	\$475	\$1,025
Servicios profesionales	\$150	\$325	\$1,025
Servicios financieros			
Computadoras	\$3,200	\$16,000	\$31,000
Equipo de comunicaciones	\$3,200	\$7,000	\$14,000
Software	\$1,600	\$11,000	\$23,000
Servicios de comunicación	\$4,800	\$24,000	\$44,000
Servicios profesionales	\$3,200	\$7,000	\$14,000
Servicios de salud			
Computadoras	\$400	\$1,250	\$2,289
Equipo de comunicaciones	\$200	\$675	\$1,350
Software	\$400	\$1,755	\$4,050
Servicios de comunicación	\$600	\$1,823	\$3,525
Servicios profesionales	\$400	\$1,250	\$2,289

Fuente: Grid Computing - Projected Impact on North Carolina's Economy and Broadband Use through 2010.

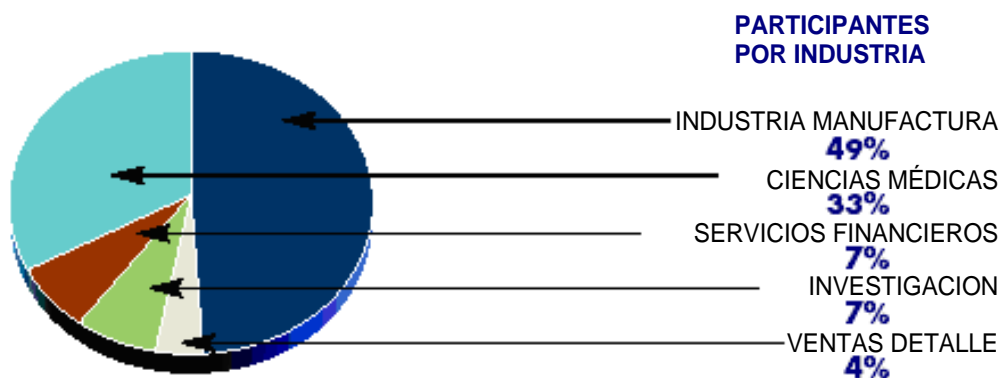
J. Negocios de la Grid Computing

En el año 2004, Platform Computing realizó un estudio que incluyó a 100 ejecutivos de tecnologías de información y fue realizado con el fin de determinar las prioridades con respecto a la inversión en la tecnología de informática distribuida.

En el estudio se incluyeron a 50 compañías, las cuales fueron seleccionadas debido a su familiaridad con la tecnología de informática distribuida y su experiencia en cuanto a los problemas que se pueden presentar durante la implementación y puesta en ejecución de proyectos de este tipo.

Tal y como se muestra en el Gráfico N° 12, el 49% de los participantes son compañías de las industria manufacturera, en las cuales se utiliza la informática distribuida en tareas tales como modelos de computación compleja y simulaciones (mecánica de fluidos y pruebas de coches de automóviles). Un tercio de las empresas son de la industria de ciencias de la vida que utilizan la tecnología para acelerar la búsqueda de nuevos medicamentos y tratamientos. La industria financiera está comenzando a incluir la informática distribuida en de sus operaciones de negocio.

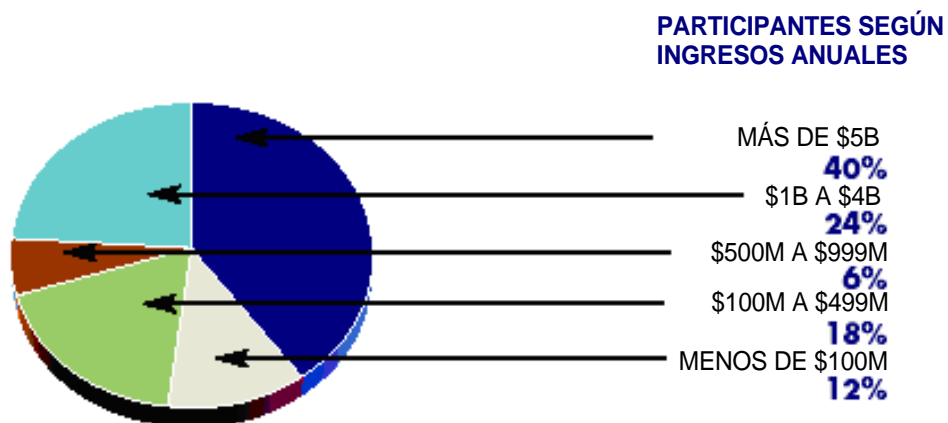
Gráfico N° 12 Participantes según industria



Fuente: Platform Computing (2004).

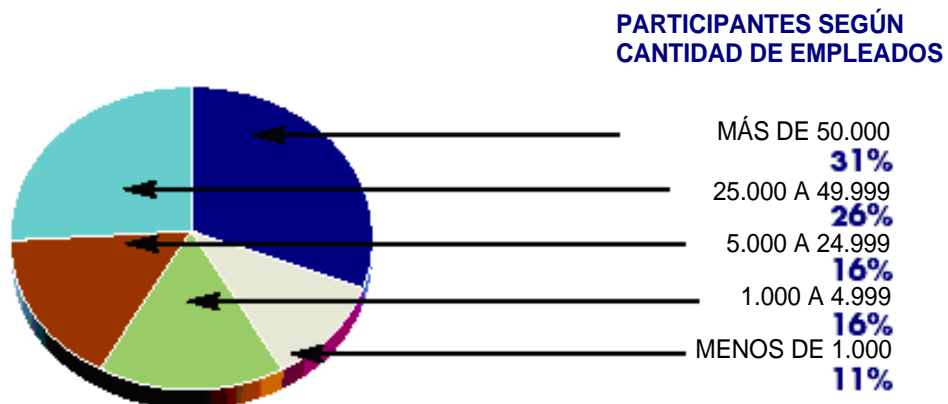
En el Gráfico N° 13 y Gráfico N° 14, se puede apreciar el tamaño de las empresas que participaron del estudio, casi dos terceras partes de estas empresas tienen ingresos por más de un billón de dólares estadounidenses al año y poseen más de 25,000 empleados distribuidos en todo el mundo.

Gráfico N° 13 Ingresos anuales de las empresas participantes



Fuente: Platform Computing (2004).

Gráfico N° 14 Cantidad de empleados de las empresas participantes



Fuente: Platform Computing (2004).

Parte de las empresas participantes en el estudio se listan a continuación, las cuales representan varias de las empresas más competitivas a nivel mundial:

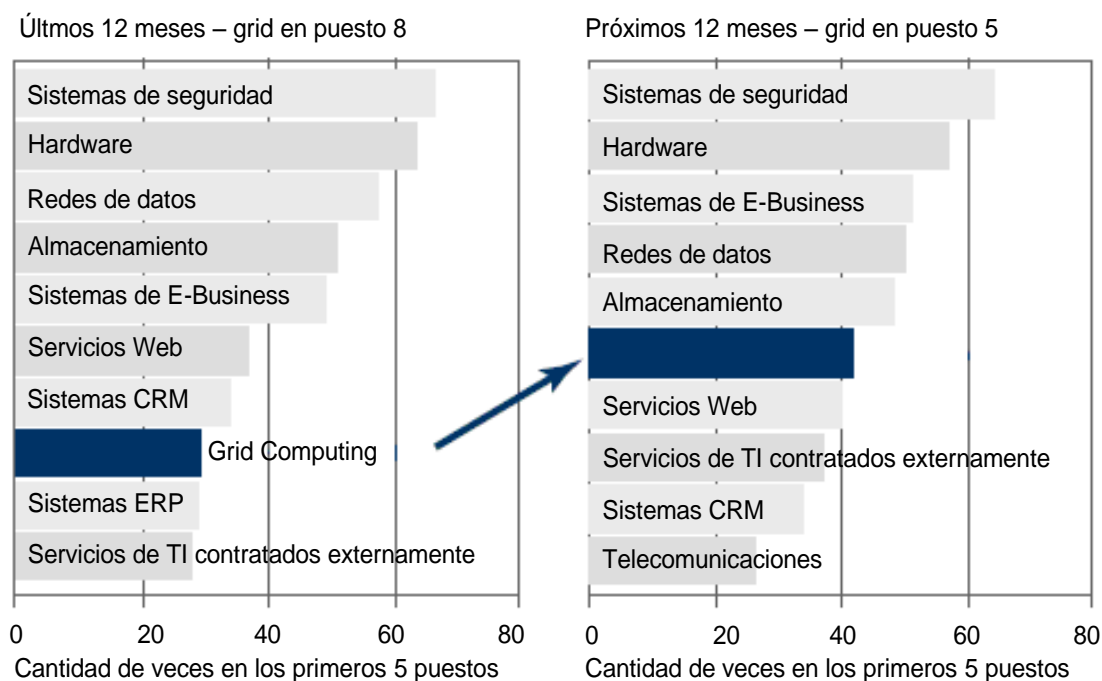
- Adaptec Inc.
- AMCC (Applied Micro Circuits Corporation).
- AMD (Advanced Micro Devices).
- ARM Ltd.
- ATI Technologies Inc.
- Aventis.
- Conexant Systems, Inc.
- Entelos, Inc.
- ETH (Swiss Federal Institute of Technology
• Zurich)
- GlobeXplorer, LLC.
- Incyte Genomics, Inc.
- J.P. Morgan Chase & Co.
- Montage.DMC eBusiness Services.
- Myriad Proteomics, Inc.
- Novartis AG.
- Partners HealthCare System, Inc.
- United Technologies Corp. – Pratt & Whitney
- Qualcomm Inc.
- science+computing AG
- SHARCNET.

Entre las conclusiones de este estudio se encuentran las siguientes:

- Los ejecutivos encuestados identificaron sus cinco prioridades de inversión en tecnologías de la información durante un periodo que incluía el año anterior y el próximo. La tecnología de informática distribuida fue calificada dentro de estas prioridades (ver Gráfico N° 15) debido a las expectativas de retorno de la inversión, el cual se espera, sea mayor que en las inversiones realizadas en outsourcing, CRM (Customer Relationship Management), ERP (Enterprise Resource Planning), almacenamiento de datos, redes y otros.

Gráfico N° 15 Prioridades de inversión de TI

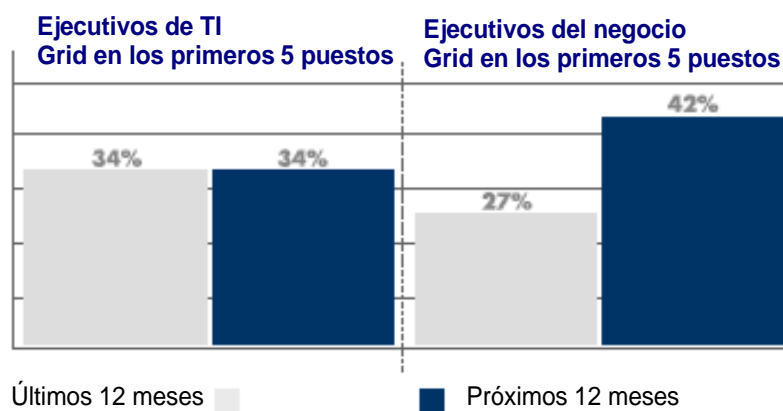
**PRIORIDADES DE INVERSIÓN DE LOS ÚLTIMOS 12 MESES Y
LOS PARA LOS SIGUIENTES 12 MESES**



Fuente: Platform Computing (2004).

- La informática distribuida tiene un impacto directo en las aplicaciones clave de la empresa y, por lo tanto, en la productividad y efectividad de los usuarios de esas aplicaciones, lo cual lleva a mejorar el retorno de la inversión (ver Gráfico N° 17).

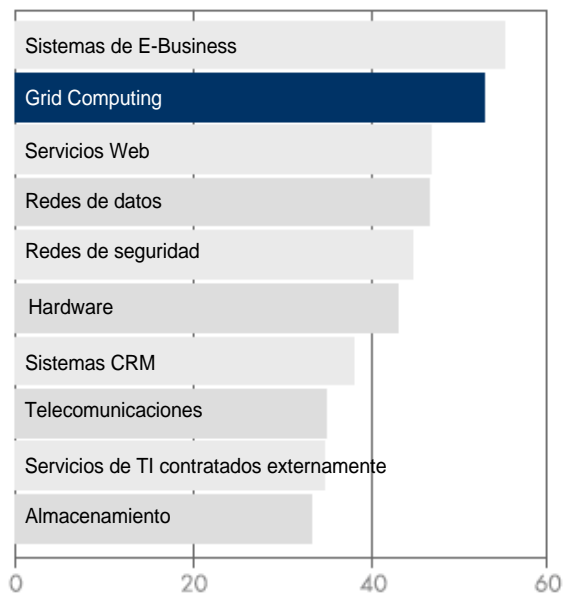
Gráfico N° 16 Proridades los ejecutivos con respecto a la inversión en informática distribuida



Fuente: Platform Computing (2004).

- Dentro de las barreras para la implementación de la informática distribuida, las más importantes son las no técnicas que incluyen las políticas organizacionales (Ver Gráfico N° 18).

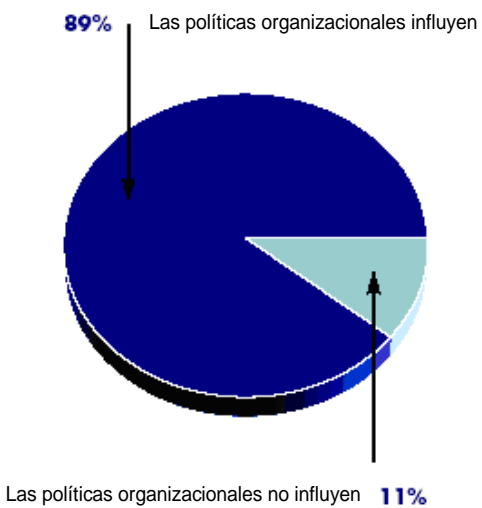
Gráfico N° 17 Inversiones que van a tener un retorno sobre la inversión más alto



Fuente: Platform Computing (2004).

Gráfico N° 18 Principales barreras en la implementación de informática distribuida

Son las políticas organizacionales una barrera para implementar grid computing?



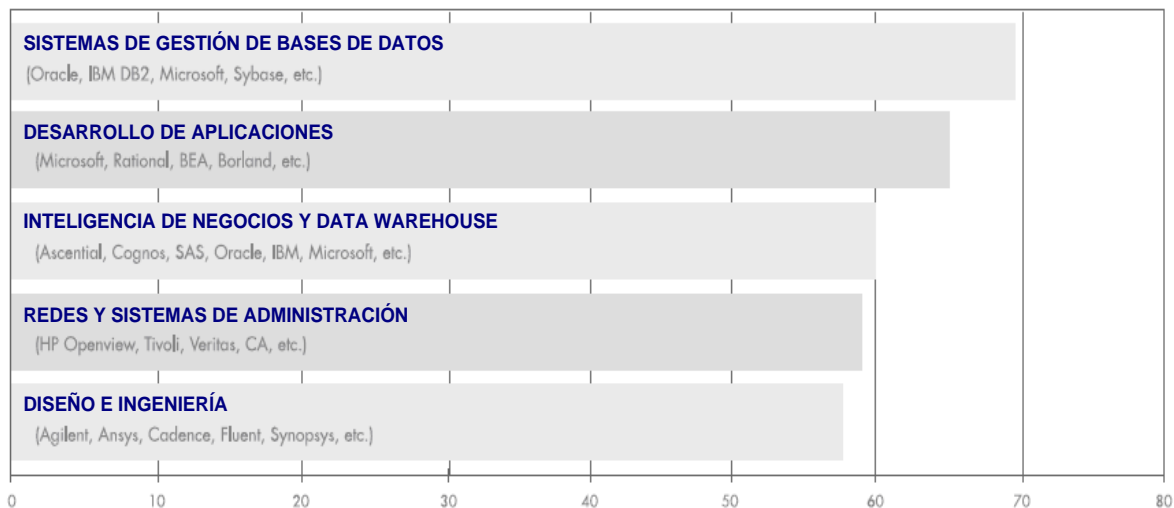
Fuente: Platform Computing (2004).

Como conclusión, los ejecutivos consideran que las aplicaciones más apropiadas para ejecutar dentro de un ambiente de informática distribuida incluyen los sistemas de administración de bases de datos, el desarrollo de aplicaciones, las aplicaciones de inteligencia de negocios y de almacenes de datos, los sistemas de administración de redes y finalmente las tareas de diseño. La distribución de estas preferencias se puede apreciar en el Gráfico N° 19. Muestra como los ejecutivos están empezando a aplicar el valor esperado de grid computing en la operación de sus sistemas críticos y procesos, de estos datos se pueden identificar dos categorías de beneficios:

- Más rápido, mejor y más barato. Los ejecutivos manifiestan que la ejecución de análisis complejo de datos y software de diseño e ingeniería en un ambiente de informática distribuida, incrementa la productividad de los usuarios y su efectividad para acelerar el desempeño de las aplicaciones y se maximiza la utilización de los recursos computacionales.
- Incremento en la confianza operacional. Los resultados del estudio resalta que los ejecutivos encuestados aprecian beneficios al ejecutar las aplicaciones empresariales en una plataforma de informática distribuida. Identifican mejoras en predictibilidad, estabilidad, escalabilidad cuando se administran e instalan sistemas de bases de datos y el desarrollo e integración de aplicaciones muestran mejoras con respecto al monitoreo y administración de toda la infraestructura.

Se puede concluir que los resultados del estudio son una alerta para los productores de software, pues los ejecutivos de las empresas están dispuestos a ejecutar la mayoría de las aplicaciones críticas en la plataforma de informática distribuida. Por lo tanto, los creadores de software deberán adaptar sus soluciones para que sean compatibles con esta nueva plataforma de ejecución.

Gráfico N° 19 Aplicaciones más importantes que se pueden ejecutar en una plataforma de informática distribuida



Fuente: Platform Computing (2004).

De este estudio, se puede concluir que la industria está interesada en adoptar la tecnología de informática distribuida. Se debe, principalmente, a que se podría incrementar la productividad de las empresas con tasas de retorno de inversión favorables desde el punto de vista económico.

IV. Metodología

“El propósito de su tesis es documentar claramente una contribución original al conocimiento. Usted puede desarrollar programas de computadora, prototipos, u otras herramientas como medios para demostrar sus puntos, pero recuerde, la tesis no es sobre la herramienta, es sobre la contribución al conocimiento. Herramientas tales como los programas de computadora son productos buenos, pero no puede obtenerse un título de postgrado solamente por la herramienta. La herramienta debe usarse para demostrar que se ha hecho una contribución original al conocimiento; por ejemplo, a través de su uso, o de las ideas que puedan inferirse gracias a su uso.” (John W. Chinneck, “Cómo organizar una tesis”, Carleton University, Ottawa, Canadá)

A. Introducción

Este apartado contiene la definición de la metodología, la cual indica al investigador las tareas que deben realizarse para alcanzar los objetivos de estudio y ayudan a responder a las preguntas generales de conocimiento planteadas. Contiene la explicación de los aspectos metodológicos necesarios para crear un modelo que permita medir la eficiencia de los algoritmos de calendarización o de asignación de recursos en un sistema de grid computing. En este capítulo, se hace mención de la matriz básica de investigación, matriz de operacionalización y la matriz de relación entre los índices del estudio e hipótesis de investigación. Además, se definirá el tipo de estudio, los análisis practicados, la determinación de la muestra, la descripción de las fuentes de información y las técnicas de recolección de datos. Por último, se presenta la descripción de la metodología utilizada para los análisis practicados.

El tipo de investigación por realizar está altamente relacionada con las fuentes de información que existen, las cuales suministran los insumos para realizar el análisis de los datos y, de ese modo, rechazar o aceptar las hipótesis planteadas. De este análisis, se desprenderán los parámetros o indicadores por mejorar para que las economías de servicios

basados en grid computing sean más eficientes. Además, se definen los índices contempladas por esta investigación, así como las técnicas que serán utilizadas durante el estudio.

De la misma manera, se plantea nuevamente la hipótesis general y su relación con otros elementos tales como el problema de estudio, los objetivos generales y los objetivos específicos.

La hipótesis general de esta investigación es la siguiente: “La eficiencia de los sistemas de informática distribuida (IBECOD) puede determinarse mediante el análisis de la gestión económica (IGE), de la eficiencia en infraestructura (IEI) y de la gestión operacional del grid computing (IGO).”.

La validez de los instrumentos contenidos en este capítulo, permiten confirmar o rechazar las hipótesis, según los resultados inferidos en los análisis comparativos, entre experiencias de distintos mecanismos de calendarización aplicados en sistemas de informática distribuida.

B. Matriz básica de investigación

En esta sección, se presenta la matriz básica de investigación en la cual se relacionan los objetivos con cada una de las hipótesis planteadas. Los objetivos que se encuentran en el Cuadro N° 6, ya fueron presentados en la introducción contenida en el Capítulo I, pero ahora se le añaden las hipótesis de trabajo; con ello se logra comprender de mejor manera la relación funcional entre hipótesis y objetivos.

Cuadro N° 6 Planteamiento de hipótesis de trabajo según los objetivos de investigación

Objetivos de la investigación	Hipótesis
<p>Objetivo General</p> <p>Definir un modelo económico matemático que permita medir la eficiencia de la distribución de procesos y asignación de recursos en un ambiente de grid computing.</p> <p>El objeto de estudio es el Índice Básico de Eficiencia de Computación Distribuida (IBECOD), el cual resume la eficiencia de los grids de computadoras encargados del procesamiento de datos.</p> <p>Las índices independientes del estudio son: IGE (Índice de Gestión Económica), IEI (Índice de Eficiencia en Infraestructura) e IGO (Índice de Gestión Operacional).</p> <p>Los datos del estudio comprenden un periodo de cuatro años: del 2000 al 2003 considerando los elementos</p>	<p>H_I</p> <p>La eficiencia de los sistemas de informática distribuida (IBECOD) puede determinarse mediante el análisis de la gestión económica (IGE), de la eficiencia en infraestructura (IEI) y de la gestión operacional (IGO) del grid computing. Además, los índices independientes tienen influencia significativa en el índice IBECOD.</p>

Objetivos de la investigación	Hipótesis
que permite el desarrollo de esta investigación.	
<p>Objetivos específicos</p> <p>Analizar el impacto económico y en eficiencia que tienen los algoritmos de distribución de procesos y asignación de recursos en un ambiente de grid computing</p>	
<p>Analizar la forma como está siendo utilizada la infraestructura existente en las empresas que brindan servicios de grid computing y su aporte en la eficiencia.</p>	
<p>Analizar la gestión operacional de los sistemas de grid computing con el fin de determinar su impacto en la eficiencia.</p>	
<p>Analizar el comportamiento de indicadores económicos de los sistemas de grid computing comprendidos dentro esta investigación.</p>	

Fuente: Elaboración propia.

C. Matriz de operacionalización de índices

Este apartado presenta los índices independientes y dependientes, que serán sujetos de investigación y están directamente relacionadas con la hipótesis de investigación.

1. Índices dependientes

IBECOD: Índice Básico de Eficiencia de Computación Distribuida

Definición conceptual: Se entenderá por “eficiencia en los servicios de procesamiento de información” al concepto que define el rendimiento de las redes de informática distribuida y resume una serie de indicadores que lo determinan y llevan a medir este concepto en sus fases operacionales.

Definición instrumental: Se concebirá como el Índice Básico de Eficiencia de Computación Distribuida (IBECOD), aquel que resume la eficiencia de los grid de computadoras en cuanto al procesamiento de cargas de trabajo, el cual representa la el eje central de este estudio.

2. Índices Independientes

Se determinaron los índices que condicionan el índice dependiente (IBECOD), la cual se indicó anteriormente y que puede resumirse en la siguiente ecuación:

$$IBECOD = IGE + IEI + IGO + \xi$$

IGE: Índice de Gestión Económica.

IEI: Índice de Eficiencia en Infraestructura.

IGO: Índice de Gestión Operacional.

ξ : Margen de error.

Operacionalización: Entre estos indicadores, se consideraron los aspectos operacionales de un sistema de grid computing, los cuales se definen más adelante.

Instrumentalización: Complementariamente, cada uno de estos índices se explica mediante una serie de indicadores los cuales se definen más adelante.

Los indicadores de rendimiento del grid computing son mediciones de los aspectos de funcionamiento del grid y, generalmente, se derivan de la información recopilada de las bitácoras de los servidores. Estos registros son el resultado de la actividad diaria y reflejan la atención dada a las cargas de trabajo recibidas, así como el volumen de carga recibida durante ese periodo. Esta información nos señala los recursos utilizados y los ingresos percibidos por un grid en particular.

El proceso de manejo de esta información y atención de trabajos de procesamiento pueden estandarizarse (Standart Workload Format³¹), de manera tal que se comparen los procesos de manejo de cargas de trabajo llevados a cabo por distintos grids. De esa forma, podría establecerse un ranking que permita a los administradores determinar su posición con respecto a los sistemas de informática distribuida de la competencia.

Con esta fuente de datos estandarizada es posible analizar la información para convertirla en un conjunto de indicadores, los cuales pueden originar políticas o acciones que posibiliten la mejor planificación de los recursos de un grid en particular y mejoren su gestión operacional.

Este juego de indicadores se puede utilizar en negociaciones relacionadas con establecimiento de tarifas (un grid eficiente es más apreciado que uno ineficiente), decisión sobre inversiones (determinar si es necesario adquirir más hardware para que sea utilizado dentro del grid) y, además, para comparar la eficiencia general de un grid en particular con

³¹ El Standart Workload Format (SWF), especificado por Chapin y Cirne (1999), permite que bitácoras de trabajos de distintos sistemas de informática distribuida puedan ser expresados en un formato estándar, con lo cual se hace posible la comparación de sistemas de grid computing distintos.

las metas que la administración ha establecido en cuanto a eficiencia operacional y económica.

Los cambios en los indicadores de rendimiento de las operaciones de un grid se deben, en particular, a las distintas condiciones relacionadas con este modelo de procesamiento de información; por ejemplo: tráfico de información entre los distintos componentes del grid, velocidad de las líneas de transmisión de datos, calidad del hardware que forma parte del grid, entre otros.

Definitivamente, la obtención de una serie de indicadores de rendimiento no es suficiente para su mejora. Es necesario que estos valores sean comparados con las metas fijadas como objetivo, labor que debe ser realizada por los administradores del grid, sin descuidar en ningún momento la relación costo / rendimiento.

Según el razonamiento anterior, se presentan los índices de eficiencia de un sistema de computación distribuida:

- Índice de Gestión Económica (IGE)
- Índice de Gestión Operacional (IGO)
- Índice de Eficiencia en Infraestructura (IEI)

3. Índice de Gestión Económica (IGE)

Este índice brinda información sobre las finanzas de las empresas que proporcionan servicios de grid computing. Con base en esta información, se puede determinar si la aplicación del modelo de operación de la informática distribuida tiene un impacto en las finanzas de las empresas. Este índice comprende los siguientes indicadores por estudiar:

Cuadro N° 7 Índice de Gestión Económica (IGE)

Indicador	Unidad de medida
Ingresos en la empresa	Dólares
Evolución de los ingresos de las empresas a través de los años	Porcentaje
Porcentaje de cambio en la eficiencia a través de los años (tareas no atendidas)	Porcentaje
Cambio en los costos debido a la aplicación de distintos modelos de asignación	Dólares
Porcentaje de cambio en los costos debido a la aplicación de distintos modelos de asignación	Porcentaje

Fuente: Elaboración propia.

La descripción de los indicadores de gestión económica se muestra a continuación:

- **Ingresos de la empresa (INGEMP):** Los ingresos son expresados en dólares y se detallan mensualmente. Estos ingresos son el resultado de la suma de los montos cobrados por cada una de las tareas procesadas durante el mes.
- **Evolución de los ingresos de las empresas a través de los años (EVOING):** Este indicador se expresa en dólares y se calcula comparando la diferencia en monto que existe entre el periodo actual y el mismo periodo del año anterior.
- **Porcentaje de cambio en la eficiencia a través de los años (tareas no atendidas) (PORCAMEF):** Expresa el porcentaje de tareas que no han sido atendidas con éxito dentro del grid y se confronta el comportamiento de este indicador a través de los periodos estudiados.
- **Cambio en los costos debido a la aplicación de distintos modelos de asignación (CAMCOSTO):** Se mide en dólares y permite comparar el cambio producido en los costos debido a la aplicación de un modelo de administración y asignación de recursos.
- **Porcentaje de cambio en los costos debido a la aplicación de distintos modelos de asignación (PORCAMCOSTO):** Mide el cambio porcentual en los costos debido a la aplicación de distintos modelos de asignación de recursos y atención de tareas.

4. Índice de Gestión Operacional (IGO)

Los indicadores que componen el Índice de Gestión Operacional (IGO) son tomados en determinado tiempo durante la operación del grid y se puede tener una idea de su rendimiento en ese momento. Este tipo de indicadores permite que el rendimiento de distintos mecanismos de distribución de carga pueda ser comparado.

Este grupo de indicadores permiten determinar la eficiencia con que se están utilizando los recursos de hardware y software que componen el grid, su tiempo de operación y el tiempo de respuesta; así, se puede determinar cuál es la calidad de servicio que se está brindando: tiempo de atención y respuesta de los procesos enviados al grid, disponibilidad de los recursos para que sean utilizados por los clientes, tiempo de espera para que la carga de información recibida sea atendida, entre otros.

Cuadro N° 8 Índice de Gestión Operacional (IGO)

Indicador	Unidad de medida
Indicador de tiempo de espera por mes	Segundos
Indicador tiempo de espera por día de la semana	Segundos
Indicador de tiempo de ejecución de las tareas por mes	Segundos
Indicador cantidad de trabajos atendidos por mes	EP ³² /Proceso
Indicador de memoria utilizada por mes	Megabytes
Indicador cantidad de trabajos recibidos por día	EP/Día
Indicador cantidad de trabajos atendidos según hora del día	EP/Hora

Fuente: Elaboración propia.

La descripción de estos indicadores operacionales se lista a continuación:

³² EP (Elementos de Procesamiento) es la medida con la cual se define cuán exigente resulta un procesamiento en un caso determinado. Generalmente, están dados en diferentes unidades de velocidad, tales como MIPS o SPEC

- **Indicador de tiempo de espera por mes (ITEM):** Mide cuanto tiempo, expresado en segundos, tiene que esperar una tarea desde que es recibida en el grid hasta que se inicia su atención. En este índice, se toma segundos como unidad de trabajo.
- **Indicador tiempo de espera por día de la semana (ITEDS):** Permite identificar los patrones de uso del grid y, principalmente, conocer cuál es el día de la semana en que las colas de procesamiento aumentan. Si se conoce este comportamiento, será posible crear estrategias que permitan distribuir la carga de procesamiento de forma equitativa durante la semana.
- **Indicador de tiempo de ejecución de las tareas por mes (TETM):** Admite identificar cuanto tiempo tardan las tareas en ser procesadas dentro del grid; de esta forma se pueden tomar las medidas necesarias para aumentar la eficiencia de ejecución de las tareas.
- **Indicador cantidad de trabajos atendidos por mes (CTAM):** Posibilita determinar cuáles son los meses más ocupados y, de esa forma, preparar el grid para que pueda atender de forma adecuada las tareas enviadas en los meses de mayor actividad.
- **Indicador de memoria utilizada por mes (CMUM):** Propicia establecer en cuáles meses se requiere que el grid de procesamiento posea más o menos memoria, de forma tal que se puedan atender las tareas adecuadamente.
- **Indicador cantidad de trabajos recibidos por día (CTRD):** Facilita fijar cuáles son los días en que se presenta mayor demanda por parte de los consumidores. Con este indicador se puede establecer estrategias adecuadas para que las cargas de procesamiento sean atendidas adecuadamente en los días que presentan mayor actividad.
- **Indicador cantidad de trabajos atendidos según hora del día (CTAHD):** Permite prescribir cuál es el comportamiento de las tareas recibidas y saber cuáles son los periodos críticos de atención de tareas.

5. Índice de Eficiencia en Infraestructura (IEI)

El Índice de Eficiencia en Infraestructura (IEI) será investigado a la luz de efectos operacionales de otros conceptos dependientes tales como: manipulación de carga dentro del grid y tiempo de transmisión de los datos -desde el punto de origen hasta su destino dentro del grid de procesamiento). Ambos parámetros son de gran importancia en el transporte y procesamiento de datos y la operación del grid computing. El análisis instrumental de este índice nos lleva a incluir aspectos tales como: carga por transportar, capacidad de la red, distancia que se debe recorrer, entre otros.

Las variables que se utilizarán en el modelo matemático, que contribuye en el cálculo de la eficiencia en infraestructura de sistemas de grid computing, son descritas a continuación.

Cuadro N° 9 Índice de Eficiencia en Infraestructura (IEI)

Indicador	Unidad de medida	Siglas Utilizadas
Número de viajes necesarios para transportar la información	Números naturales	NV
Cantidad de información por transportar	MegaBits	K
Cantidad de información que puede manejar la red de transmisión de datos.	MegaBits	Mbp
Número de viajes en periodo de actividad (ida y retorno).	Números naturales	Nvp
Cantidad de información transportable en un periodo de actividad y en ambos sentidos.	MegaBits	Kp
Tiempo que es ocupada la red en un viaje completo	Segundos	Tv

Indicador	Unidad de medida	Siglas Utilizadas
de información		
Tiempo empleado por los paquetes de información en su operación dentro del grid.	Segundos	Tpg
Tiempo empleado por la información desde su salida hasta recibir los resultados del procesamiento.	Segundos	Tng
Tiempo de actividad de los nodos del grid	Horas	Tg
Cantidad de información manejada en el grid (carga de información y generación de resultados)	MegaBits	Kmg
Velocidad de procesamiento de la información dentro del grid	MegaBits	Vp

Fuente: Elaboración Propia

La descripción de estos indicadores se presenta a continuación:

- **Número de viajes necesarios para transportar la información (NV):** Este indicador considera la cantidad de viajes necesarios para transportar la información desde su origen en el cliente hasta llegar al grid computing. El número de viajes depende de la velocidad de la red y la cantidad de información por transportar.
- **Cantidad de información por transportar (K).** Establece la cantidad de información que debe ser transportada desde el cliente hasta el grid: cuanto mayor sea la cantidad de información tardará más en ser trasladada.
- **Cantidad de información que puede manejar la red de transmisión de datos (Mbp):** Este indicador mide la capacidad de transmisión de datos de la red: cuánto más información soporta la red, más rápida será la transmisión.

- **Número de viajes en periodo de actividad (Nvp):** Indica el número de viajes que pueden realizarse - de ida y retorno de información- en un periodo de actividad.
- **Cantidad de información transportable en un periodo de actividad y en ambos sentidos (Kp):** Mide la capacidad de la infraestructura para el transporte de información, establece cual es la máxima cantidad de información soportada por los medios de transmisión de datos.
- **Tiempo que es ocupada la red en un viaje completo de información (Tv):** Este indicador mide cuanto tiempo tarda el transporte de una carga de información desde su origen hasta el destino.
- **Tiempo empleado por los paquetes de información en su operación dentro del grid (Tpg):** Mide el tiempo -en segundos- que tardan las tareas dentro del grid; toma en cuenta el tiempo desde que se recibe la tarea hasta que el destinatario obtiene el resultado dentro del grid.
- **Tiempo empleado por la información desde su salida hasta recibir los resultados del procesamiento (Tng):** Contempla todo el tiempo transcurrido desde que el cliente envía la información que debe ser procesada hasta que recibe los resultados del procesamiento.
- **Tiempo de actividad de los nodos del grid (Tg):** Indica cuál es el tiempo de disponibilidad del grid. Por ejemplo, un grid puede estar disponible las 24 horas del día o únicamente durante un horario determinado.
- **Cantidad de información manejada en el grid (carga de información y generación de resultados) Kmg:** Mide la cantidad de información manejada dentro del grid para una tarea en particular, pues -además de las información que debe ser procesada dentro del grid- se han de crear estructuras temporales para almacenar los resultados parciales del procesamiento y, al final, se debe mantener el resultado hasta que sea enviado al cliente.
- **Velocidad de procesamiento de la información dentro del grid (Vp):** Este indicador mide la velocidad con la cual la información es procesada dentro del grid: un grid más rápido debería ser capaz de atender más tareas.

6. Matriz de Operacionalización

En el Cuadro N° 10, se muestra la relación entre las hipótesis, los índices y los indicadores mencionados anteriormente.

Cuadro N° 10. Relación entre hipótesis, índices e indicadores

Hipótesis	Índice Dependiente	Índice Independiente	Indicadores	Instrumento/Metodología
La eficiencia de los sistemas de informática distribuida (IBECOD) puede determinarse mediante el análisis de la gestión económica (IGE), de la eficiencia en infraestructura (IEI) y de la gestión operacional (IGO) del grid computing.	IBECOD	Índice de Gestión Económica (IGE)	<ol style="list-style-type: none"> Ingresos en la empresa Evolución de los ingresos de las empresas a través de los años Porcentaje de cambio en la eficiencia a través de los años Cambio en los costos debido a la aplicación de distintos modelos de asignación Porcentaje de cambio en los costos debido a la aplicación de distintos modelos de asignación 	<ol style="list-style-type: none"> Relación entre cantidad de tareas atendidas y las tarifas establecidas en el grid. Análisis históricos de los ingresos de las empresas. Análisis del desempeño en atención de tareas a través de los años. Aplicación de simulación para determinar el impacto económico debido al cambio en los algoritmos de administración de recursos. Análisis de comparaciones de los costos del grid bajo diferentes modelos de administración y asignación de recursos.
	IBECOD	Índice de Gestión Operacional (IGO)	<ol style="list-style-type: none"> Índice de tiempo de espera por mes Indicador tiempo de espera por día de la semana Indicador de tiempo de ejecución de las tareas por mes Indicador cantidad de trabajos atendidos por mes Indicador de memoria utilizada por mes Indicador cantidad de trabajos recibidos por día Indicador cantidad de trabajos atendidos según hora del día 	<ol style="list-style-type: none"> Este análisis corresponde a una evaluación cualitativa operando los distintos índices de operación del grid computing, que en su conjunto definen los intereses del negocio: datos, tiempos de espera, atención de los trabajos, tiempo de ejecución, cantidad de trabajos atendidos, memoria utilizada por mes, trabajos recibidos por día y atención de trabajos según hora del día. Se analizan los datos históricos que influyen en el cálculo de los indicadores de gestión operacional. Se analiza cada uno de estos indicadores y su comportamiento a través del tiempo.
	IBECOD	Índice de Eficiencia en Infraestructura (IEI)	<ol style="list-style-type: none"> Número de viajes necesarios para transportar la información Cantidad de información por transportar Cantidad de información que puede manejar la red de transmisión de datos. Número de viajes en periodo de actividad (ida y retorno). Cantidad de información transportable en un periodo de actividad y en ambos sentidos. Tiempo que es ocupada la red en un viaje completo de información Tiempo empleado por los paquetes de información en su operación dentro del grid. Tiempo empleado por la información desde su salida hasta recibir los resultados del procesamiento. Tiempo de actividad de los nodos del grid Cantidad de información manejada en el grid Velocidad de procesamiento de la información dentro del grid. 	<ol style="list-style-type: none"> Con el fin de calcular el IEI, se toman parámetros de configuración y datos de operación de los sistemas de grid computing incluidos en esta investigación. Cada uno de estos datos corresponden a los indicadores de eficiencia en infraestructura. Con esa información se obtiene el IEI, gracias a la aplicación de la Ecuación 11 definida en el Capítulo VI: "Construcción de un modelo para definir la eficiencia de los algoritmos de calendarización en informática distribuida".

Fuente: Elaboración propia, con base en los objetivos e hipótesis planteados en esta investigación.

D. Tipo de investigación

El método científico ha sido por excelencia la guía para la realización de investigaciones en todas las áreas. Sin embargo, existen distintas formas de identificar su aplicación en la investigación, la cual se puede clasificar de distintas maneras. En términos generales, la metodología de ésta investigación es una combinación de varios métodos para lograr el objetivo principal planteado, orientada por el Paradigma Positivista, mediante el cual se buscan las causas de los hechos o fenómenos y se utilizan mecanismos de cuantificación y comprobación de datos.

El enfoque positivista promueve la investigación empírica. Esta cuenta con un alto grado de objetividad; supone que si alguna cosa existe, implica la existencia de una cantidad y, por lo tanto, dicha cantidad se puede medir. Esto origina el desarrollo de investigaciones de tipo cuantitativo, las cuales se apoyan en mecanismos estadísticos e históricos. En esta investigación se realiza una interpretación de datos cuantitativos e históricos con el fin de explicar los fenómenos complejos, los cuales son sujetos de estudio y están orientados hacia las cualidades que pueden ser derivadas de ellos.

Meza (2003) establece que “el positivismo es una corriente de pensamiento cuyos inicios se suele atribuir a los planteamientos de Auguste Comte, y que no admite como válidos otros conocimientos sino los que proceden de las ciencias empíricas. De acuerdo con Dobles, Zúñiga y García (1998) la teoría de la ciencia que sostiene el positivismo se caracteriza por afirmar que el único conocimiento verdadero es aquel que es producido por la ciencia, particularmente con el empleo de su método. En consecuencia, el positivismo asume que sólo las ciencias empíricas son fuente aceptable de conocimiento”.

Con base en las definiciones anteriores, se puede establecer que la investigación en curso es dirigida por el enfoque positivista y tiene como objetivo principal la medición de

variables implícitas. Además, debido a su naturaleza y al enfoque filosófico, será necesario elaborar instrumentos que permitan realizar una medición cuantitativa de dichas variables.

Por otra parte, una investigación compleja no puede clasificarse exclusivamente en un tipo específico, sino que –usualmente- se persigue un objetivo y se utilizan distintas formas de investigación para lograrlo: es el caso de esta investigación. Según la clasificación establecida por Murillo (2006), esta investigación se basa en una estrategia combinada y –según el propósito perseguido- es de tipo mixto, en el cual se mezcla la investigación básica³³ y la investigación aplicada³⁴. Es básica, porque reúne elementos completamente teóricos, los cuales han sido planteados en el capítulo de Marco Teórico y se define un modelo matemático con el cual se puede medir la Eficiencia en Infraestructura (IEI) de los sistemas de grid computing. También es aplicada, pues con la definición del Índice Básico de Eficiencia de Computación Distribuida (IBECOD) se pretende su uso práctico y que los administradores comparen el rendimiento de sus sistemas de grid computing con sistemas similares. De esta forma, se pueden tomar las decisiones pertinentes para el aumento en la eficiencia del sistema de informática distribuida.

³³ Investigación básica: También recibe el nombre de investigación pura, teórica o dogmática. Se caracteriza porque parte de un marco teórico y permanece en él; la finalidad radica en formular nuevas teorías o modificar las existentes, en incrementar los conocimientos científicos o filosóficos, pero sin contrastarlos con ningún aspecto práctico. Murillo (2006)

³⁴ Investigación aplicada: Este tipo de investigación también recibe el nombre de práctica o empírica. Se caracteriza porque busca la aplicación o utilización de los conocimientos que se adquieren. Se encuentra estrechamente vinculada con la básica, pues depende de los resultados y avances de esta última. Esto queda aclarado si nos percatamos de que toda investigación aplicada requiere de un marco teórico. Sin embargo, en una investigación empírica, lo que le interesa al investigador, primordialmente, son las consecuencias prácticas. Murillo (2006)

Según la clase de medios utilizados para obtener los datos, esta investigación es documental³⁵, de campo³⁶ y experimental³⁷. Es una investigación documental, ya que durante su desarrollo se realizó una revisión bibliográfica exhaustiva, que incluyó libros, artículos y ensayos; para mayor detalle se puede consultar el apartado de Fuentes de información. Se clasifica de campo, porque para profundizar en el estudio del tema, se realizaron consultas electrónicas con expertos en el campo de estudio, con el fin de evitar una duplicidad de trabajos y a la vez contar con un guía experto que le aportara innovación y soporte a su desarrollo. Esta investigación es experimental, pues para probar teóricamente la definición del Índice de Eficiencia en Infraestructura (IEI) se realizaron simulaciones específicas con el fin de determinar el comportamiento de este índice en distintas situaciones. Un mayor detalle de estos experimentos se puede encontrar en el Capítulo VI “Construcción de un modelo para definir la eficiencia de los algoritmos de calendarización en informática distribuida”.

Por el nivel de conocimientos que se adquieren, esta investigación es exploratoria³⁸, descriptiva³⁹ y explicativa⁴⁰. Exploratoria, ya que se han realizado estudios previos sobre la

³⁵ Investigación documental: Este tipo de investigación es la que se realiza, como su nombre lo indica, apoyándose en fuentes de carácter documental, esto es, en documentos de cualquier especie. Como subtipos de esta investigación encontramos la investigación bibliográfica, la hemerográfica y la archivística; la primera se basa en la consulta de libros, la segunda en artículos o ensayos de revistas y periódicos, y la tercera en documentos que se encuentran en los archivos, como cartas, oficios, circulares, expedientes, etcétera. Murillo (2006)

³⁶ Investigación de campo: Este tipo de investigación se apoya en informaciones que provienen, entre otras, de entrevistas, cuestionarios, encuestas y observaciones. Como es compatible desarrollar este tipo de investigación junto a la investigación de carácter documental, se recomienda que primero se consulten las fuentes de carácter documental, a fin de evitar una duplicidad de trabajos. Murillo (2006)

³⁷ Investigación experimental: Recibe este nombre la investigación que obtiene su información de la actividad intencional realizada por el investigador y que se encuentra dirigida a modificar la realidad con el propósito de crear el fenómeno mismo que se indaga, y así poder observarlo. Murillo (2006)

³⁸ Investigación exploratoria: Recibe este nombre la investigación que se realiza con el propósito de destacar los aspectos fundamentales de una problemática determinada y encontrar los procedimientos adecuados para

eficiencia de los sistemas de grid computing. Sin embargo, éstos se han concentrado en aspectos operativos sin tomar en cuenta un enfoque global (IBECOD) que contemple aspectos de infraestructura, operación y análisis económico. De esta forma se ha establecido una nueva línea de investigación que podría ser explorada por futuras investigaciones. Esta investigación es descriptiva, pues no existe previamente un trabajo que reúna las características básicas sobre grid computing y su relación con los modelos económicos. El detalle de esta investigación descriptiva puede ser encontrado en el capítulo de Marco Teórico. También incluye aspectos explicativos, porque durante el análisis de resultados se realizaron deducciones e inducciones con base en los resultados obtenidos.

Se considera de carácter descriptivo, puesto que mide de forma independiente cada una de las variables explicativas de la investigación y, según Hernández (1991), en este tipo de investigación se selecciona las variables y cada una de ellas se mide independientemente para describir lo que se investiga. En la presente investigación, se elige los índices de Gestión Económica (IGE), de Gestión Operacional (IGO) y Eficiencia en Infraestructura (IEI). Cada una de estos índices se mide mediante fórmulas diferentes para obtener valores que permitan calcular el índice dependiente IBECOD.

elaborar una investigación posterior. Es útil desarrollar este tipo de investigación, porque al contar con sus resultados se simplifica abrir líneas de investigación y proceder a su consecuente comprobación. Murillo (2006)

³⁹ Investigación descriptiva: Mediante este tipo de investigación, que utiliza el método de análisis, se logra caracterizar un objeto de estudio o una situación concreta, señalar sus características y propiedades. Combinada con ciertos criterios de clasificación sirve para ordenar, agrupar o sistematizar los objetos involucrados en el trabajo indagatorio. Además, la investigación que hemos descrito anteriormente, puede servir de base para investigaciones que requieran un mayor nivel de profundidad. Murillo (2006)

⁴⁰ Investigación explicativa: Mediante este tipo de investigación, que requiere la combinación de los métodos analítico y sintético en conjugación con el deductivo y el inductivo, se trata de responder o dar cuenta de los porqués del objeto que se investiga. Murillo (2006)

Por otra parte, el presente estudio es explicativo, puesto que proporciona un sentido de entendimiento sobre la valoración de la eficiencia de los servicios prestados por empresas de grid computing. En esta investigación, se analiza la valoración desde un enfoque económico, operacional y de infraestructura; por otro lado, se generan nuevas contribuciones científicas, fruto del análisis y discusión de la información disponible. Por tanto, la capacidad explicativa de esta investigación doctoral la convierte en una tesis que sirva de base para futuros estudios.

Según la clasificación de los métodos de la ciencia establecidos por Cerbero (2006), esta investigación es de análisis, pues se utiliza la separación de las partes de un todo a fin de estudiar las variables por separado, así como examinar las relaciones existentes entre ellas. Es de síntesis, ya que se utiliza la reunión lógica de varios elementos dispersos en una nueva totalidad. Por lo tanto, el análisis y la síntesis están correlacionados y son inseparables. Según Cerbero (2006): “el investigador como ya se explicó antes efectúa suposiciones o conjeturas sobre la relación de tales o cuales fenómenos, pero la conexión entre ambos fenómenos no es evidente por sí misma. El investigador las sintetiza en la imaginación para establecer una explicación tentativa que será puesta a prueba.”.

Finalmente, las conclusiones y recomendaciones que se obtengan de esta investigación, podrán servir de base para medir los nuevos algoritmos de distribución de tareas en la informática distribuida.

E. Universo de estudio

El universo que involucra esta investigación, se concentra en el estudio y comparación de la eficiencia de tres grandes sistemas de informática distribuida, los cuales han sido escogidos por ser ejemplos representativos de los ambientes reales de operación, en lo que a informática distribuida se refiere.

La evaluación se concentra en la comparación objetiva de índices de eficiencia de distintos sistemas de informática distribuida.

Este universo de estudio está delimitado por los sistemas de informática distribuida OSC Linux Cluster y SDSC Blue, los cuales atienden –anualmente– más de 500,000 tareas o trabajos de procesamiento de información durante el periodo que cubre del año 2000 al 2003.

F. Determinación de la muestra

Según el alcance y las características propias de esta investigación sobre datos históricos y estimaciones de eficiencia, los datos muestrales provienen –especialmente– de la base de datos de referencia, los cuales han sido seleccionados de forma cuidadosa, con el fin de utilizarlos con la mayor objetividad y contenido científico formal.

Debido a que en esta investigación se propone un modelo de evaluación de la eficiencia, es necesario que los datos empleados para evaluar dicho modelo sean fiables. Por eso se utilizará la base de datos de Dr. Feitelson (2006), ya que ha sido utilizada como datos de referencia por los autores que analizan el comportamiento de algoritmos en un ambiente de informática distribuida.

Cabe destacar que los datos seleccionados de esta fuente poseen autenticidad, veracidad y formalidad requerida; son adecuadas para generar análisis amplios y propios, los cuales se requieren por la naturaleza de esta investigación.

El periodo de la base de datos incluye información sobre sistemas de grid computing y se concentra en el periodo que comprende del año 2000 al año 2003. Se recalca que los datos seleccionados de dicha fuente presentan autenticidad, veracidad y formalidad requerida;

además, son adecuadas para generar criterios de comparación amplios y propios, tales como los requeridos para un estudio de la naturaleza de esta investigación. De los años de estudio se extrae una serie de datos estadísticos correspondientes a la industria de la informática distribuida circunscrito en el contexto del comportamiento de los índices sujeto de estudio que se utiliza para definir el índice de eficiencia de informática distribuida propuesto en esta investigación.

De esta forma, los datos estadísticos e históricos que se obtengan del periodo muestral, se aplicarán de la siguiente forma:

- En el análisis de índices de operación, se está considerando datos históricos y promedios estadísticos que cubren –totalmente- la operación de los sistemas de informática distribuida estudiados; su representación está dada en tablas, cuadros y gráficos.
- En el análisis económico, se está considerando datos históricos de ingresos calculado con base en las bitácoras de operación de los sistemas de grid computing.

G. Fuentes de información

El alto nivel de investigación requerido en esta tesis doctoral y el alto contenido de datos estadísticos demandados, provocan que el investigador deba recurrir a fuentes de información primaria y secundaria confiables, con el fin de utilizar las que generen objetividad y alto contenido científico al estudio realizado. Estas fuentes son cruciales para alcanzar los objetivos de la investigación y para establecer los análisis incorporados a ella. Durante el proceso de elaboración de este trabajo, se lleva a cabo una revisión exhaustiva de fuentes bibliográficas de autores internacionales; incluyen libros, revistas, artículos, tesis, entre otros. Para ello son visitadas bibliotecas del país tales como las de la Universidad de Costa Rica, Instituto Tecnológico de Costa Rica y Universidad Latinoamericana de Ciencia

y Tecnología; además, se consultaron bibliotecas internacionales como la ACM Digital Library y la IEEE Digital Subscription Library.

De la misma forma, se ha incluido información proveniente de eventos relacionados con el tema de estudio tales como seminarios y congresos, así como revisión extensa en sitios de la Red Internet. De los congresos y seminarios se incluyen los siguientes:

- IEEE International Conference on Cluster Computing, 2-3 diciembre, 1999, Melbourne, Australia.
- First International Workshop on Grid Computing (GRID 2000), 17 diciembre, 2000, Bangalore, India.
- First IEEE International Symposium on Cluster Computing and the Grid (CCGrid 2001), 15-18 mayo, 2001, Brisbane, Australia.
- IEEE International Conference on e-Science and Grid Technologies (e-Science 2005), 5-8 diciembre, 2005, Melbourne, Australia.

Las características de esta investigación, la informática distribuida y la atención de grandes cantidades de información determinan una selección cuidadosa de las fuentes de datos. Sin embargo, la especialización del tema determina que los referentes de información tengan su origen en sistemas de informática distribuida internacionales, operados por centros de investigación de gran prestigio, los cuales son utilizados como apoyo en las investigaciones científicas.

Dado que la investigación expresa veracidad y formalidad en los datos, durante su etapa de análisis se consideró sistemas de informática distribuida representativos y se utilizaron las bitácoras de trabajo recolectadas por el Dr. Feitelson (2006), las cuales son fuente primaria puesto que contienen todos los datos de operación reales y veraces de los sistemas de grid computing analizados en esta investigación. La fuente de datos utilizada para realizar este análisis utiliza el Standard Workload Format especificado por Chapin y Cirne (1999), este

modelo fue definido con el fin de facilitar el uso de bitácoras de trabajo de centros de procesamiento de datos. Este estándar posee las siguientes características:

- Las cargas de trabajo son almacenadas en un archivo de texto.
- Cada carga de trabajo es representada por una línea dentro del archivo.
- Se utiliza el mismo formato, tanto para registrar las cargas de trabajo como para describir las características de los recursos que posee la infraestructura del grid.

En de este análisis, se incluirán las siguientes infraestructuras de grid:

- SDSC Blue Horizon (San Diego Supercomputer Center).
- OSC Linux Cluster (Ohio Supercomputer Center).

El San Diego Supercomputer Center SDSC Blue Horizon es un sistema compuesto por un total de 144 nodos cada uno con 8 procesadores, dentro del estudios se incluyen 250,440 cargas de procesamiento. Los procesadores están conectados entre sí mediante una estructura de memoria compartida. Las cargas de trabajo son atendidas por un conjunto de colas, dentro de las cuales se encuentran:

Tabla N° 19 Colas del sistema SDSC Blue Horizon

Cola	Tiempo límite	Límite de nodos
Interactiva	2 horas	8
Expres	2 horas	8
Alta	36 horas	Sin límite
Normal	36 horas	Sin límite

Fuente: Feitelson (2006).

Finalmente, El OSC Linux Cluster está compuesto por 32 nodos con cuatro procesadores y 25 nodos con dos procesadores, para un gran total de 178 procesadores. El detalle se puede observar en la Tabla N° 20.

Tabla N° 20 Composición del OSC Linux Cluster

Grid	Cantidad de nodos	Características de los nodos	Total de procesadores
01	32	4 procesadores cada nodo	128
02	25	2 procesadores cada nodo	50
	Total		178

Fuente: Feitelson (2006).

Paralelo a lo anterior, se consultan distintas fuentes de datos tales como las utilizadas en los experimentos del Centro de Investigación sobre Grid Computing de la Universidad de Melbourne Australia. Dichos datos se encuentran disponibles en el sitio web de este centro de investigación (www.gridbus.org).

Las publicaciones, informes y documentos emitidos por investigadores internacionales cumplen un papel primordial en esta investigación. Por lo tanto, gran parte de la información que incluye este trabajo proviene –principalmente- de fuentes primarias elaboradas por especialistas vinculados con el tema planteado y realizadas en distintos lugares del mundo (Estados Unidos, Reino Unido, India, Australia, Japón, Corea del Sur, entre otros), cuyas experiencias han permitido extraer de ese contexto lecciones de interés para la elaboración de esta tesis.

Asimismo- se acude a revistas, reportes científicos de investigadores internacionales y empresas multinacionales que brindan servicios de grid computing. Entre las revistas se encuentran la IEEE (Institute of Electrical and Electronics Engineers), la ACM

(Association for Computing Machinery), Springer Verlag, American Banker, Computer World, eWeek, Securities Industry News, Mechanical Engineering, Technology Review, America's Network, InfoWorld, MIT Sloan Management Review, entre otras. Dentro de las empresas internacionales consultadas se encuentra Sun Microsystems (www.sun.com), Oracle Corporation (www.oracle.com), Internacional Business Machines IBM (www.ibm.com), Microsoft Corporation (www.microsoft.com), Hewlett-Packard (www.hp.com), Entropia (www.entropia.com), Platform Computing (www.plaform.com), United Devices (www.ud.com), DataSynapse (www.datasynapse.com), Avaki (www.avaki.com), Globus Project (www.globus.org), entre otros. Dentro de los reportes científicos se consultaron los siguientes: "Grid Computing - Projected Impact on North Carolina's Economy and Broadband Use through 2010" publicado por el Dr. Robert B. Cohen, los proyectos de investigación del Dr. Buyya (www.buyya.com) - The Gridbus Project, Grid Economy and Nimrod-G Scheduler, World Wide Grid, Virtual Laboratory, Grid Simulation (GridSim), PARMON – así como la tesis del Dr. Buyya “Economic-based Distributed Resource Management and Scheduling for Grid Computing”.

La totalidad de las fuentes de información puede ser consultada en el apartado de bibliografía.

H. Sujeto de estudio

La eficiencia de los sistemas de informática distribuida es el sujeto de estudio. Para ello, se ha diseñado el IBECOD (Índice Básico de Eficiencia de Computación Distribuida), el cual involucra la Gestión Económica (IGE), la Gestión Operacional (IGO) y la Eficiencia en Infraestructura (IEI). Cada uno de estos componentes está relacionado uno con el otro y los convierten en sujeto de investigación.

I. Técnicas de obtención de la información

En la investigación que se plantea, las técnicas para obtener información consisten básicamente en la revisión estadística y documental electrónica. La consulta a expertos por medio de entrevistas físicas y electrónicas, y la técnica de la observación está ligada con apreciaciones y discusiones que se mantienen permanentemente durante la investigación.

1. Recopilación documental

Las técnicas aplicadas para reunir la información provienen de fuentes primarias (bitácoras de los sistemas de informática distribuida) y secundarias, que residen en el análisis, formación y capacitación del investigador en el tema; las continuas consultas por Internet a especialistas y una variada colección de libros, revistas especializadas y artículos científicos, fuesen éstos de informática, economía o una combinación de ambos.

Otra técnica para adquirir información, la constituye la consulta a trabajos realizados en Estados Unidos de América, creados por empresas consultoras dedicadas a la elaboración de investigaciones y documentos relacionados con el tema. Entre estas empresas se encuentran Gartner Group y Platform Computing.

Las recopilaciones iniciales de información (materiales de referencia, físicos y virtuales) fueron analizadas, priorizadas y jerarquizadas de forma que pudieran entregar su aporte, argumentos y distintas posiciones, con respecto a los temas de esta investigación: los modelos económicos aplicados a la informática distribuida y el análisis de eficiencia de operación. Una gran cantidad de materiales, se compilaron gracias a la ayuda de los investigadores de la Universidad de Melbourne en Australia.

Así mismo, se da un monitoreo internacional a través de Internet con el fin de realizar una revisión de informes e investigaciones, con lo cual se extrae gran número de lecciones acerca de experiencias sobre el tema planteado y se da sentido a la estructura lógica y

científica de esta investigación. A través de la consulta por Internet, se obtiene gran cantidad de información de diversas instituciones y autores, que enriquecen el conocimiento del investigador y le permiten mayor objetividad analítica para el desarrollo del estudio planteado en esta investigación.

2. Revisión estadística

Para la realización de esta investigación, se elabora una guía de análisis basada en un conjunto de indicadores anotados en la matriz de operacionalización de variables.

La instrumentalización para los indicadores está diseñada de tal forma que se anoten – respectivamente- las cifras estadísticas en monto o porcentajes para cada uno de los años de estudio, con la finalidad de contar con la información necesaria para hacer un análisis robusto y confiable en el estudio.

3. Entrevistas y consulta a expertos

A través de los avances de esta investigación, se han confrontado opiniones con expertos internacionales tales como el Dr. Rajkummar Buyya; además, se ha contado con la supervisión y control del Dr. Leopoldo Santibáñez, lo cual ha sido fundamental en el trato de aspectos relacionados con modelos de eficiencia.

Objetivo

La consulta con expertos tiene como objetivo validar, mediante la supervisión y el criterio matemático, la utilidad del modelo de medición de eficiencia planteado en la investigación, para su utilización como herramienta de análisis y comparación de sistemas de grid computing.

Perfil

El perfil de los expertos permite realizar consultas a personas con grandes conocimientos de la industria de grid computing y el entorno en que esta actividad se desarrolla a nivel mundial. En ese sentido, según las variables de estudio, se busca que los expertos tengan no menos de cinco años de trabajo continuo en la industria de grid computing y cumpla como mínimo uno de los puntos del siguiente perfil seleccionado en lo referente a conocimientos:

1. sobre evaluación y definición de modelos de asignación de tareas en grid computing.
2. sobre definición y utilización de herramientas de simulación.
3. estadísticos sobre el comportamiento operacional de sistemas de grid computing.
4. sobre modelos de eficiencia.

Información solicitada

Se harán preguntas acerca de distintos modelos de eficiencia aplicados en entornos de transporte de carga y de información, así como sobre mecanismos de evaluación de infraestructura, mediciones de eficiencia operacional y modelos económicos aplicados al grid computing.

La información solicitada también incluye temas sobre técnicas de simulación, las cuales son vitales para la evaluación de modelos de eficiencia.

Confidencialidad

La información recolectada se clasifica en datos de dominio público y datos confidenciales. Los primeros incluyen aquellos obtenidos de los centros de investigación de las universidades. Los segundos corresponden a la información sobre tarifas e ingresos percibidos por los centros de grid computing, es decir, se protege la identidad de la persona física o jurídica que suministra la información: los aportes no se detallan de forma individual sino como un todo.

Metodología de la consulta

Durante el desarrollo de la investigación, se dirigen preguntas a los expertos lo cual facilita una conversación interactiva que da respuesta a las interrogantes planteadas. El planteamiento de dudas y preguntas tiene como finalidad guiar la discusión, buscar mayores detalles o desarrollar temas relacionados entre el experto y el investigador. De la interacción resultante, se toman notas y se incluyen en el desarrollo de la investigación.

4. Simulación

Debido a que en esta investigación se propone un modelo de medición de la eficiencia, es necesario que dicho modelo sea probado en sistemas de grid computing, con el fin de determinar si su planteamiento es el adecuado. Los sistemas de grid computing constituyen mecanismos de procesamiento altamente complejos y su acceso es limitado. Por eso, durante el afinamiento del modelo planteado, se recurre a la simulación antes de aplicarlo en los sistemas reales.

Del resultado de las simulaciones, se obtienen datos necesarios para afinar el modelo propuesto.

5. Discusiones y apreciaciones

El intercambio de opiniones, ideas, apreciaciones y diferentes puntos de vista con aquellas personas, directa o indirectamente, se ha visto involucrado en el tema de estudio; permiten que el investigador cuente con la supervisión y contacto permanente con expertos en el campo metodológico, económico, estadístico, matemático y computacional. De esta manera, se unen fuerzas independientes que contribuyen integralmente al fortalecimiento y corrección de aspectos de forma o fondo que se dieran durante el desarrollo de esta investigación.

J. Estado de la cuestión

Hasta el momento, existen varias publicaciones correspondientes a información recibida desde el extranjero, enfocadas en el estudio de la informática distribuida; entregan una información importante sobre la evolución histórica de la informática distribuida y la aplicación de modelos económicos en ésta. En todos estos artículos, libros, revistas y textos -entre otros- existen aportes y análisis que explican -mediante experimentación, teoría o justificaciones matemáticas- orígenes y consecuencias derivadas de la aplicación de modelos económicos en la informática distribuida.

En este sentido, es importante evaluar aquellas publicaciones que contribuyan con aportes teóricos o experimentales respecto al tema de estudio y que sirven de complemento a esta investigación. En el Cuadro N° 11, se muestra una lista de las publicaciones relacionadas con este asunto.

Cuadro N° 11 Investigaciones relacionadas con el tema de estudio

Autor	Localidad	Objetivo General	Metodología	Conclusiones
Srikumar Venugopal A grid service broker for scheduling distributed data-oriented applications on global grids	Melbourne Australia	Proponer y desarrollar un calendarizador que sirve de intermediario en el acceso a recursos distribuidos	Propuesta del calendarizador y realización de pruebas de ejecución utilizando la física aplicada a la energía.	Los calendarizadores de grid computing pueden ser adaptados para trabajar con datos geográficamente distribuidos y, a la vez, incluir mecanismos económicos de calendarización.
Anthony Sulistio, Gokul Poduval Constructing a grid simulation with differentiated network service using GridSim	Singapur, Australia	Revisar la forma cómo se pueden extender los sistemas de simulación para incorporar mecanismos de calidad de servicio.	Análisis de escenarios reales para luego simularlos y determinar si los resultados de la simulación se ajustan a los parámetros reales esperados.	Modificación de los mecanismos de simulación para que estos modelen, adecuadamente, el comportamiento de los sistemas distribuidos reales.
Marcos Dias de Asunción	Australia	Analizar los requerimientos de comunicación para	Creación de un ambiente de simulación en la cual se realizaron distintos	El método de subasta Inglesa necesita mayores requerimientos de comunicación, mientras que las subastas dobles continuas presentan menos

Autor	Localidad	Objetivo General	Metodología	Conclusiones
An evaluation of communication demand of auction protocols in grid environments		cuatro mecanismos de subastas: Primer Precio Sellado, Inglesa, Holandesa y Doble Subasta	experimentos con el fin de cumplir con los objetivos planteados.	necesidades de comunicación. Se demuestra que los mecanismos de subasta inglesa y holandesa llevan al mismo resultado en el precio de la subasta.
A Fernández Servicios de asignación y planificación de recursos grid	España	Presentar y estudiar técnicas para la ejecución eficiente de tareas dentro de un ambiente de grid computing.	Creación de un agente de calendarización y aplicarlo en sistemas reales para el análisis su comportamiento.	Las aplicaciones interactivas y de asignación de recursos requieren de un tiempo de respuesta rápido para que el usuario pueda estar satisfecho con los resultados.
Rajkummar Buyya An architecture for a resource management and scheduling system in a global computation grid	Australia	Propone la creación de un calendarizador de recursos distribuidos basados en conceptos económicos.	Se implementó el sistema Nimrog/G y se probó en sistemas reales mediante el uso de sistemas reales distribuidos alrededor del planeta.	Presenta los distintos parámetros económicos que afectan los mecanismos de calendarización distribuidos y la creación de GRACE (Grid Architecture for Computational Economy).
Dani Goldberg	Estados Unidos	Analizar mecanismos de planificación de	Construcción de robots con sensores en tiempo real para	Demuestran que las estrategias de asignación de tareas y recursos basados en mecanismos

Autor	Localidad	Objetivo General	Metodología	Conclusiones
Market-Based multi-robot planning in a distributed layered architecture		capacidad de robots con base en mecanismos económicos de subastas.	probar los mecanismos de subastas propuestos.	económicos, tienen gran impacto en el rendimiento de los robots y pueden aplicarse en gran variedad de dominios y condiciones.
Nathaniel Rockwood Economic Allocation of Computation Time with computation markets	Estados Unidos	Analizar la asignación de recursos de procesamiento mediante el uso de un enfoque económico basado en mercados.	Analiza y experimenta con distintos modelos económicos para comparar el rendimiento obtenido por cada uno de estos en la asignación de recursos.	Los mecanismos de subastas satisfacen todas las demandas que necesitan los sistemas de calendarización.
Carsten Ernenmman Economic Scheduling in Grid Computing	Alemania	Analizar el impacto de los modelos económicos en las tareas de calendarización.	Implementación y programación de los modelos propuestos en sistemas computacionales reales con el fin de analizar los resultados	Demuestra que los mecanismos económicos ayudan a obtener mayor rendimiento que los mecanismos convencionales.

Fuente: Elaboración propia, según datos consultados.

K. Alcances del estudio

Desde el punto de vista conceptual, esta investigación se caracteriza por los aspectos metodológicos planteados. La investigación, como tal, está condicionada por la realidad misma de los hechos que contemplan criterios propios de quienes participan en la industria de grid computing fuera del país y, en este sentido, la investigación se ve afectada por estos.

En cuanto al alcance geográfico, se reduce a sistemas de grid computing de Estados Unidos de América. La elección de este país se debe a que posee la mayor concentración de poder computacional mundial con un 60%⁴¹, lo cual lo convierte en un líder. Por lo tanto, los sistemas de grid computing presentes en Estados Unidos son un representativo del estado del arte de esta tecnología a nivel mundial.

El alcance temporal, el análisis teórico y estadístico de la investigación que se plantea comprende el período comprendido desde el año 2000 hasta el 2003.

Además, los resultados de esta investigación servirán como punto de inicio para futuras investigaciones en materia de evaluación y categorización de sistemas de grid computing; porque al existir una forma de medir su eficiencia, es posible comparar los resultados de este trabajo con los obtenidos por investigaciones posteriores, de forma tal que se ratifique o no el modelo presentado.

La presente tesis se centra en la determinación por medio de un conjunto de indicadores de la eficiencia de sistemas de grid computing. De tal forma, se puedan comparar los modelos de asignación de tareas, administración de recursos y de mecanismos de operación de distintos sistemas de grid computing.

⁴¹ Según el High Performance Computing University of Tennessee, 2004

El modelo matemático que se propone podría convertirse en un medio para la valoración de eficiencia en sistemas de procesamiento con características distribuidas; también, en un insumo para la implementación de estrategias de competencia y mejoramiento económico de los sistemas de grid computing, de forma tal que se regulen las tarifas según su eficiencia.

Asimismo, es un aporte metodológico para comprender las relaciones entre los indicadores que explican el modelo matemático diseñado en materia de medición de eficiencia (IBECOD) y que se puede extender como modelo para el análisis de sistemas de grid computing en otra región de grid computing en particular.

L. Limitaciones experimentadas en el estudio

Conceptuales: Esta investigación utiliza las herramientas de la disciplina económica y matemática con el fin de alcanzar los objetivos del estudio. Sin embargo, el criterio de análisis e interpretación propio es un elemento presente y no estará exento de desacuerdos o discrepancias con las opiniones y juicios de otros especialistas en el tema.

Geográficas: Dada la gran cantidad de participantes que interactúan en la industria de grid computing, las conclusiones a las que llegue la investigación con respecto a los modelos de eficiencia quedan restringidas al alcance geográfico de Estados Unidos de América.

Temporal: Los indicadores de sistemas de grid computing analizados comprenden datos de los años 2000 al 2003, con los cuales se originan las bitácoras de funcionamiento de cada uno de los sistemas incluidos en el estudio.

Confidenciabilidad y disponibilidad: Los datos analizados en el estudio presentan limitaciones de disponibilidad, esto debido a que cuando la tecnología de grid computing inició las bitácoras de estos sistemas eran accesibles de forma pública y gratuita. Sin embargo, conforme se empezó a comercializar la tecnología, estas bitácoras ya no fueron publicadas y fueron clasificadas de carácter estratégico por los proveedores. Esto por esta razón que los datos analizados en esta investigación son aquellos que eran públicos en el momento en que realiza la misma.

De análisis. Debido a que el objetivo del presente trabajo es identificar el modelo de eficiencia de la informática distribuida, ciertos elementos suyos han sido considerados de forma general. Por tanto, los análisis se efectuaron con el fin de ilustrar ciertos aspectos claves de los grids relacionados con la eficiencia de su operación y no se describen a completitud todos los aspectos de implementación del grid y su operación. La base del análisis es principalmente cuantitativa, pero estos son parciales puesto que no se describen ni estudian la completitud de los componentes de un sistema de informática distribuida.

Fuentes de información. Las fuentes utilizadas resumen la información disponible sobre el tema de estudio, para situaciones que no necesariamente representan la situación real de la tecnología de grid computing en Costa Rica.

M. Tipos de análisis practicados

Los análisis realizados para la demostración de las hipótesis son los siguientes:

- **Análisis económico:** Este desarrollo se logra al establecer relaciones de comparación entre indicadores financieros de las empresas que brindan servicios de

grid computing y se estudia el impacto que ha tenido ésta sobre las finanzas. La revisión permite al investigador identificar el nivel de incidencia en las finanzas de las estrategias de grid computing.

- **Análisis operacional:** La influencia que tienen los indicadores operacionales sobre la eficiencia de un sistema de grid computing es estudiada en este análisis, el cual se concentra en comparar los indicadores de distintos modelos de operación. Técnicamente, en este estudio también es posible comparar las tareas atendidas en los sistemas de grid computing y la diferencia existente entre los distintos modelos de operación.

N. Descripción de metodología para los análisis practicados

Este apartado detalla los esquemas de evaluación que se aplican dentro del análisis de este trabajo, con el propósito de que la información generada sea suficiente y permita emitir criterio firme acerca del tema en estudio.

Para lograr los objetivos descritos en esta investigación, se procede con los siguientes análisis:

1. Representación histórica de datos

Los datos y su representación forman parte importante de esta investigación. Se representa mediante cuadros y tablas, los cuales son la fuente principal para la creación de gráficos que permitan mostrar los datos de forma que se aprecien de forma más clara y ordenada. Además, los gráficos sirven como un medio de análisis de la información recolectada y para evaluar el comportamiento de los indicadores establecidos con respecto al tema de estudio.

Específicamente, la técnica de gráficos será utilizada para representar curvas históricas de cifras que resumen el Índice de Gestión Operacional (IGO), el Índice de Gestión Económica (IGE) y el Índice de Eficiencia en Infraestructura (IEI), así como para representar el comportamiento anual para los indicadores sujeto de estudio comprendidos en el alcance temporal de la investigación.

2. Metodología para el análisis de indicadores

En este apartado se presenta la forma como se calcularán los distintos indicadores utilizados para medir los índices planteadas según la hipótesis de trabajo descrita anteriormente.

El modelo propuesto está definido de la siguiente forma:

$$IBECOD = IGE + IEI + IGO + \xi$$

IGE: Índice de Gestión Económica.

IEI: Índice de Eficiencia en Infraestructura.

IGO: Índice de Gestión Operacional.

ξ : Margen de error.

El IBECOD debe dar como resultado numérico que indique la eficiencia global del sistema de grid computing, de esta forma es posible comparar distintas empresas y establecer un ranking en el cual se pueda ubicar una empresa en una posición en particular. Con el establecimiento dentro del ranking, los administradores de las empresas pueden determinar su posición con respecto a la competencia y tomar las medidas necesarias dependiendo de la situación en la cual se encuentren.

Cada una de los índices (IGE, IEI e IGO), se calculan mediante la asignación de pesos relativos a los indicadores (Ind_i) que los componen. En esta investigación, las ponderaciones para cada componente (PC_i) tienen un peso. Después de que cada indicador es calculado, se procede de la siguiente manera:

1. Si la máxima ponderación deseada es el valor más alto para el indicador entonces

$$PC_i = \frac{Ind_i - \min(Ind_i)}{\max(Ind_i) - \min(Ind_i)}$$

2. Si la máxima ponderación deseada es el valor más bajo para el indicador entonces

$$PC_i = \frac{\max(Ind_i) - Ind_i}{\max(Ind_i) - \min(Ind_i)}$$

Con las fórmulas anteriores, se lograr expresar un indicador en relación al cociente que existe entre el valor obtenido por un indicador en un sistema en particular con respecto al valor relativo máximo obtenido en la industria, donde Ind_i es el valor observador para el indicador i en el sistema i y $\max(Ind_i)$ es el valor máximo obtenido para ese indicador en el conjunto de sistemas estudiados.

Una vez que se ha asignado ponderaciones a cada uno de los componentes, se procede al cálculo del índice compuesto por los indicadores ponderados. El valor del índice corresponderá a la suma de las ponderaciones de los componentes, expresado como:

$$\text{Índice} = \sum_{i=1}^n PC_i$$

Por tanto, con base en lo expuesto anteriormente, se calcula IBECOD de la siguiente forma:

$$IBECOD = \sum_{i=1}^n PC_i + \sum_{j=1}^m PC_j + \sum_{k=1}^o PC_k$$

Donde,

n es el número de componentes del Índice de Gestión Económica (IGE)

m es el número de componentes del Índice de Eficiencia en Infraestructura (IEI)

o es el número de componentes del Índice de Gestión Operacional (IGO)

PC_i es el peso asignado al componente i del Índice de Gestión Económica (IGE)

PC_j es el peso asignado al componente j del Índice de Eficiencia en Infraestructura (IEI)

PC_k es el peso asignado al componente k del Índice de Gestión Operacional (IGO)

Una vez calculado el IBECOD para cada sistema, se puede establecer un ranking donde el primer lugar lo tiene el sistema que haya obtenido la puntuación más alta en el índice IBECOD.

3. Metodología para los análisis realizados utilizando simulación

Para la comparación de distintos modelos de calendarización, se utilizará una herramienta de simulación. En esta investigación, se recurre al software GridSim desarrollado en la Universidad de Melbourne y cuya descripción completa se encuentra en el Capítulo VII.

La evaluación de distintos modelos de asignación de recursos requiere que éstos se ingresen dentro del simulador y después se utilicen bitácoras reales para determinar su comportamiento. En este trabajo, se cuenta con la información de las bitácoras para cada uno de los sistemas; con el fin de probar cada modelo, se utilizará una muestra representativa del total de tareas de cada uno de los sistemas.

Según Fernández (1996), cuando la población es finita, o sea, se conoce la cantidad total de datos de los cuales se desea obtener una muestra y se desea saber cuál es el número adecuado de datos que deben incluirse dentro de la muestra, se debe utilizar la siguiente fórmula:

$$n = \frac{N * Z_{\alpha}^2 * p * q}{d^2 * (N - 1) + Z_{\alpha}^2 * S^2}$$

Donde:

n es la tamaño de la muestra que se debe tomar

N es el tamaño de la población

Z_{α}^2 es el nivel de confianza que se desea, por ejemplo un 95% de confianza es 1.96^2

S_{ξ}^2 es la varianza encontrada en la muestra

En esta investigación para determinar la muestra, se utiliza un nivel de confianza del 95%.

Con base en estos parámetros, la cantidad de datos de muestras tomados son los siguientes:

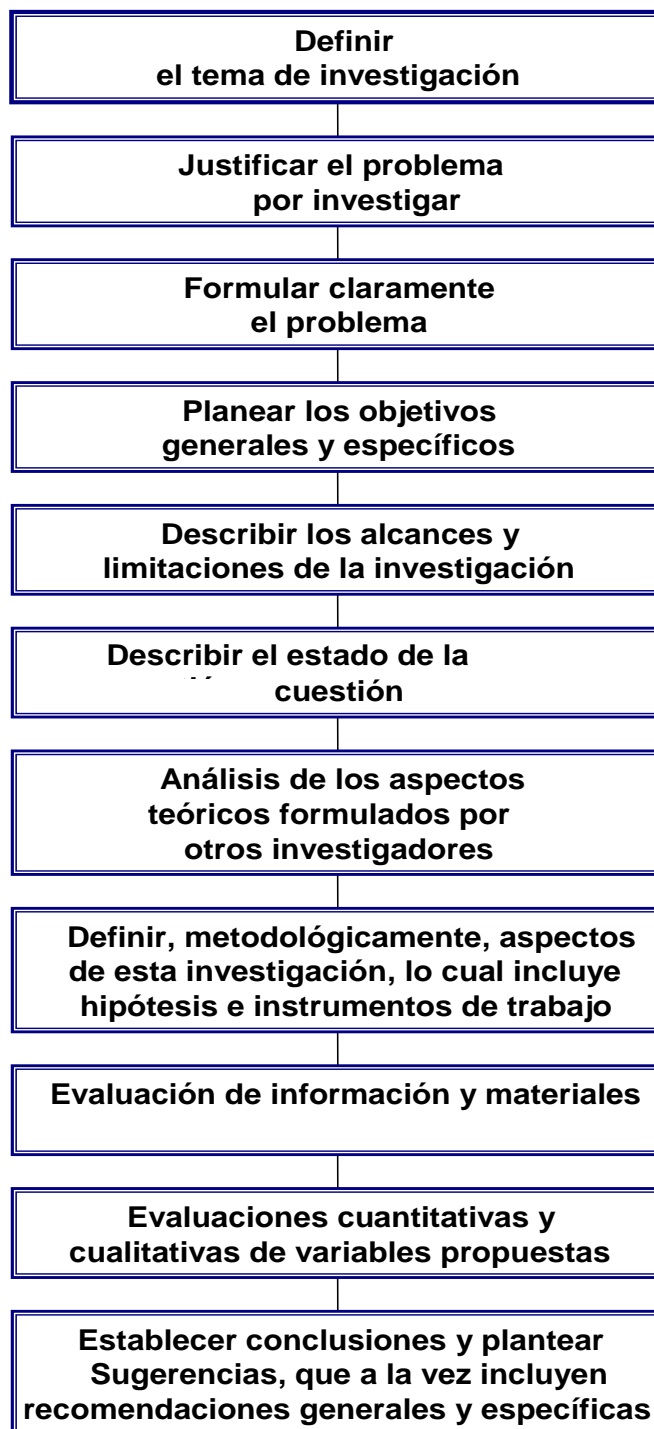
Cuadro N° 12 Muestreo aleatorio de tareas de procesamiento

Sistema	Población	Muestra Estimada	Muestra Aplicada
OSC Linux Cluster	36,097	1,036	1,500
SDSC Blue	243,314	1,062	1,500

Fuente: Elaboración propia.

Con base en los cálculos anteriores para la investigación, se debe utilizar una muestra de al menos 1,062 tareas de cada uno de los sistemas de grid computing (ver Cuadro N° 12). Puede observarse que el tamaño de la muestra aplicada es mayor que la estimada, lo cual aumenta la representatividad de la muestra con respecto a la población total.

O. Cronograma de investigación



V. Construcción de un modelo para definir la eficiencia en infraestructura de los algoritmos de calendarización en informática distribuida

A. Introducción.

En esta sección, se presenta el modelo con el propósito de definir la eficiencia en infraestructura de los algoritmos de calendarización en informática distribuida. Para el desarrollo de este modelo, se tomó como base el modelo de Índice de Eficiencia Portuaria, propuesto por el Dr. Leopoldo Santibáñez (2004). El modelo del Dr. Santibáñez fue adaptado y extendido para que sea aplicable en la medición de la eficiencia en grid computing: este modelo puede ser consultado en el Anexo I.

B. Factores básicos del transporte y procesamiento de información

Los elementos principales que se deben considerar en un ambiente de informática distribuida son los siguientes:

- La cantidad de información que han de calendarizarse: los nodos del grid atenderán y procesarán la información que reciben.
- La distancia que deberá recorrer la información: la información por procesar debe enviarse desde el origen hacia el grid. Por tanto, se tomará en cuenta el tiempo que tarda esa información en llegar al grid que la procesará.

- La capacidad de la red, expresada en Mbps (Megabits por segundo). El tiempo que tarde el envío de la información desde el origen hacia el grid, afecta la calidad del servicio. Por ejemplo, si los algoritmos de calendarización asignan la información a un grid, cuya red es muy lenta, entonces el tiempo de respuesta será lento y afectará al dueño de la información.

Cada ciclo de transporte de información consta de dos partes: la primera corresponde al envío de la información que debe ser procesada en el grid, la segunda es el resultado del procesamiento que ha de viajar desde el grid hacia el dueño de la información. Se debe considerar que el envío inicial de información podría requerir varios viajes debido a las restricciones que imponga el medio de transmisión de red.

C. Definición del modelo matemático

Los parámetros por utilizar en el modelo matemático, que define la eficiencia de los algoritmos de calendarización en informática distribuida, son descritas a continuación.

Siglas Utilizadas	Significado de l parámetro del Modelo Matemático
NV	Número de viajes necesarios para transportar la información
K	Cantidad de información por transportar.
Mbp	Cantidad de información que puede manejar la red de transmisión de datos.
Nvp	Número de viajes en periodo de actividad (ida y retorno).
Kp	Cantidad de información transportable en un periodo de actividad y en ambos sentidos.
Tv	Tiempo que es ocupada la red en un viaje completo de información

T _{pg}	Tiempo empleado por los paquetes de información en su operación dentro del grid.
T _{ng}	Tiempo empleado por la información desde su salida hasta recibir los resultados del procesamiento.
T _g	Tiempo de actividad de los nodos del grid.
K _{mg}	Cantidad de información manejada en el grid (carga de información y generación de resultados).
V _p	Velocidad de procesamiento de la información dentro del grid

Cantidad de viajes para enviar la información = $\frac{\text{Cantidad de información a transportar}}{\text{Velocidad de la red empleada}}$

Reemplazando los términos con los de la tabla inicial se tiene:

$$NV = \frac{K}{Mbp}$$

Ecuación 1

En el proceso de información que debe procesarse en un grid, usualmente se envía dicha información y se reciben los resultados computados utilizando el mismo canal. Como es de esperarse, en algunas ocasiones, la cantidad de información que se envía es poca pero los resultados son muchos; el caso extremo se provoca cuando se envía mucha información, pero el resultado recibido es pequeño.

Para acotar el diseño del modelo, se asumen las siguientes restricciones:

- El medio de transporte de red se emplea en su capacidad máxima, tanto cuando se envía la información, como cuando se reciben los resultados.
- Siempre existe información que debe ser transportada de un punto a otro.

Según estas restricciones, se obtiene que se envía D_1 cantidad de información para que sea procesada y se recibe D_2 cantidad de resultados. Pero según nuestras fórmulas, se tiene que $D_1 = D_2$, por tanto nuestra ecuación original se modifica a:

$$NVp = \frac{Kp}{2Mbp}$$

Ecuación 2

Una vez establecida la ecuación anterior, es necesario introducir el factor tiempo, ya que nuestro objetivo primordial es medir la eficiencia del procesamiento de información. Los tiempos necesarios de un periodo de actividad del grid son los siguientes:

- Tiempo para enviar la información a procesar desde el punto de origen hacia el grid.
- Tiempo de cargar la información dentro de los nodos del grid y tiempo que se obtenga y consoliden los resultados del procesamiento de cada uno de los nodos.
- Tiempo que se tarda para cargar la información en el punto origen dentro del medio de transporte de red.
- Tiempo para que viajen los paquetes con los resultados desde el grid hacia el punto de origen.

Con esto nuestra ecuación de tiempo se establece como:

$$Tv = Tpg + Tng$$

Ecuación 3

T_v : Tiempo que está ocupada la red en un viaje completo de información.

T_{pg} : Tiempo empleado por los paquetes de información en su operación dentro del grid.

T_{ng} : Tiempo empleado por la información desde la salida de la misma hasta recibir los resultados del procesamiento.

El tiempo disponible durante un periodo de actividad que está disponible en el grid, se ha denominado T_g . De esta forma, si se divide el tiempo disponible para enviar información T_g por el tiempo ocupado durante el envío de un viaje de información T_v , se obtienen la cantidad de viajes de información N_{vp} que se pueden realizar durante un periodo de actividad.

$$N_{vp} = \frac{T_g}{T_v}$$

Ecuación 4

Después de establecida esta ecuación podemos reemplazar T_g por la Ecuación 3, quedando nuestra fórmula como:

$$N_{vp} = \frac{T_g}{T_{pg} + T_{ng}}$$

Ecuación 5

Si se iguala la Ecuación 2 con la Ecuación 5, se obtiene que:

$$\frac{K_a}{2Mbp} = \frac{T_g}{T_{pg} + T_{ng}}$$

Dando como resultado, la siguiente ecuación que relaciona la cantidad de información transportable Kp durante el tiempo de empleado por los paquetes de información en su operación dentro del grid Tpv .

$$Ka = \frac{Tg * 2Mbp}{Tpg + Tng}$$

Ecuación 6

Ahora, se considera la cantidad de información manejada en el grid. La cual sería la información que debe cargarse en el grid para su manipulación y la cantidad de información que ha de enviarse como resultado del procesamiento. Por tanto, tenemos que la cantidad de información manejada en el grid Kmg será el doble de la información transportada (carga y generación de resultados), que se expresa como:

$$Kmg = 2Kp$$

Ecuación 7

Se debe considerar, además, la velocidad con la cual se procesa la información dentro del grid, teniendo que:

$$Vp = \frac{Kmg}{Tpg}$$

Ecuación 8

Podemos, precisar que el tiempo empleado en el grid Tpv se puede expresar como:

$$Tpg = \frac{Kmg}{Vp}$$

Ecuación 9

Ahora, podemos reemplazar Tpg en la Ecuación 6, con lo cual obtenemos:

$$Kp = \frac{2Mbp * Tg}{\frac{Kmg}{Vp} + Tng}$$

Ecuación 10

Finalmente, se debe incluir la calidad del servicio dentro del planteamiento de evaluación de la eficiencia, quedando la fórmula de la siguiente forma:

$$Kp = \left(\frac{2Mbp * Tg}{\frac{Kmg}{Vp} + Tng} \right) * QoS$$

Ecuación 11

Esta última ecuación permite conocer la cantidad de información transportada y procesada durante un periodo de actividad determinada, con base en las variaciones que se den en la velocidad de manipulación de información.

Por tanto, el modelo de medición de la eficiencia en infraestructura de los algoritmos de calendarización está expresado en la Ecuación 11 y es el modelo propuesto en este trabajo. En un futuro desarrollo, será necesario probar -por medio de simulación- la aplicabilidad de este modelo.

a) Determinación del plazo de ejecución y del presupuesto

Cuando un consumidor interactúa en el mercado económico de la informática distribuida, debe tomar en cuenta dos variables principales: el plazo de ejecución (durante cuánto tiempo dentro debe estar procesada toda la información) y el presupuesto (cantidad de dinero que está dispuesto a invertir para que un productor procese su información).

Estos factores deben ser utilizados por el agente intermediario cuando se busca un proveedor de recursos. A continuación, se presenta el desarrollo matemático que sirve como base para que el agente intermediario determine cual opción de mercado es la mejor según las necesidades del consumidor. Dichos factores, el plazo de ejecución (PLE) y factor de presupuesto (PRP), se ponderan con valores que están en el rango de cero a uno.

Un plazo de ejecución (PLE) cercano a uno significa la buena voluntad del usuario de fijar un plazo altamente relajado. Esto significa que es posible procesar la información, incluso cuando solamente los recursos más lentos están disponibles.

Un factor de presupuesto (PRP) cercano a uno significa que el usuario está dispuesto a invertir tanto dinero como sea requerido, incluso cuando solamente se utilicen los recursos más costoso.

El agente intermediario utiliza estos factores en la determinación del plazo absoluto (Ecuación 12) y de los valores del presupuesto (Ecuación 13) para un panorama dado de la ejecución.

$$\text{LímiteTiempoEjecución} = T_{\min} + PLE * (T_{\max} - T_{\min})$$

Ecuación 12

Donde:

- T_{\min} es el tiempo requerido para procesar todos los trabajos, en el paralelo, dando al recurso más rápido la máxima prioridad.
- T_{\max} es el tiempo requerido para procesar todos los trabajos, en serie, usando el recurso más lento.
- Un factor de PLE menor que cero implica que los requerimientos del usuario nunca serían procesados.
- Un factor de PLE mayor o igual a cero implica que los requerimientos del usuario siempre serán completados si existen recursos disponibles durante el plazo de ejecución.

$$Presupuesto = C_{\min} + PRP * (C_{\max} - C_{\min})$$

Ecuación 13

Donde:

- C_{\min} es el costo requerido para procesar todos los trabajos, en el paralelo, dando al recurso más barato la máxima prioridad.
- C_{\max} es el tiempo requerido para procesar todos los trabajos, en serie, usando el recurso más barato.
- Un factor de PRP menor que cero implica que los requerimientos del usuario nunca serían procesados.
- Un factor de PRP mayor o igual a uno implica que los requerimientos del usuario siempre serán completados si existen recursos de presupuesto disponibles durante el plazo de ejecución.

b) Objetivo de los agentes intermediarios.

En un modelo económico aplicado a la informática distribuida, la economía contiene un conjunto determinado de recursos $r_1, r_2, r_3, \dots, r_m$. Uno o varios de estos recursos tienen que ser asignados a un consumidor, que puede expresarse como:

$$RA = \{c_1, c_2, c_3, \dots, c_i\}$$

Ecuación 14

Donde:

RA = recursos asignados a un consumidor

C_i = cantidad del recurso i asignado al consumidor

El objetivo del agente intermediario es lograr un RA, tal que se obtenga los mejores beneficios para su cliente o consumidor. Esta preferencia puede ser representada por una función de utilidad U . Si U representa una función de utilidad y se poseen dos conjuntos de recursos RA_i y RA_j un agente prefiere RA_i si $U(RA_i) > U(RA_j)$, en cambio podrá elegir cualquiera de las dos opciones si $U(RA_i) \approx U(RA_j)$, las preferencias son transitivas por tanto si $U(RA_i) \geq U(RA_j)$ y $U(RA_j) \geq U(RA_z)$ entonces $U(RA_i) \geq U(RA_z)$.

Si se agrega a este modelo las restricciones de presupuesto del consumidor RPC, el agente intermediario cuenta con un conjunto de recursos monetarios que pueden ser utilizados para pagar el precio P de los recursos informáticos que proveen los productores. Esta relación se puede expresar como:

$$C(\vec{p}) = \left\{ RA_i : p * RA_i \leq RPC \right\}$$

Ecuación 15

En este caso $C(\vec{p})$ es el conjunto de precio de recursos que deben cumplir con las restricciones del presupuesto del consumidor RPC. Con esto, se puede expresar la demanda de un consumidor de la siguiente forma:

$$D(\vec{p}) = \left\{ RA_i : RA_i \in C(\vec{p}), U(RA_i) \geq RA_i', \forall RA_i' \in C(\vec{p}) \right\}$$

Ecuación 16

Otro parámetro importante en este modelo es el parámetro Kp (cantidad de información transportable en un periodo de actividad y en ambos sentidos) y Mbp (cantidad de información que puede manejar la red de transmisión de datos). Son parámetros de entrada al modelo y para poder calcularlos se debe tomar en cuenta la teoría de la información, la cual tiene como objetivo hacer más eficiente la transmisión de datos desde un punto de inicio hasta un punto de destino, utilizando como medio de transporte una red de comunicaciones. Entre los aspectos que se estudian se encuentran los siguientes:

- Forma de medición de la información.
- Capacidad del canal de comunicaciones durante la transmisión de información.
- Forma en que debe codificarse la información transmitida en el canal para que éste pueda ser utilizado con una tasa mínima de errores.

Otro aspecto relacionado con la transmisión informativa es la medida de información, la cual establece la posibilidad de que un emisor pueda elegir entre varios mensajes y cada mensaje contará con una probabilidad de ser transmitido y la información que contiene para el receptor será menor.

En el caso de la transmisión de datos digital, la información se codifica con el formato binario (ceros y unos). Si estos símbolos son considerados equiprobables, la información que se recibe cuando se detecta un 1 o un 0 viene dado por el Teorema de Shannon, el cual nos indica:

$$I(m) = \log_2 \frac{1}{P(m)}$$

Donde $I(m)$ es la información que se recibe
y $P(m)$ es la probabilidad de recibir un símbolo

Ecuación 17 Teorema de Shannon

Para el caso de los sistemas digitales, se utilizan dos símbolos 0 y 1, los cuales se consideran equiprobables. Entonces la información que se recibe cuando se detecta un 0 o un 1 viene dada por:

$$I(0,1) = \log_2 \frac{1}{P(0.5)}$$

Ecuación 18 Teorema de Shannon para los sistema digitales

Otro aspecto que debe tomarse en cuenta en la transmisión de datos es la entropía. Supongamos que un emisor puede enviar varios símbolos o mensajes: $x_1, x_2, x_3, x_4, \dots, x_n$ y cada uno tiene una probabilidad asociada: $p(x_1), p(x_2), p(x_3), p(x_4), \dots, p(x_n)$. Entonces el valor medio de la información emitida o entropía, viene dada por:

$$E(\text{datos}) = \sum_{k=1}^n I(x_k) * p(x_k)$$

Ecuación 19 Entropía de los datos por transmitir

Si se sustituye la probabilidad se obtiene:

$$E(\text{datos}) = \sum_{k=1}^n \log_2(p(x_k)) * p(x_k)$$

Ecuación 20 Entropía de datos por transmitir con probabilidad

Adicionalmente, la tasa de transmisión trata de medir la velocidad con la cual la fuente transfiere la información, y el valor medio de la información que se transmite es la entropía ($H=E(\text{Datos})$) dividida por la duración media de un símbolo, y de esta forma se obtiene la tasa de información transmitida.

$$T = \frac{H(x)}{t}, \text{ } t = \text{duración promedio de un símbolo transmitido}$$

Ecuación 21 Tasa de transmisión de información

La tasa de transmisión de un símbolo viene dada por la duración promedio de un símbolo, lo cual se mide en segundos. Como en nuestro caso, el emisor envía dígitos binarios equiprobables, entonces la tasa de información de Shannon/Segundo coincide con el número de bits por segundo (bps) que puede identificarse como bit/seg, que en caso del modelo propuesto en esta investigación es el parámetro *Mbps*.

En cuanto a la transferencia de la información, se debe estudiar la relación existente entre la capacidad del canal y la tasa de información. En este caso, la tasa de información establece la cantidad de información producida por el emisor por unidad de tiempo, y la capacidad del canal (medida en bits por segundo) indica la cantidad de datos que un canal puede transmitir por cada unidad de tiempo. Estas dos mediciones dependen del ancho de banda del canal y de la relación señal ruido. Con respecto a este sentido, la teoría de telecomunicaciones establece que si la tasa de información es mayor que la capacidad del canal, no es posible transmitir datos sin errores.

Otro aspecto importante es determinar la velocidad de transmisión informativa, la cual se define con base en la velocidad de modulación, que es la inversa del intervalo de tiempo que tarda el elemento más corto de una señal utilizada para crear un pulso de transmisión y se mide en baudios:

$$V_m = \frac{1}{T(\text{segundo})} (\text{Baudios})$$

Ecuación 22 Velocidad de modulación en la transmisión de datos

Una vez establecido el concepto de velocidad de modulación es posible introducir la definición de la velocidad binaria o velocidad de transmisión, que es la cantidad de dígitos binarios transmitidos por segundo (ya sea que lleven o no información). Por tanto, para un medio de comunicación (enlace) de m canales en paralelo y n niveles de señal se tiene:

$$V_t = \sum_{i=1}^{i=m} \frac{1}{T_i} \log n_i$$

Donde T_i = duración de un pulso para el i -ésimo canal en segundos

n_i = número de niveles usados en i -ésimo canal

Ecuación 23 Medida de la velocidad binaria

Con base en la medida de velocidad binaria, se puede definir la velocidad de transferencia de datos que es el número de bits de datos transmitidos por unidad de tiempo:

$$V_{td} = \frac{d}{t}$$

Donde d = cantidad de bits de datos transmitidos

t = tiempo empleado

Ecuación 24 Velocidad de transferencia de datos

En toda transmisión se debe considerar la velocidad real de transferencia de datos V_{rtd} , en la cual se consideran –únicamente- los bits de información que alcanzan el destino libre de errores de transmisión o la cantidad promedio de bits correctos por unidad de tiempo. Desde este punto, se puede definir la eficiencia de un sistema de comunicación como el cociente entre la velocidad real de transmisión de datos y la velocidad binaria:

$$Eficiencia = \frac{V_{rtd}}{V_t}$$

Ecuación 25 Eficiencia de transmisión de datos

VI. Descripción de la herramienta de simulación de calendarizaciones en la informática distribuida

A. Introducción

Anteriormente se presentó un modelo matemático para evaluar la eficiencia de los algoritmos de calendarización de la informática distribuida. Para probar este modelo es necesario utilizar elementos de simulación, específicamente, una herramienta que permita simular los algoritmos de calendarización de forma que sean comparables. Para tal efecto se utilizará el simulador de informática distribuida, conocido como GridSim, creado por el Dr. Rajkumar Buyya de la Universidad Monash de Melbourne Australia. A continuación se presenta una descripción general de los mecanismos de este simulador, que fue obtenida de la publicación *Economic-Based Distributed Resource Management and Scheduling for Grid Computing* del Dr. Buyya del año 2002.

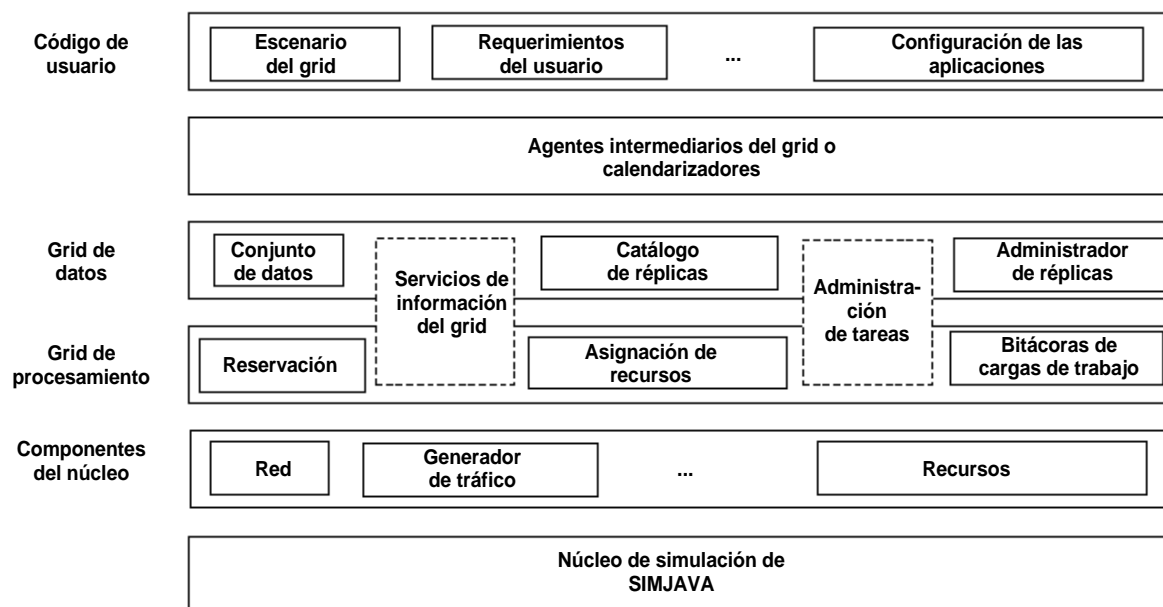
B. Descripción de GridSim

Los utilitarios de GridSim se emplean para generar simulaciones dentro de las cuales se pueden desarrollar y evaluar distintos algoritmos de calendarización o agente intermediario económico de los recursos del grid, los cuales soportan restricciones en cuanto al plazo y al presupuesto (RPP). GridSim permite cuantificar la capacidad de selección dinámica de recursos del agente intermediario en tiempo de ejecución dependiendo de su disponibilidad, capacidad, costo y calidad de servicio requeridos (QoS). El agente intermediario soporta los algoritmos con diversas estrategias de optimización: costo, tiempo, costo-tiempo y tiempo conservador con RPP, entre otros.

La evaluación de funcionamiento de los algoritmos de calendarización basados en la economía es realizada con una serie de simulaciones variando el número de usuarios, del plazo, del presupuesto y de las estrategias de optimización y mediante la simulación de recursos del grid geográficamente distribuidos.

GridSim fue desarrollado con base en SimJava, el cual es un paquete genérico de simulación de eventos discretos y está implementado en Java. Buyya y su equipo desarrollaron GridSim con una arquitectura multicapa⁴², de esta forma el simulador es altamente personalizable y se asemeja más a la arquitectura de las soluciones de informática distribuida. Esta se muestra en la Figura N° 39 y Figura N° 40.

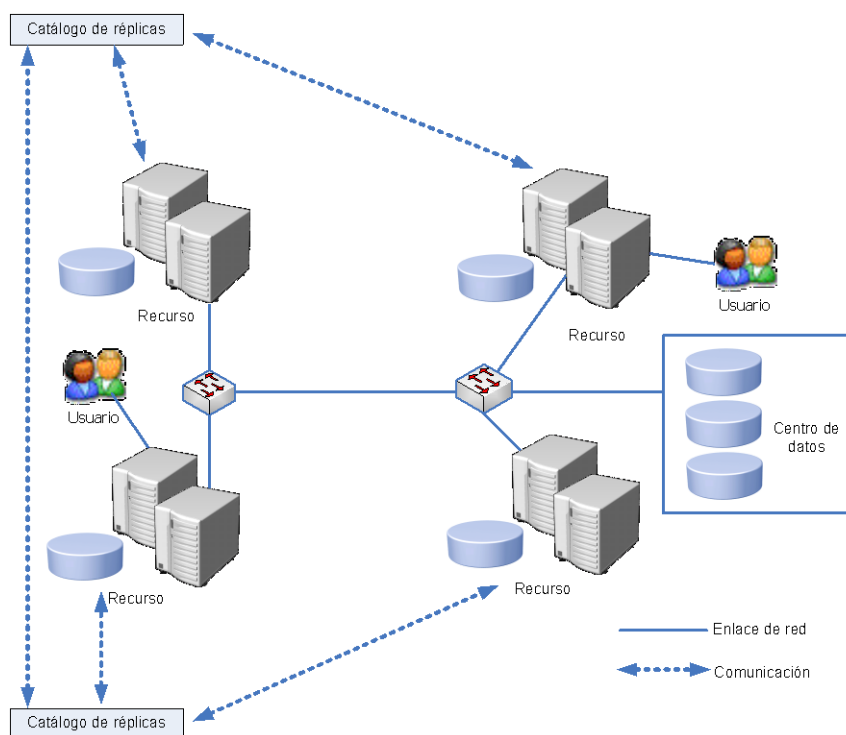
Figura N° 39 Arquitectura general de SimGrid



Fuente: Sulistio 2006.

⁴² En una arquitectura multicapa, los componentes del sistema se agrupan en capas y se establecen mecanismo de comunicación entre estas capas; de esta forma, si un componente dentro de una capa cambia internamente, las demás capas no se ven afectadas.

Figura N° 40 Arquitectura general de un grid de datos



Fuente: Sulistio 2006.

En la Figura N° 40, se muestra un ejemplo de grid de datos. En este caso, un instrumento científico o antena satelital genera gran cantidad de información, la cual es almacenada en el centro de datos. El centro de datos informa a Catálogo de Réplica sobre la existencia de los conjuntos de la nueva información. De esta forma, se permite que los recursos puedan consultar los nuevos datos, ya que está disponible en el catálogo.

C. Simulación económica del agente intermediario

Se utiliza GridSim para simular un ambiente de grid y distintos algoritmos de calendarización restringidos por tiempo de ejecución y por presupuesto. El ambiente simulado del grid contiene múltiples recursos y entidades definidas por el usuario.

Ante todo, el usuario crea un experimento que especifique la aplicación (un sistema de Gridlets⁴³ que representa trabajos de la aplicación con diversos procesos) y la calidad de los requisitos del servicio (con restricciones en cuanto al plazo y al presupuesto con estrategia de optimización).

Se crean entidades que simulan a usuarios y al agente intermediario. Durante la simulación, cada entidad del usuario que posee su propia especificación aplicable y requerimientos de calidad de servicio crea su propia instancia del agente intermediario para programar Gridlets en los recursos.

1. La arquitectura del agente intermediario

La arquitectura de la entidad del agente intermediario junto con su organigrama de la interacción con otras entidades, se muestra en la Figura N° 41.

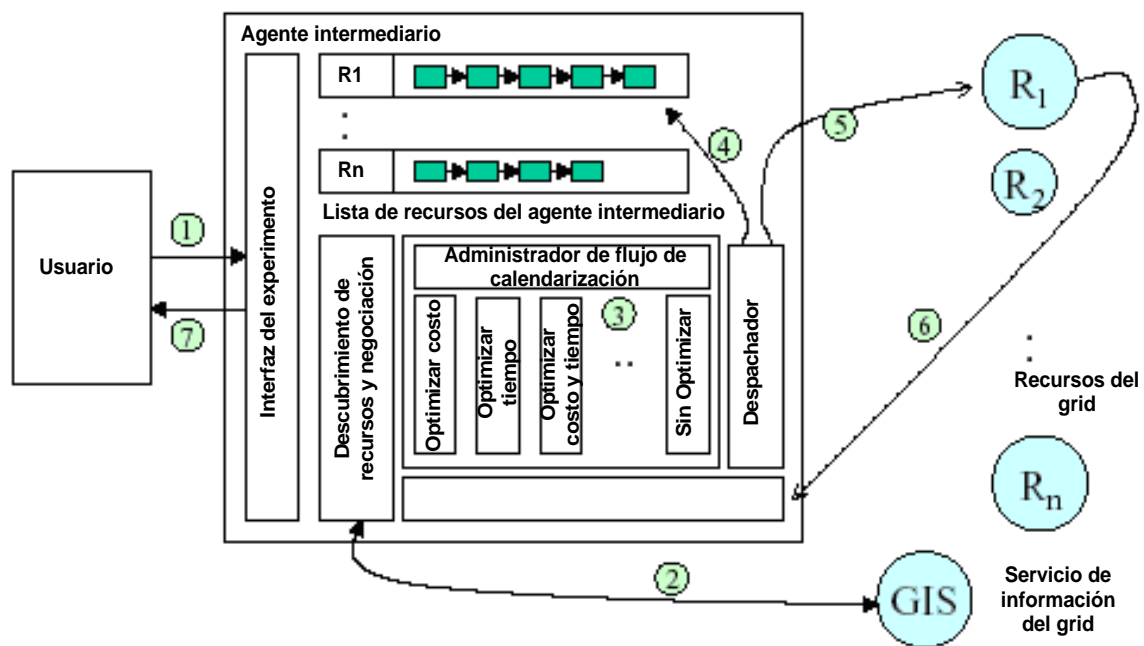
Los componentes principales del agente intermediario son los siguientes: interfaz del experimento, descubrimiento del recurso, encargado del flujo de calendarización (que utiliza heurísticas y algoritmos de calendarización), despachador de Gridlets y receptor de Gridlets. Los siguientes pasos generales describen la funcionalidad de los componentes del agente intermediario y de su interacción:

1. La entidad del usuario crea un experimento que describa la aplicación (una lista de los Gridlets que se procesarán), contenga las exigencias del consumidor y las comunica al agente intermediario por medio de la interfaz del experimento.
2. El módulo de descubrimiento de recursos y de negociación del agente intermediario interactúa recíprocamente con la entidad de GridSim GIS, para identificar la información del contacto de recursos y después interactúa con los recursos para

⁴³ Gridlet es un término propio del modelo definido por el Dr. Buyya y se utiliza para identificar tareas que deben ser realizadas para una actividad determinada.

establecer su configuración y costo del acceso. Crea una lista de recursos que actúe como repositorio para las características de los recursos por mantener, una lista de Gridlets asignada para la ejecución dentro del recurso y los datos del funcionamiento del recurso según lo predicho mediante la metodología de la extrapolación.

Figura N° 41 Componentes de GridSim



Fuente Economic-Based Distributed Resource Management and Scheduling for Grid Computing del Dr. Rajkumar Buyya

3. El encargado del flujo de calendarización selecciona un algoritmo, asigna los Gridlets a los recursos dependiendo de las exigencias del consumidor (los límites del plazo y del presupuesto; y estrategia de optimización (costo, costo-tiempo, tiempo, o variante del tiempo). Los Gridlets, que son asignados a un recurso en específico, se agregan a la lista de estos mantenida por el agente intermediario.

4. Para cada uno de los recursos, el despachador selecciona el número de Gridlets que se puede asignar durante la ejecución según la política de uso con el fin de evitar la sobrecarga de recursos con trabajos de un único usuario.
5. El despachador entonces somete Gridlets a los recursos usando el servicio asincrónico del GridSim.
6. Cuando el proceso de Gridlet termina, el recurso lo devuelve al módulo receptor de Gridlets del agente intermediario, el cual mide y actualiza los parámetros de ejecución. Esto ayuda a predecir la tarifa de consumo del trabajo para tomar decisiones de calendarización.
7. Los pasos 3-6, continúan hasta que se procesan todos los Gridlets o el agente intermediario excede los límites del plazo o del presupuesto. El agente intermediario entonces devuelve los datos actualizados del experimento junto con Gridlets procesados a la entidad del usuario.

El paquete del agente intermediario del grid incluye las siguientes clases:

- Clase Experimento: Actúa como contenedor que se utiliza para representar la configuración del experimento de la simulación, el cual incluye la aplicación sintetizada (un sistema de Gridlets almacenado en GridletList) y las exigencias del consumidor tales como plazo máximo de ejecución, las restricciones del presupuesto y la estrategia de optimización. Proporciona los métodos para poner al día y consultar los parámetros y el estado del experimento. La entidad del usuario invoca a la entidad del agente intermediario y pasa sus requerimientos mediante el objeto del experimento. Cuando se recibe un experimento del usuario, el agente intermediario calendariza los Gridlets conforme las políticas de optimización del experimento.
- Clase EntidadUsuario: Es una entidad de GridSim que simula al usuario. Invoca el agente intermediario y pasa las exigencias del consumidor. Cuando recibe los resultados del proceso de la aplicación, registra los parámetros de interés por medio de

la entidad de `gridsim.Statistics`. Cuando no tiene más requerimientos de procesamiento, envía un mensaje de `END_OF_SIMULATION` al agente intermediario y a las entidades de `gridsim.GridSimShutdown`.

- Clase `AgenteIntermediario`: Entidad de `GridSim` que simula al agente intermediario de los recursos del grid. Cuando recibe el experimento de la entidad del usuario, lleva a cabo el descubrimiento de los recursos y determina los valores del plazo y del presupuesto basados en factores de `D` (tiempo máximo de ejecución) y de `B` (presupuesto) y después procede con la calendarización. Calendariza `Gridlets` en los recursos dependiendo de los requerimientos del usuario, de la estrategia de optimización, del costo de recursos y de su disponibilidad. Cuando recibe los resultados del proceso de la aplicación, registra los parámetros de interés por medio de la entidad `gridsim.Statistics`. Cuando no tiene más requerimientos de procesamiento, envía el mensaje de `END_OF_SIMULATION` a la entidad de `gridsim.GridSimShutdown`.
- Clase `RecursoAgenteIntermediario`: Actúa como contenedor para que el agente intermediario mantenga un expediente detallado de los recursos y de las aplicaciones para procesar los requerimientos del usuario. Mantiene las características del recurso, una lista de `Gridlets` asignados al recurso, la cantidad real de MIPS disponibles para el usuario y un informe sobre los `Gridlets` procesados. Estas medidas ayudan en la extrapolación y para predecir el funcionamiento del recurso desde el punto de vista del usuario; además, en trabajos de calendarización dinámica durante el tiempo de ejecución.
- Clase `GeneradorDeReportes`. Entidad definida por el usuario, utilizada para crear un informe al final de cada simulación e interactúa con la entidad de `gridsim.Statistics`. Si el usuario no desea crear un informe, después puede pasar el parámetro "falta de información" como el nombre de la entidad de `GeneradorDeReportes`. Los usuarios

tienen la posibilidad de elegir cualquier nombre para la entidad de `GeneradorDeReportes` y para el nombre de la clase, puesto que todas las entidades son identificadas por su nombre definido en el tiempo de ejecución.

D. Sistema de administración de recursos y calendarización

Uno de los componentes principales de la arquitectura de informática distribuida es el sistema de administración y calendarización de recursos. Buyya, Abramson y Giddy proponen un sistema calendarizador conocido como Nimrod/G.

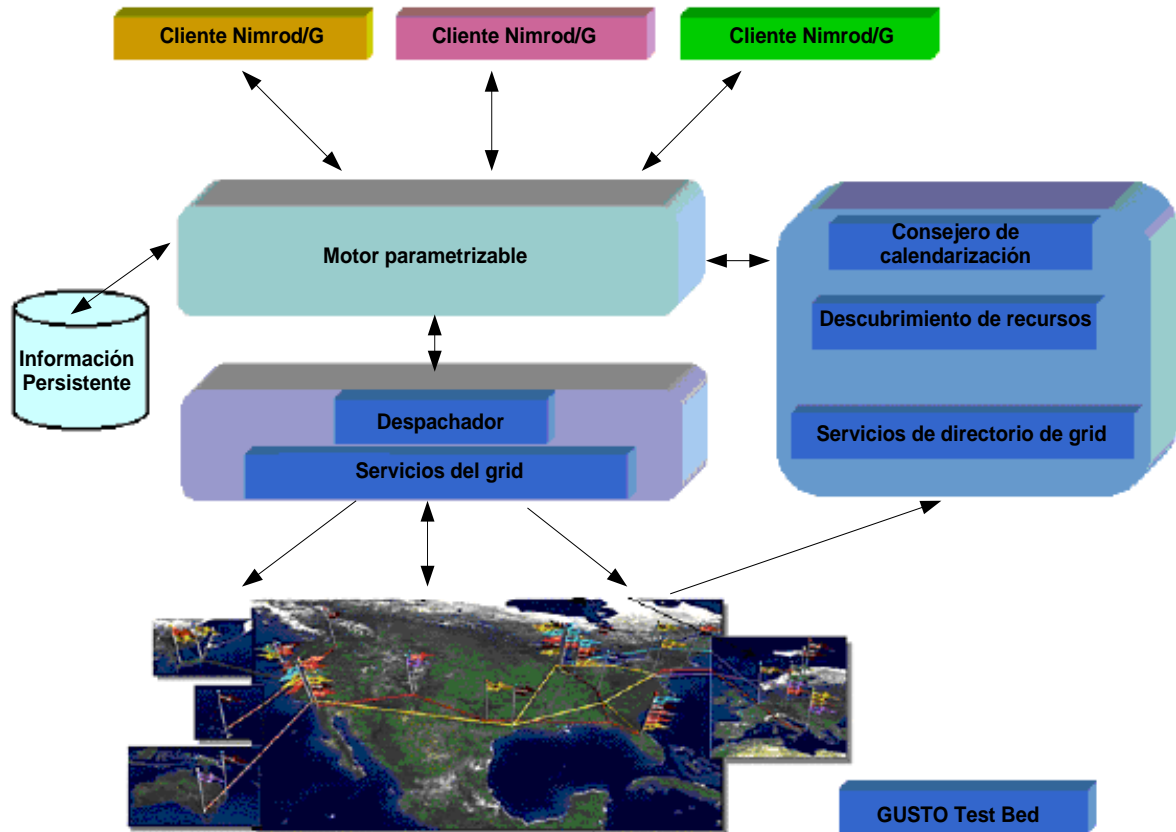
Nimrod/G es un sistema de administración de recursos y de calendarización que utiliza los servicios de Globus Toolkit y puede ser extendido de forma fácil para interoperar con otros paquetes de software de capa intermedia; por ejemplo, NetSolve desarrollado por Casanova, Plank y Dongarra (1997). Nimrod/G se enfoca en la calendarización dinámica de cargas de información que deben ser procesadas utilizando recursos computacionales, los cuales están geográficamente distribuidos y conectados a través de una red.

Una de las características más sobresaliente de Nimrod/G es la posibilidad de utilizar un lenguaje para crear modelos de operación, con el cual se pueden especificar las tareas que deben ser ejecutadas. Los usuarios pueden crear un plan y utilizar el motor de ejecución de Nimrod para enviar, ejecutar y recolectar resultados de varias computadoras. Entre las posibles aplicaciones prácticas de Nimrod se encuentran las de bioinformática, operaciones de investigación y simulación de procesos de negocios. La integración con el Globus Toolkit permite que Nimrod descubra recursos de forma dinámica y utilice computadores de distintas arquitecturas distribuidos geográficamente.

La arquitectura de Nimrod/G se muestra en la Figura N° 42 e incluye los siguientes componentes:

- Estación de trabajo cliente: Se utiliza como interfaz de usuario para controlar y supervisar los experimentos de calendarización. El usuario puede modificar parámetros tales como tiempo y costo, los cuales serán utilizados por el calendarizador en el momento de crear los planes de procesamiento computacional de cargas de información. También posee funciones de monitoreo del progreso de las tareas de procesamiento. Un aspecto flexible de Nimrod es se pueden ejecutar varias consolas de administración, de forma tal que un experimento puede ser iniciado en una estación de trabajo, monitoreado en otra y, de ser necesario, detenido con otra consola de administración.
- Motor parametrizable: Se encarga de la creación de trabajos de procesamiento e interactúa con los clientes y el disparador de tareas. El insumo de este componente es el plan parametrizable creado por el administrador del sistema y permite que el experimento sea persistente. De esta forma, si la ejecución de un experimento se detiene por algún motivo (fallas en la red, computadores o cualquier otro componente), éste puede ser reiniciado en un tiempo posterior.
- Calendarizador de tareas: Es responsable de descubrir recursos, seleccionarlos y asignarles trabajos de procesamiento de cargas de información. Este componente interactúa con el directorio de servicios de Globus y de NetSolve o cualquier otro disponible. El algoritmo de selección de recursos es el encargado de escoger los recursos que cumplen con las restricciones de tiempo máximo de ejecución y que al mismo tiempo minimizan el costo total de la ejecución.
- Disparador de tareas: Inicia la ejecución de las tareas asignadas a un recurso seleccionado por el calendarizador.
- Ejecutor de tareas: Este componente se encarga de transportar las tareas y datos de una aplicación en particular y de recolectar los resultados de dicha ejecución.

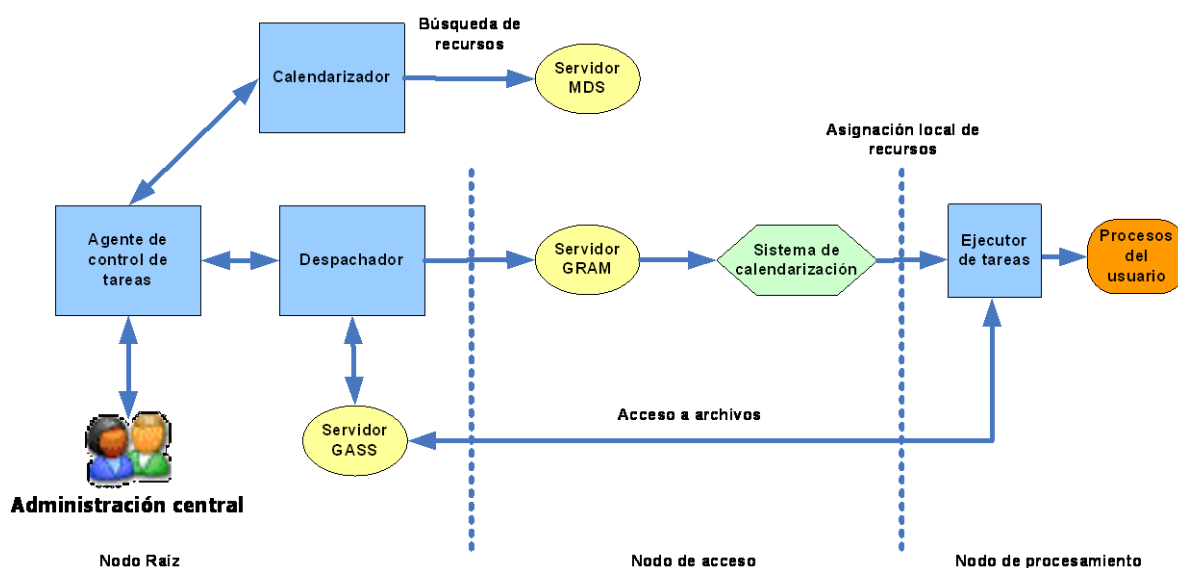
Figura N° 42 Arquitectura de Nimrod/G



Fuente: Buyya Rajkumar, Abramson David y Giddy Jonathan. Nimrod/G: An architecture for a Resource Management and Scheduling System in a Global Computational Grid (2006)

La interacción entre estos componentes se muestra en la Figura N° 43 y se puede apreciar el intercambio de mensajes de control que se da entre las capas de administración, la de acceso a los recursos y la capa de procesamiento de información.

Figura N° 43 Interacción de los componentes de Nimrod/G



Fuente: Buyya, Abramson y Giddy. Nimrod/G: An architecture for a Resource Management and Scheduling System in a Global Computational Grid (2006).

El uso de modelos económicos permite que Nimrod/G actúe en lugar del usuario e intente completar los trabajos asignados respetando las restricciones de tiempo y costo impuestas, siempre tratando de obtener el máximo beneficio para el usuario. Normalmente las restricciones de tiempo son establecidas por el usuario y representa el tiempo que está dispuesto a esperar para que complete el procesamiento de la información. El tiempo está restringido por factores tales como horarios de producción o tiempo límites impuestos a las investigaciones. Otra forma de establecer parámetros al sistema económico es mediante restricciones tales como “esta es la cantidad de dinero que estoy dispuesto a pagar si puede terminar el trabajo de procesamiento en este tiempo”. Con esta restricción, se inicia un proceso de negociación con los productores hasta encontrara uno que cumpla con los requerimientos del usuario.

El sistema de calendarización puede utilizar distintos tipos de parámetros con el fin de establecer una política de calendarización que permita completar la ejecución de una aplicación de manera óptima. Entre estos parámetros se encuentran los siguientes:

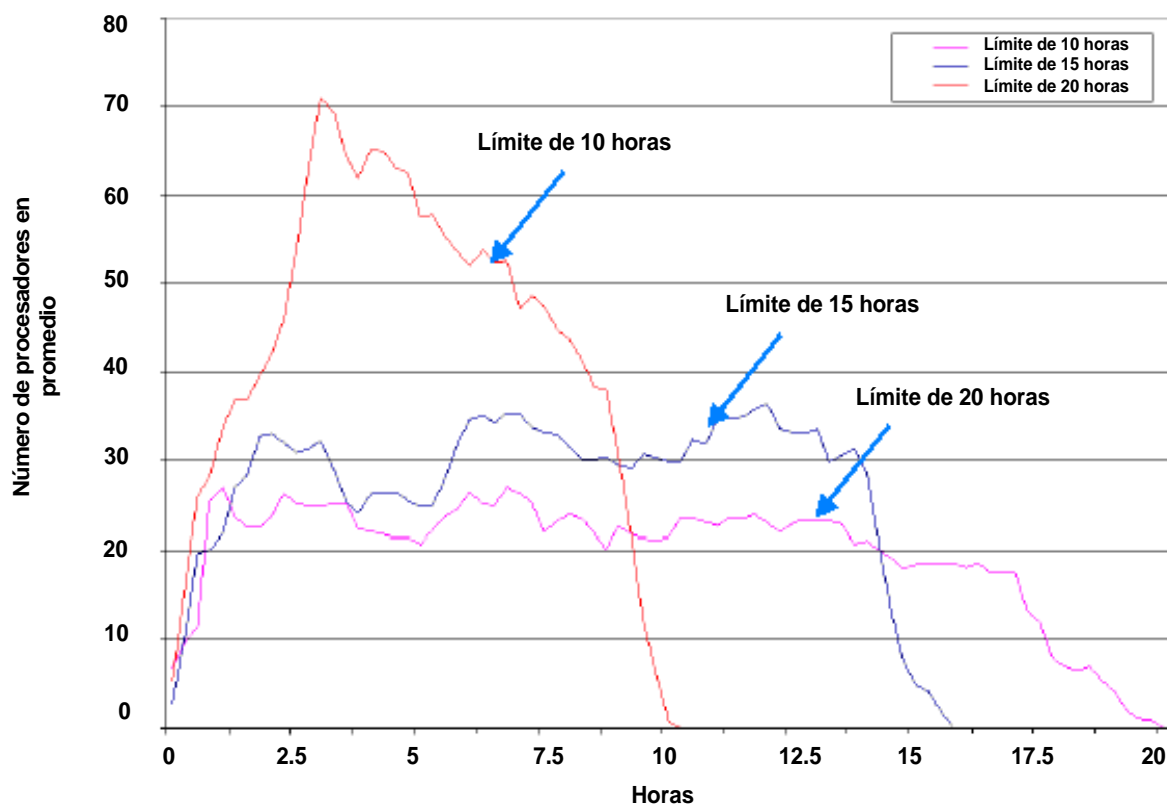
- Capacidad de los recursos: velocidad, cantidad de memoria.
- Estado del recurso: carga actual del procesador, memoria disponible y espacio de almacenamiento libre.
- Arquitectura y configuración del recurso.
- Requerimientos de la aplicación en cuanto a los recursos que necesita.
- Velocidad de acceso, por ejemplo velocidad de los discos duros.
- Cantidad de nodos libres o disponibles.
- Prioridad que tendría el usuario para la utilización de los recursos.
- Ancho de banda de red y latencia de comunicación entre los trabajos de procesamiento.
- Confiabilidad de la conexión a los recursos.
- Preferencias del usuario.
- Información histórica existente sobre el productor.
- Plazo para la finalización de ejecución de las tareas.
- Cantidad de dinero que está dispuesto a pagar el usuario.
- Costo de los recursos del productor.
- Cambio en los costos debido a las economías de escala: cuánto baja el precio de consumo de los recursos por más tiempo o si utilizo mayor cantidad de nodos.

Los parámetros que influyen más en la selección de los productos son el costo de los recursos, la cantidad de dinero que el usuario está dispuesto a pagar y las restricciones en cuanto al tiempo de finalización de las tareas.

En el Gráfico N° 20, se muestra la ejecución de distintos experimentos que utilizan Nimrod/G como mecanismo de calendarización: conforme se aumentan las restricciones

con respecto al tiempo de procesamiento (una restricción implica que los resultados deben obtenerse en mayor o menor cantidad de horas), el calendarizador necesita utilizar más recursos para cumplir con estas restricciones. Cabe resaltar que el uso de más recursos implica que los costos de ejecución total aumentan.

Gráfico N° 20 Ejemplo de ejecución de Nimrod/G.



Fuente: Buyya Rajkumar, Abramson David y Giddy Jonathan. Nimrod/G: An architecture for a Resource Management and Scheduling System in a Global Computational Grid (2006).

Con base en estos resultados, se concluye que cuando las restricciones de tiempo son limitadas, el cliente o usuario del grid tendrá que pagar más para cumplir con el plazo de ejecución establecido. En el Gráfico N° 20, se muestra cómo el costo se eleva cuando el límite de ejecución es de 10 horas, pues se utilizan más procesadores, lo cual aumenta el

costo de total de la ejecución. Si se desea disminuir el costo económico total de la ejecución, será necesario que se flexibilicen los límites de tiempo de ejecución; sin embargo, si el costo se baja demasiado, los tiempos de ejecución podrían ser de 15 o de hasta 20 horas.

VII. Análisis de resultados

A. Introducción

Esta investigación se centrará en el análisis de los resultados obtenidos por medio de análisis estadístico de datos históricos y análisis de simulaciones. Las simulaciones realizadas tienen como objetivo estudiar la eficiencia de distintos algoritmos de distribución de carga y el uso de los recursos de la informática distribuida. Con el resultado de las simulaciones se mide el impacto económico que tiene la aplicación de un algoritmo determinado. De la misma forma, se analiza la base de datos histórica sobre sistemas de informática distribuida seleccionados y utilizando este insumo se estudia cada uno de los indicadores propuestos en esta investigación.

Para efectos generales, se entenderá como el Índice Básico de Eficiencia en Computación Distribuida (IBECOD) aquel que resume la eficiencia de los grid de computadoras en cuanto al procesamiento de cargas de trabajo. Este índice ya fue explicado detalladamente Capítulo de Metodología y se define como:

$$IBECOD = IGE + IEI + IGO + \xi$$

IGE: Índice de Gestión Económica.

IEI: Índice de Eficiencia en Infraestructura.

IGO: Índice de Gestión Operacional.

ξ : Margen de error.

Para cada uno de los componentes de este modelo, se presenta el cálculo de los indicadores y el análisis correspondiente.

B. Índice de Gestión Económica (IGE)

Este índice brinda información sobre las finanzas de las empresas que brindan servicios de grid computing. Con base en esta información, se puede determinar si la aplicación del modelo de operación de la informática distribuida tiene un impacto en las finanzas de las empresas. Este índice comprende los siguientes indicadores:

Cuadro N° 13 Composición del Índice de Gestión Económica (IGE)

Indicador	Unidad de medida
Ingresos en la empresa	Dólares
Evolución de los ingresos de las empresas a través de los años	Porcentaje
Porcentaje de cambio en la eficiencia a través de los años (tareas no atendidas)	Porcentaje
Cambio en los costos debido a la aplicación de distintos modelos de asignación	Grid Units G\$
Porcentaje de cambio en los costos debido a la aplicación de distintos modelos de asignación	Porcentaje

Fuente: Elaboración propia.

La descripción de los indicadores de gestión económica se muestra a continuación:

- **Ingresos de la empresa (INGEMP):** Los ingresos se expresan en dólares y se detallan mensualmente. Estos ingresos son el resultado de la suma de los montos cobrados por cada una de las tareas procesadas durante el mes.
- **Evolución de los ingresos de las empresas a través de los años (EVOING):** Este indicador se expresa en dólares y se calcula comparando la diferencia en monto existen entre el periodo actual y el mismo periodo del año anterior.
- **Porcentaje de cambio en la eficiencia a través de los años (tareas no atendidas) (PORCAMEF):** Da a conocer el porcentaje de tareas que no han sido atendidas con

éxito dentro del grid. Se compara el comportamiento de este indicador a través de los periodos estudiados.

- **Cambio en los costos debido a la aplicación de distintos modelos de asignación (CAMCOSTO):** Se mide en Grid Units y permite comparar el cambio producido en los costos debido a la aplicación de un modelo de administración y asignación de recursos.
- **Porcentaje de cambio en los costos debido a la aplicación de distintos modelos de asignación (PORCAMCOSTO):** Mide el cambio porcentual en los costos debido a la aplicación de distintos modelos de asignación de recursos y atención de tareas.

1. SDSC Blue Horizon (San Diego Supercomputer Center)

El sistema de grid computing SDSC Blue Horizon está compuesto por un total de 144 nodos, en donde cada uno cuenta con 8 procesadores para sumar un total de 1152 disponibles. Este sistema se encarga de ejecutar tareas de procesamiento recibidas con una tarifa estándar de \$1 por cada hora de procesador utilizado. A continuación se analizarán los distintos indicadores económicos para este grid.

a) Ingresos de la empresa (INGEMP)

En el Cuadro N° 14, se observa el detalle de los ingresos percibidos por el sistema de procesamiento SDSC Blue Horizon. Durante el año 2000, se atendieron un total de 49.044,00 tareas de procesamiento de datos con lo cual se logró captar un total de \$1.975.514,85, con un promedio de ingresos de \$49,04 por tarea atendida y procesada. Al siguiente año, el total de tareas atendidas fue de 97.290,00 y los ingresos alcanzaron el monto \$5.527.830,20 con lo cual el promedio por trabajo atendido pasó a ser \$56.82. Esto representa un aumento significativo en los ingresos de la empresa. Durante el año 2002, se

atendieron 96.980,00 tareas de procesamiento con un promedio de ingreso de \$51.40. Da un gran total de \$4.985.322,56 para ese periodo.

Cuadro N° 14 Ingresos de SDSC Blue Horizon

Mes	2000			2001			2002		
	Total Trabajos	Tiempo de CPU Utilizado (segundos)	Ingresos Miles US\$	Total Trabajos	Tiempo de CPU Utilizado (segundos)	Ingresos Miles US\$	Total Trabajos	Tiempo de CPU Utilizado (segundos)	Ingresos Miles US\$
1				6,632.00	15,057,432	\$ 250.96	8,082.00	22,910,299	\$ 381.84
2				7,591.00	28,657,057	\$ 477.62	7,434.00	13,893,808	\$ 231.56
3				6,939.00	30,424,345	\$ 507.07	12,791.00	31,309,332	\$ 521.82
4	12.00	146,973	\$ 2.45	7,881.00	31,295,442	\$ 521.59	11,627.00	29,845,438	\$ 497.42
5	8,630.00	14,160,433	\$ 236.01	9,068.00	37,276,401	\$ 621.27	10,979.00	28,909,851	\$ 481.83
6	7,842.00	17,244,128	\$ 287.40	8,660.00	34,716,283	\$ 578.60	10,313.00	28,482,068	\$ 474.70
7	6,534.00	17,242,312	\$ 287.37	8,069.00	29,078,543	\$ 484.64	6,729.00	28,218,812	\$ 470.31
8	6,628.00	15,558,879	\$ 259.31	8,636.00	31,665,998	\$ 527.77	6,325.00	29,294,966	\$ 488.25
9	5,935.00	16,153,836	\$ 269.23	8,630.00	41,709,350	\$ 695.16	5,231.00	27,874,551	\$ 464.58
10	4,666.00	14,834,054	\$ 247.23	8,488.00	25,382,887	\$ 423.05	5,902.00	28,682,225	\$ 478.04
11	5,004.00	12,152,534	\$ 202.54	8,964.00	26,406,075	\$ 440.10	6,198.00	29,698,003	\$ 494.97
12	3,793.00	11,037,740	\$ 183.96	7,732.00	26,201,917	\$ 436.70	5,369.00	30,318,879	\$ 505.31
Total	49,044.00	107,493,150.70	\$ 1,975.51	97,290.00	357,871,729.71	\$ 5,527.83	96,980.00	329,438,232.66	\$ 4,985.32

Fuente: Elaboración propia con datos de las bitácoras de SDSC Blue Horizon

En el Cuadro N° 15, se observa que durante el año 2001 se percibió mayor cantidad de ingresos, lo cual es consistente con el número de tareas atendidas, puesto que en ese año se atendieron 310 taras más que el año 2002, incluso el ingreso promedio es superior a cualquier otro periodo. Es importante destacar que para ese año el promedio de uso de CPU de las tareas fue de 3.678,40 segundos, el cual es el más alto dentro del rango de años estudiados. Esto justifica el porqué los ingresos durante ese año son superiores.

Cuadro N° 15. Resumen de Ingresos de SDSC Blue Horizon

Año	Total Trabajos	Total Ingresos	Ingreso Promedio	Duración Promedio
2000	49.044,00	\$ 1.975.514,85	\$ 40,28	2.191,77
2001	97.290,00	\$ 5.527.830,20	\$ 56,82	3.678,40
2002	96.980,00	\$ 4.985.322,56	\$ 51,41	3.396,97

Fuente: Elaboración propia con datos de las bitácoras de SDSC Blue Horizon

Con base en los datos anteriores, el escenario ideal para SDSC Blue Horizon se presenta cuando los trabajos atendidos utilizan mayor cantidad de procesador, puesto que los ingresos dependen de esa variable. Sin embargo, para optimizar el uso de la infraestructura se debe aumentar la tasa de atención de tareas concurrentes.

b) Evolución de los ingresos de las empresas a través de los años (EVOING)

En el Cuadro N° 16, se encuentra el detalle de la evolución de los ingresos SDSC Blue Horizon. Durante el año 2001, se presentó un aumento significativo en ellos. En promedio, los ingresos aumentaron un 64%; específicamente, durante los meses de mayo y setiembre, se presenta un cambio superior con respecto a los otros meses. En general, la menor tasa de crecimiento se presenta en julio con un 41% y la mayor es de un 62% en el mes de mayo, cuando se atendieron 9,068 trabajos.

Cabe resaltar que este crecimiento se debe, en general, a un aumento sostenido de la cantidad de trabajos atendidos durante el año 2001. En promedio, los trabajos procesados por el grid aumentaron un 50%. El mes de mayor crecimiento fue diciembre con un 51% y el de menor crecimiento mayo con un 5%. Con base en los datos anteriores, es paradójico que en mayo se presenta la menor tasa de crecimiento en cuanto a cantidad de tareas atendidas, pero al mismo tiempo se aumentaron los ingresos en un 62%. Este fenómeno se explica debido a que durante el mes de mayo el consumo de tiempo de CPU es de 37.276.401 de segundos, tal y como se aprecia en el Cuadro N° 16, el promedio de CPU

por tarea es de 4.110,76 segundos, lo cual resulta el promedio más alto durante ese año. Con estos datos cabe resaltar que cuanto más tiempo de CPU sea utilizado por las tareas, mayores serán los ingresos percibidos por la empresa.

Cuadro N° 16 Evolución de los ingresos de SDSC Blue Horizon

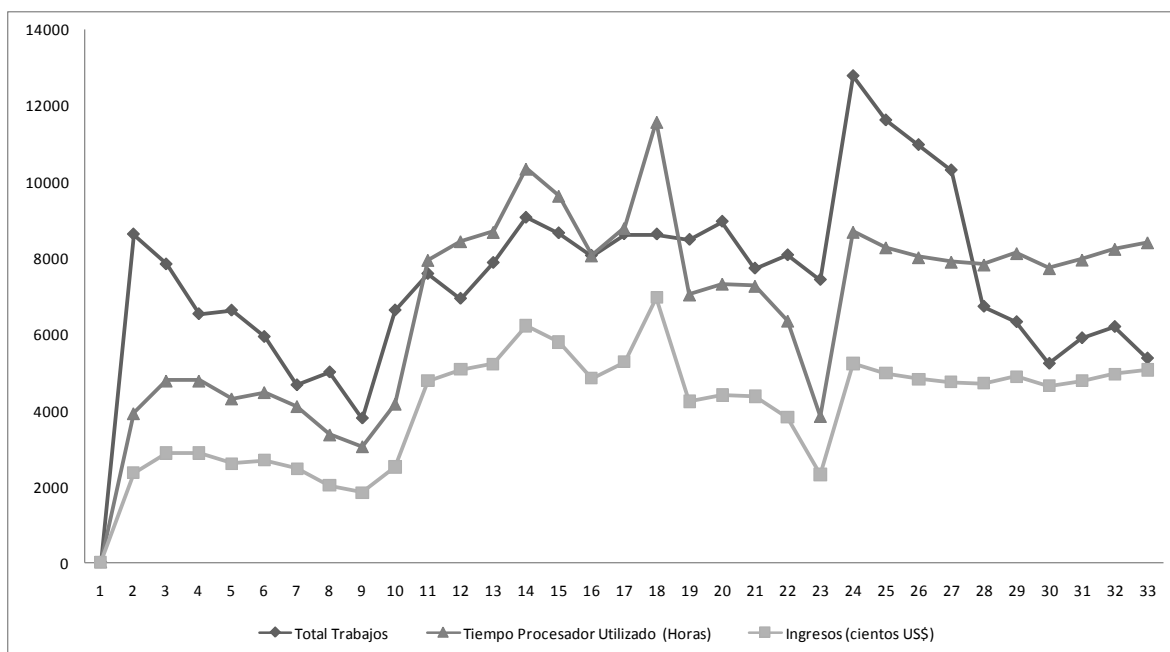
Mes	2000		2001				2002			
	Total Trabajos	Ingresos US\$	Total Trabajos	Ingresos US\$	Cambio US\$	Cambio %	Total Trabajos	Ingresos US\$	Cambio US\$	Cambio %
1			6,632.00	\$ 250,957.20	\$ 250,957.20	100%	8,082.00	\$ 381,838.32	\$ 130,881.12	34%
2			7,591.00	\$ 477,617.61	\$ 477,617.61	100%	7,434.00	\$ 231,563.47	\$ (246,054.14)	-106%
3			6,939.00	\$ 507,072.41	\$ 507,072.41	100%	12,791.00	\$ 521,822.20	\$ 14,749.79	3%
4	12.00	\$ 2,449.56	7,881.00	\$ 521,590.70	\$ 519,141.14	100%	11,627.00	\$ 497,423.97	\$ (24,166.73)	-5%
5	8,630.00	\$ 236,007.21	9,068.00	\$ 621,273.35	\$ 385,266.14	62%	10,979.00	\$ 481,830.86	\$ (139,442.49)	-29%
6	7,842.00	\$ 287,402.14	8,660.00	\$ 578,604.72	\$ 291,202.58	50%	10,313.00	\$ 474,701.13	\$ (103,903.59)	-22%
7	6,534.00	\$ 287,371.87	8,069.00	\$ 484,642.38	\$ 197,270.51	41%	6,729.00	\$ 470,313.53	\$ (14,328.85)	-3%
8	6,628.00	\$ 259,314.65	8,636.00	\$ 527,766.63	\$ 268,451.98	51%	6,325.00	\$ 488,249.43	\$ (39,517.20)	-8%
9	5,935.00	\$ 269,230.60	8,630.00	\$ 695,155.84	\$ 425,925.23	61%	5,231.00	\$ 464,575.85	\$ (230,579.98)	-50%
10	4,666.00	\$ 247,234.24	8,488.00	\$ 423,048.12	\$ 175,813.88	42%	5,902.00	\$ 478,037.08	\$ 54,988.96	12%
11	5,004.00	\$ 202,542.24	8,964.00	\$ 440,101.25	\$ 237,559.01	54%	6,198.00	\$ 494,966.71	\$ 54,865.46	11%
12	3,793.00	\$ 183,962.34	7,732.00	\$ 436,698.62	\$ 252,736.28	58%	5,369.00	\$ 505,314.65	\$ 68,616.02	14%
Total	49,044.00	\$ 1,975,514.85	97,290.00	\$ 5,527,830.20	\$ 3,552,315.35	64%	96,980.00	\$ 4,985,322.56	\$ (542,507.64)	-11%

Fuente: Elaboración propia con datos de las bitácoras de SDSC Blue Horizon

Una situación muy distinta se presenta en el año 2002, en el cual los ingresos disminuyeron un 11% en promedio. El mes de febrero se convierte en el más crítico, pues los ingresos bajaron un 106% con respecto al año anterior, lo cual representa una disminución en ingresos por un total de \$246,054.14. Este fenómeno se presenta debido a que durante este mes el tiempo de CPU utilizado por las tareas es de 13.893.808 segundos, lo cual es sumamente inferior si se compara con el mismo mes del año anterior en el cual se utilizaron 28.657.057 segundos. Cabe resaltar que durante este mes la cantidad de tareas es únicamente un 2% inferior con respecto al año anterior, con esto se refuerza la hipótesis de que poseer una cantidad de tareas atendidas no implica ingresos superiores.

En el Gráfico N° 21, se observa la evolución de los ingresos a través del tiempo. El comportamiento de los ingresos es sostenido; aquellos periodos en los cuales se muestra una disminución en los ingresos se debe a una disminución en la cantidad de tiempo de CPU utilizada por los procesos, aspecto que fue analizado en los párrafos anteriores.

Gráfico N° 21 Comportamiento de los ingresos de SDSC Blue Horizon



Fuente: Elaboración propia con datos de las bitácoras de SDSC Blue Horizon

c) Porcentaje de cambio en la eficiencia a través de los años (tareas no atendidas) (PORCAMEF)

En este índice se analiza la eficiencia en la atención de tareas en el marco de los servicios que presta SDSC Blue Horizon; específicamente, si un cliente envía una tarea para ser atendida en el grid se espera que esta ingrese dentro del sistema y sea ejecutada lo más pronto posible.

En el Cuadro N° 17, se observa el detalle de los trabajos de procesamiento que han sido enviados a SDSC Blue Horizon, pero cuya atención no fue realizada con éxito. La tasa de tareas no atendidas tiene su máximo en el año 2000 con un 25% y su mínimo en el año 2002 con un 17%. Desde un punto de vista de negocio, estos índices son muy altos puesto

que -según Accenture (2007)- los clientes de un servicio de grid computing esperan que la disponibilidad y atención de las tareas sea igual o superior al 99.99%. Un porcentaje alto de tareas no atendidas implica que los clientes pierden confiabilidad en la capacidad de la empresa de grid computing para realizar el procesamiento oportuno de tareas y podría implicar que busquen otros proveedores de servicios, los cuales les brinden un nivel de servicio más alto.

Cuadro N° 17 Porcentaje de tareas no atendidas en SDSC Blue Horizon

Mes	2000		2001		2002	
	Total Trabajos Cancelados	%	Total Trabajos Cancelados	%	Total Trabajos Cancelados	%
1			1.438,00	22%	1.083,00	13%
2			1.858,00	24%	1.486,00	20%
3			1.606,00	23%	2.799,00	22%
4	3,00	25%	1.532,00	19%	2.615,00	22%
5	2.064,00	24%	1.903,00	21%	2.365,00	22%
6	1.746,00	22%	1.689,00	20%	1.533,00	15%
7	1.720,00	26%	2.245,00	28%	740,00	11%
8	2.188,00	33%	1.753,00	20%	647,00	10%
9	1.352,00	23%	1.347,00	16%	741,00	14%
10	921,00	20%	1.459,00	17%	898,00	15%
11	1.339,00	27%	1.193,00	13%	627,00	10%
12	1.009,00	27%	1.133,00	15%	689,00	13%
Total	12.342,00	25%	19.156,00	20%	16.223,00	17%

Fuente: Elaboración propia con datos de las bitácoras de SDSC Blue Horizon

La atención no oportuna de las tareas, además de afectar la imagen de la empresa de grid computing ante sus clientes, también implica dinero no percibido. Esto se debe a que la tarea fue enviada por los clientes al grid, posteriormente esta tarea fue calendarizada y su ejecución se inicia. Sin embargo, debido a que dicha tarea no es completada dentro de los términos especificados por el cliente, se cancela. Por tanto, si la tarea es inhabilitada el cliente no paga ninguna tarifa a la empresa de grid computing. Esta es una situación crítica, ya que la tarea cancelada utilizó tiempo de CPU que no fue cobrado y que podría haber sido utilizado para atender a otros clientes.

En el Cuadro N° 18, se muestra la cantidad de tareas no atendidas exitosamente por SDSC Blue Horizon, así como el tiempo que ellas estuvieron utilizando recursos dentro del grid. Se observa que la cantidad de dinero no percibido debido a no brindar la atención completa de las tareas asciende a sumas muy importantes. Por ejemplo, para el año 2000 ese monto fue de \$653.062,73; sin embargo, para los años 2001 y 2002 esos montos ascienden a \$1.847.123,30 y \$1.849.837,83 correspondientemente.

Cuadro N° 18 Tareas no atendidas en SDSC Blue Horizon

Mes	2000			2001			2002		
	Total Trabajos Cancelados	Tiempo de CPU Utilizado (segundos)	Ingresos US\$ no percibidos	Total Trabajos Cancelados	Tiempo de CPU Utilizado (segundos)	Ingresos US\$ no percibidos	Total Trabajos Cancelados	Tiempo de CPU Utilizado (segundos)	Ingresos US\$ no percibidos
1				1.438,00	4.719.923,21	\$ 78.665,39	1.083,00	4.015.712,48	\$ 66.928,54
2				1.858,00	9.213.271,13	\$ 153.554,52	1.486,00	4.275.864,69	\$ 71.264,41
3				1.606,00	9.381.832,84	\$ 156.363,88	2.799,00	7.717.043,66	\$ 128.617,39
4	3,00	5.582,00	\$ 93,03	1.532,00	9.956.169,17	\$ 165.936,15	2.615,00	11.173.018,59	\$ 186.216,98
5	2.064,00	5.912.203,83	\$ 98.536,73	1.903,00	12.400.769,79	\$ 206.679,50	2.365,00	10.607.455,13	\$ 176.790,92
6	1.746,00	6.282.026,69	\$ 104.700,44	1.689,00	12.718.677,70	\$ 211.977,96	1.533,00	10.685.288,91	\$ 178.088,15
7	1.720,00	6.528.000,31	\$ 108.800,01	2.245,00	12.474.313,38	\$ 207.905,22	740,00	11.224.251,64	\$ 187.070,86
8	2.188,00	6.833.825,19	\$ 113.897,09	1.753,00	11.237.695,25	\$ 187.294,92	647,00	9.378.611,49	\$ 156.310,19
9	1.352,00	4.237.938,93	\$ 70.632,32	1.347,00	10.614.521,29	\$ 176.908,69	741,00	13.375.796,67	\$ 222.929,94
10	921,00	2.788.334,61	\$ 46.472,24	1.459,00	8.401.001,19	\$ 140.016,69	898,00	15.675.448,40	\$ 261.257,47
11	1.339,00	2.258.654,32	\$ 37.644,24	1.193,00	9.709.223,22	\$ 161.820,39	627,00	12.861.778,10	\$ 214.362,97
12	1.009,00	4.337.197,98	\$ 72.286,63	1.133,00	8.150.364,90	\$ 135.839,42	689,00	13.707.567,49	\$ 228.459,46
Total	12.342,00	34.846.565,88	\$ 653.062,73	19.156,00	118.977.763,07	\$ 1.847.123,30	16.223,00	124.697.837,25	\$ 1.849.837,83

Fuente: Elaboración propia con datos de las bitácoras de SDSC Blue Horizon

Con base en esta información, la empresa de grid computing debería revisar sus algoritmos de asignación, priorización y calendarización de recursos, pues no se están atendiendo correctamente las tareas y se está perdiendo dinero. En este caso, la empresa debería establecer un mecanismo capaz de detectar que el grid cuenta con la capacidad necesaria para ejecutar la tarea antes de iniciarla. De esta forma se evitará que se invierta tiempo y recursos en tratar de ejecutar una tarea, cuyo procesamiento excede la capacidad de respuesta de la infraestructura actual del grid.

d) Cambio en los costos debido a la aplicación de distintos modelos de asignación (CAMCOSTO) y Porcentaje de cambio en los costos debido a la aplicación de distintos modelos de asignación (PORCAMCOSTO)

Para realizar el análisis de los indicadores CAMCOSTO y PORCAMCOSTO, se utilizó la herramienta de simulación GridSim descrita en el Capítulo VII. El insumo de las simulaciones son datos reales de las bitácoras de SDSC Blue y se aplicaron dos modelos de administración de recursos: tiempo compartido y espacio compartido.

En el modelo de tiempo compartido, todos los consumidores del mercado comparten de forma equitativa todos los recursos; en cambio, en el modelo de espacio compartido los consumidores que solicitan primero los recursos tienen prioridad sobre los consumidores que lo hacen después de ellos.

Dentro de este experimento, se utilizaron las tareas del Sistema SDSC Blue y se ejecutaron utilizando varios recursos y procesos de ejecución dentro de cada recurso. El código fuente de esta simulación se encuentra en el Anexo II. Los recursos utilizados durante la simulación poseen las siguientes características:

- Velocidad de red: 1 Gigabit por segundo
- Retardos de propagación: 10
- Máxima unidad de transmisión: 100,000 bytes
- Costo de los recursos: 3 G\$⁴⁴ por EP⁴⁵

⁴⁴ G\$ es el acrónimo para Grid Units por Entidad de Procesamiento, la cual es una medida de costo estándar definida dentro del modelo de operación de GridSim

⁴⁵ Los Elementos de Procesamiento es la unidad mínima de procesamiento de información, la cual se utiliza para medir la cantidad de información que puede procesar un nodo de informática distribuida. Esta unidad es definida en Buyya Rajkumar. (2005). *Grid Computing Info Centre*. Recuperado el 11 de junio del 2005, de <http://www.GridComputing.com>.

- Conexiones mediante enrutadores de 10 Gbps por segundo
- Los recursos son computadores Sun Ultra Sparc con sistema operativo Solaris

El Cuadro N° 19 y Cuadro N° 20 muestran el tiempo requerido para enviar y recibir resultados desde el grid de procesamiento. Se observa que, prácticamente, no existe diferencia en el tiempo de transporte de información. Este fenómeno se debe a que los dos modelos utilizan la misma infraestructura de comunicación, por tanto, el tiempo de transporte de información no depende de los algoritmos internos de procesamiento del grid, sino de la capacidad de infraestructura de red.

Cuadro N° 19 SDSC Blue tiempo de transporte modelo tiempo compartido

SDSC Blue - Tiempo Compartido				
Tiempo de transporte de Información				
Usuario	Tiempo de Ida y Retorno (segundos)	Número de Saltos	Ancho Banda (bits)	Tiempo Envío Recepción de Trabajos
Usuario 00	0,07440	3,00000	1.000.000,00	18,863600
Usuario 01	0,07280	3,00000	1.000.000,00	18,920000
Usuario 02	0,07280	3,00000	1.000.000,00	18,912000

Fuente: Elaboración propia con resultados de las simulaciones en GridSim

Cuadro N° 20 SDSC Blue Tiempo de transporte modelo espacio compartido

SDSC Blue - Espacio Compartido				
Tiempo de transporte de Información				
Usuario	Tiempo de Ida y Retorno (segundos)	Número de Saltos	Ancho Banda (bits)	Tiempo Envío Recepción de Trabajos
Usuario 00	0,07280	3,00000	1.000.000,00	18,422000
Usuario 01	0,07280	3,00000	1.000.000,00	18,432000
Usuario 02	0,07280	3,00000	1.000.000,00	18,442000

Fuente: Elaboración propia con resultados de las simulaciones en GridSim

En el Cuadro N° 21 y Cuadro N° 22, se muestran los tiempos de duración en atender las tareas dentro del grid. Los resultados muestran que no existe una diferencia entre los tiempos de atención al utilizar el modelo de tiempo compartido o el modelo de espacio compartido. Cabe recordar que la infraestructura de componentes del grid son los mismos, lo único que cambia es el modelo de asignación de recursos.

Cuadro N° 21 SDSC Blue Tiempo Duración Modelo Tiempo Compartido

SDSC Blue - Tiempo Compartido				
Duración Usuario 00				
Entidad	Tiempo Entrada	Tiempo Salida	Duración	Ancho Banda
Usuario 00	18,78520	18,78920	0,00400	1.000.000,00
Enrutador 01	18,79920	18,79960	0,00040	10.000.000,00
Enrutador 02	18,80960	18,81360	0,00400	1.000.000,00
Recurso 01	18,82360	18,82760	0,00400	1.000.000,00
Enrutador 02	18,83760	18,83800	0,00040	10.000.000,00
Enrutador 01	18,84960	18,85360	0,00400	1.000.000,00
Usuario 00	18,86360	N/A		N/A

Fuente: Elaboración propia con resultados de las simulaciones en GridSim

Cuadro N° 22 SDSC Blue Tiempo de duración modelo espacio compartido

SDSC Blue - Espacio Compartido				
Duración Usuario 00				
Entidad	Tiempo Entrada	Tiempo Salida	Duración	Ancho Banda
Usuario 00	18,34520	18,34920	0,00400	1.000.000,00
Enrutador 01	18,35920	18,35960	0,00040	10.000.000,00
Enrutador 02	18,36960	18,37360	0,00400	1.000.000,00
Recurso 01	18,38360	18,38760	0,00400	1.000.000,00
Enrutador 02	18,39760	18,39800	0,00040	10.000.000,00
Enrutador 01	18,40800	18,41200	0,00400	1.000.000,00
Usuario 00	18,42200	N/A	N/A	N/A

Fuente: Elaboración propia con resultados de las simulaciones en GridSim

En el Cuadro N° 23, se muestra el costo de los trabajos de procesamiento para cada uno de los modelos de asignación (tiempo compartido y espacio compartido) . La unidad de costo es G\$. También se aprecia que el costo de procesamiento es más alto si se utiliza un esquema de asignación de recursos basado en el modelo de tiempo compartido, específicamente el costo de este modelo es 4,31 G\$ más alto, lo cual representan un costo adicional de 2,24%.

Cuadro N° 23 SDSC Blue Comparación de Costos Usuario

SDSC Blue Cluster Costos Usuario 00					
Id Trabajo	Estado	Recurso	Costo Tiempo Compartido	Costo Espacio Compartido	
0	Exitoso	10	37,50	39,00	
1	Exitoso	10	40,33	38,88	
2	Exitoso	10	39,90	38,45	
3	Exitoso	10	39,78	38,33	
4	Exitoso	10	39,35	37,90	
Total			196,86	192,55	

Fuente: Elaboración propia con resultados de las simulaciones en GridSim

De igual manera, se realizaron simulaciones con más tareas de procesamiento. El resultado de estas pruebas se muestra en el Cuadro N° 24 y es consistente con los resultados anteriores: nuevamente el costo del modelo de tiempo compartido es superior, específicamente 4,616.17 G\$, lo cual representa que el costo de aplicación de este modelo es 42,07% superior al modelo de espacio compartido.

Cuadro N° 24 SDSC Blue Comparación de costos tareas personalizadas

SDSC Blue					
Costos Tareas Personalizadas					
Id Trabajo	Estado	Recurso	Costo Tiempo Compartido	Costo Espacio Compartido	
1	Exitoso	5	291,26	183,00	
2	Exitoso	5	303,00	183,00	
3	Exitoso	5	541,50	363,00	
4	Exitoso	5	996,74	813,00	
5	Exitoso	5	558,82	363,00	
6	Exitoso	5	775,46	540,08	
7	Exitoso	5	1.871,23	1.623,00	
8	Exitoso	5	351,07	183,00	
9	Exitoso	5	477,00	243,00	
10	Exitoso	5	1.062,75	813,00	
11	Exitoso	5	807,41	543,00	
12	Exitoso	5	810,37	540,01	
13	Exitoso	5	540,00	273,00	
14	Exitoso	5	810,41	543,00	
15	Exitoso	5	360,00	183,00	
16	Exitoso	5	850,59	543,00	
17	Exitoso	5	360,00	183,00	
18	Exitoso	5	372,17	153,00	
19	Exitoso	5	612,17	273,00	
20	Exitoso	5	2.838,82	2.433,00	
Total			15.590,78	10.974,08	

Fuente: Elaboración propia con resultados de las simulaciones en GridSim

En general, las simulaciones realizadas para el Sistema SDSC Blue muestran que la aplicación de distintos modelos de asignación y calendarización de recursos influyen directamente en el costo asociado a la operación del sistema de grid computing; por tanto, la elección correcta de estos algoritmos es vital para catalogar como óptima la operación del grid.

2. Sistema OSC Linux Cluster (Ohio Supercomputer Center)

El sistema de grid computing OSC Linux cluster está compuesto por un total de 57 nodos, los cuales contienen 178 procesadores. Este sistema se encarga de ejecutar tareas de procesamiento recibidas con una tarifa estándar de \$1 por cada hora de procesador utilizado. A continuación se analizarán los distintos indicadores económicos para este grid.

a) Ingresos de la empresa (INGEMP)

En el Cuadro N° 25, se muestran los ingresos percibidos por el sistema OSC Linux Cluster. Durante el año 2000, se realizaron un total de 19.421,00 trabajos de procesamiento de información; esto generó un total de \$ 6.229.955,28 y da como resultado un ingreso promedio de \$ 320.78 por trabajo atendido. Durante el año 2001, se atendieron 16.676,00 tareas de procesamiento, las cuales generaron ingresos por un total de \$ 14.202.518,96 con un ingreso promedio de \$ 851,67 por trabajo atendido.

En el año 2001, los ingresos fueron superiores a los del año anterior a pesar de que se atendieron menos tareas (2.745,00 menos que el año anterior). Es importante destacar que obtener mejores ingresos con menor atención de tareas va de la mano con la calidad de las tareas atendidas; ya que, según los datos presentados no es suficiente atender muchas tareas sino aquellas que necesiten mayor capacidad de procesamiento y, por tanto, generaran más ingresos para la empresa. Esta característica se ve reflejada en el tiempo de procesador utilizado por las tareas en el año 2000: las 19.421,00 tareas atendidas utilizaron 91.123,98 horas de procesamiento, mientras que en el año 2001 un total de 16.676,00 tareas necesitaron 236.708,65 horas de procesamiento.

Con base en los datos anteriores, en un escenario ideal, las tareas que se atiendan en el grid deben ser complejas -de forma tal que se maximicen los ingresos para la empresa que brinda los servicios de procesamiento de tareas.

Cuadro N° 25 Ingresos de OSC Linux Cluster

Mes	2000			2001			2002					
	Total Trabajos	Tiempo de ejecución (segundos)	Tiempo de CPU Utilizado (segundos)	Ingresos Miles US\$	Total Trabajos	Tiempo de ejecución (segundos)	Tiempo de CPU Utilizado (segundos)	Ingresos Miles US\$	Total Trabajos	Tiempo de ejecución (segundos)	Tiempo de CPU Utilizado (segundos)	Ingresos Miles US\$
1	1,086.00	2,325,126.00	1,480,454.38	\$ 24.67	984.00	48,706,893.00	58,950,356.79	\$ 982.51	1,348.00	96,469,301.58	99,736,935.28	\$ 1,662.28
2	1,644.00	17,102,909.00	17,088,282.86	\$ 284.80	1,451.00	62,965,306.00	75,332,573.84	\$ 1,255.54	1,432.00	110,986,979.80	115,732,817.33	\$ 1,928.88
3	813.00	19,753,627.00	47,431,440.40	\$ 790.52	2,307.00	97,284,073.00	109,084,703.73	\$ 1,818.08	2,161.00	129,471,791.59	147,781,482.90	\$ 2,463.02
4	2,179.00	15,906,414.00	23,886,191.94	\$ 398.10	1,354.00	76,087,285.00	118,378,721.09	\$ 1,972.98	1,314.00	116,950,418.66	140,657,067.80	\$ 2,344.28
5	2,074.00	21,546,569.00	43,851,843.68	\$ 730.86	991.00	52,701,400.00	53,105,670.47	\$ 885.09	884.00	108,078,826.71	118,004,767.31	\$ 1,966.75
6	2,249.00	11,207,026.00	20,922,193.11	\$ 348.70	990.00	78,338,227.00	73,159,076.52	\$ 1,219.32	829.00	115,729,474.18	116,568,264.62	\$ 1,942.80
7	2,468.00	24,149,309.00	50,704,781.84	\$ 845.08	1,275.00	70,315,007.00	74,272,961.96	\$ 1,237.88	1,341.00	118,191,852.31	132,018,367.06	\$ 2,200.31
8	1,374.00	26,105,150.00	33,284,545.65	\$ 554.74	2,403.00	80,054,973.00	98,200,396.56	\$ 1,636.67	2,245.00	124,043,575.67	135,274,106.91	\$ 2,254.57
9	1,588.00	18,948,495.00	29,831,415.28	\$ 497.19	1,958.00	96,413,611.00	104,051,590.99	\$ 1,734.19	1,772.00	128,649,504.40	136,475,588.34	\$ 2,274.59
10	1,564.00	18,560,663.00	25,658,925.16	\$ 427.65	2,144.00	58,163,039.00	77,794,571.75	\$ 1,296.58	1,688.00	109,336,537.23	121,263,631.02	\$ 2,021.06
11	1,340.00	22,521,775.00	33,906,247.62	\$ 565.10	819.00	11,485,454.00	9,820,513.82	\$ 163.68	866.00	87,986,045.81	91,403,453.46	\$ 1,523.39
12	1,042.00	33,693,273.00	45,750,994.85	\$ 762.52					1,373.00	128,895,432.99	139,091,022.72	\$ 2,318.18
Total	19,421.00	198,127,063.00	328,046,321.92	\$ 6,229.96	16,676.00	732,515,268.00	852,151,137.52	\$ 14,202.52	17,253.00	1,245,894,307.94	1,494,007,504.76	\$ 24,900.13

Fuente: Elaboración propia con datos de las bitácoras de OSC Linux Cluster

b) Evolución de los ingresos de las empresas a través de los años (EVOING)

La evolución de los ingresos de OSC Linux Cluster es positiva. En el Cuadro N° 26, se aprecia como entre los meses de enero a octubre se presenta un aumento en los ingresos significativamente, con un crecimiento de 17% en el menor de los casos y de hasta un 97% en el mes de mayor actividad. El crecimiento se debe más que todo a la calidad de los trabajos atendidos y el uso intensivo que éstos hacen de los procesadores del grid, punto que fue discutido en el Indicador de Ingresos de la empresa (INGEMP). El único mes en el cual no se presenta crecimiento es noviembre: los ingresos disminuyeron un 245%, la cantidad de trabajos atendidos pasó de 1,340 en el año 2000 a 819 en el año 2001. Esto significa una disminución del 39%. A simple vista, una disminución como esta no debería implicar una reducción tan drástica en los ingresos; sin embargo -para analizar este fenómeno- es necesario tomar en cuenta el tiempo de procesador utilizado por las tareas atendidas. En el año 2000, el tiempo promedio de uso de procesador fue de 25,303.17 y durante el año 2001 fue de 11,990.86, lo cual da como resultado una disminución del 53%. Por tanto, una disminución en la cantidad de tareas atendidas y del tiempo de procesador utilizado por estas influye, en gran medida, en los ingresos del grid.

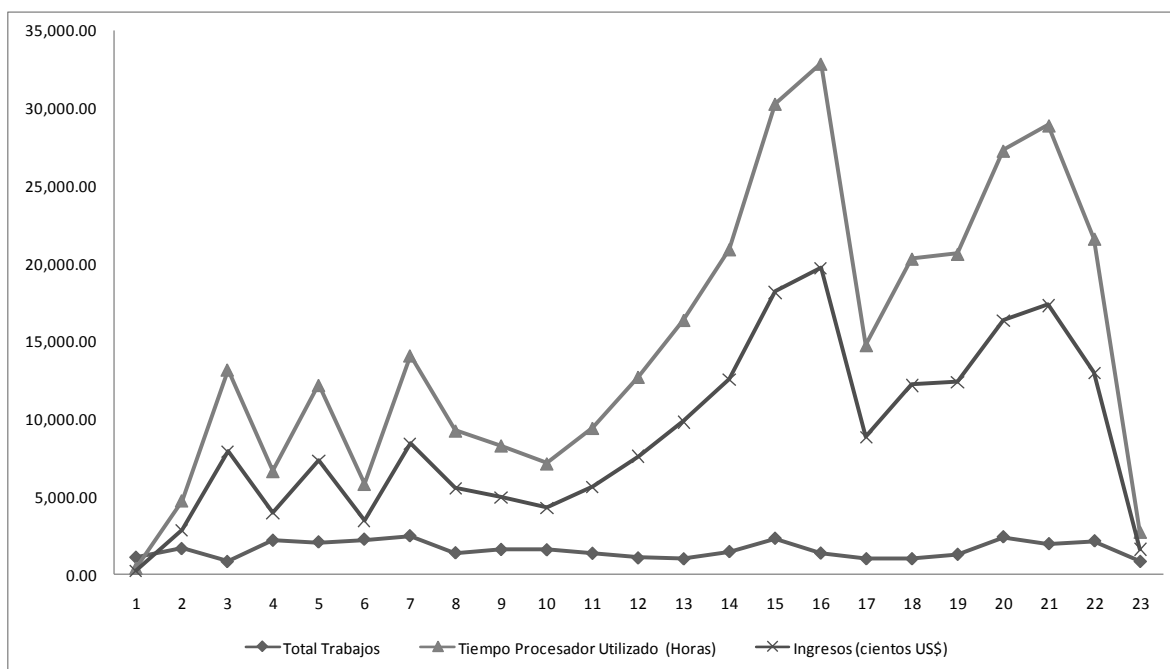
Cuadro N° 26 Evolución de los ingresos de OSC Linux Cluster

Mes	2000		2001		Cambio US\$	Cambio %	2002		Cambio US\$	Cambio %
	Total Trabajos	Ingresos US\$	Total Trabajos	Ingresos US\$			Total Trabajos	Ingresos US\$		
1	1,086.00	\$ 24,674.24	984.00	\$ 982,505.95	\$ 957,831.71	97%	1,348.00	\$ 1,662,282.25	\$ 679,776.31	41%
2	1,644.00	\$ 284,804.71	1,451.00	\$ 1,255,542.90	\$ 970,738.18	77%	1,432.00	\$ 1,928,880.29	\$ 673,337.39	35%
3	813.00	\$ 790,524.01	2,307.00	\$ 1,818,078.40	\$ 1,027,554.39	57%	2,161.00	\$ 2,463,024.72	\$ 644,946.32	26%
4	2,179.00	\$ 398,103.20	1,354.00	\$ 1,972,978.68	\$ 1,574,875.49	80%	1,314.00	\$ 2,344,284.46	\$ 371,305.78	16%
5	2,074.00	\$ 730,864.06	991.00	\$ 885,094.51	\$ 154,230.45	17%	884.00	\$ 1,966,746.12	\$ 1,081,651.61	55%
6	2,249.00	\$ 348,703.22	990.00	\$ 1,219,317.94	\$ 870,614.72	71%	829.00	\$ 1,942,804.41	\$ 723,486.47	37%
7	2,468.00	\$ 845,079.70	1,275.00	\$ 1,237,882.70	\$ 392,803.00	32%	1,341.00	\$ 2,200,306.12	\$ 962,423.42	44%
8	1,374.00	\$ 554,742.43	2,403.00	\$ 1,636,673.28	\$ 1,081,930.85	66%	2,245.00	\$ 2,254,568.45	\$ 617,895.17	27%
9	1,588.00	\$ 497,190.25	1,958.00	\$ 1,734,193.18	\$ 1,237,002.93	71%	1,772.00	\$ 2,274,593.14	\$ 540,399.96	24%
10	1,564.00	\$ 427,648.75	2,144.00	\$ 1,296,576.20	\$ 868,927.44	67%	1,688.00	\$ 2,021,060.52	\$ 724,484.32	36%
11	1,340.00	\$ 565,104.13	819.00	\$ 163,675.23	\$ (401,428.90)	-245%	866.00	\$ 1,523,390.89	\$ 1,359,715.66	89%
12	1,042.00	\$ 762,516.58					1,373.00	\$ 2,318,183.71	\$ 2,318,183.71	100%
Total	19,421.00	\$ 6,229,955.28	16,676.00	\$ 14,202,518.96	\$ 8,735,080.26	62%	17,253.00	\$ 24,900,125.08	\$ 10,697,606.12	43%

Fuente: Elaboración propia con datos de las bitácoras de OSC Linux Cluster

En el Gráfico N° 22, se observa la evolución de los ingresos a través del tiempo. Se aprecia que éstos presentan un crecimiento sostenido, excepto durante el último periodo debido a la disminución del tiempo de procesador utilizado por las tareas atendidas, situación que fue abordada anteriormente. Es importante destacar la estrecha relación que existe entre el tiempo de procesador utilizado y los ingresos percibidos, lo cual se debe a que las tarifas establecidas por el grid se basan en el tiempo de procesador utilizado.

Gráfico N° 22 Comportamiento de los ingresos de OSC Linux Cluster



Fuente: Elaboración propia con datos de las bitácoras de OSC Linux Cluster

c) Porcentaje de cambio en la eficiencia a través de los años (tareas no atendidas) (PORCAMEF)

La atención de tareas es un aspecto primordial entre los servicios que prestan las empresas de grid computing. Si un cliente envía una tarea para que sea atendida en el grid, se espera que esta ingrese dentro del sistema y sea ejecutada lo más pronto posible. Si el grid carece de recursos o la administración de los mismos es ineficiente, no se podrá atender adecuadamente las tareas recibidas. En un ambiente de grid computing, si un cliente -después de enviar sus tareas al grid- determina que no ha sido ejecutado o que está tomando demasiado tiempo de procesamiento, puede cancelar el trabajo. Por tanto, la compañía dueña del grid no recibirá los ingresos provenientes de estas tareas.

En el Cuadro N° 27, se muestra la atención de tareas para el sistema OSC Linux Cluster. Se puede observar que todas las tareas recibidas son procesadas exitosamente y ninguna de

estas ha sido cancelada por los clientes. Por tanto, OSC Linux Cluster cuenta con un alta satisfacción de los clientes y no pierde dinero, producto de tareas canceladas.

Cuadro N° 27 Tareas atendidas en OSC Linux Cluster

Mes	2000		2001		2002		Eficiencia de atención
	Total Trabajos Recibidos	Total Trabajos Atendidos	Total Trabajos Recibidos	Total Trabajos Atendidos	Total Trabajos Recibidos	Total Trabajos Atendidos	
1	1,086.00	1,086.00	984.00	984.00	1,348.00	1,348.00	100%
2	1,644.00	1,644.00	1,451.00	1,451.00	1,432.00	1,432.00	100%
3	813.00	813.00	2,307.00	2,307.00	2,161.00	2,161.00	100%
4	2,179.00	2,179.00	1,354.00	1,354.00	1,314.00	1,314.00	100%
5	2,074.00	2,074.00	991.00	991.00	884.00	884.00	100%
6	2,249.00	2,249.00	990.00	990.00	829.00	829.00	100%
7	2,468.00	2,468.00	1,275.00	1,275.00	1,341.00	1,341.00	100%
8	1,374.00	1,374.00	2,403.00	2,403.00	2,245.00	2,245.00	100%
9	1,588.00	1,588.00	1,958.00	1,958.00	1,772.00	1,772.00	100%
10	1,564.00	1,564.00	2,144.00	2,144.00	1,688.00	1,688.00	100%
11	1,340.00	1,340.00	819.00	819.00	866.00	866.00	100%
12	1,042.00	1,042.00			1,373.00	1,373.00	100%
Total	19,421.00	19,421.00	16,676.00	16,676.00	17,253.00	17,253.00	

Fuente: Elaboración propia con datos de las bitácoras de OSC Linux Cluster

d) Cambio en los costos debido a la aplicación de distintos modelos de asignación (CAMCOSTO) y Porcentaje de cambio en los costos debido a la aplicación de distintos modelos de asignación (PORCAMCOSTO)

En del análisis de los indicadores CAMCOSTO y PORCAMCOSTO, se utilizó la herramienta de simulación GridSim descrita en el Capítulo VII. El insumo de las simulaciones son datos reales de las bitácoras de OSC Linux Cluster y se aplicaron dos modelos de administración de recursos: tiempo compartido y espacio compartido, los cuales fueron explicados anteriormente.

Dentro de este experimento, se utilizaron las tareas del Sistema OSC Linux Cluster. Se ejecutaron empleando varios recursos y procesos de ejecución dentro de cada recurso. El

código fuente de esta simulación se encuentra en el Anexo II. Los recursos usados durante la simulación poseen las siguientes características:

- Velocidad de red: 1 Gigabit por segundo
- Retardos de propagación: 10
- Máxima unidad de transmisión: 100,000 bytes
- Costo de los recursos: 3 G\$⁴⁶ por EP⁴⁷
- Conexiones mediante enrutadores de 10 Gibts por segundo
- Los recursos son computadores Sun Ultra Sparc con sistema operativo Solaris

En el Cuadro N° 28 y Cuadro N° 29, se observa la duración del envío de datos y recepción de resultados desde el grid de procesamiento. Los resultados indican que –prácticamente– no existe diferencia en el tiempo de transporte de información. Este fenómeno es de esperar puesto que los dos modelos utilizan la misma infraestructura de comunicación; por tanto, el tiempo de transporte de información no depende de los algoritmos internos de procesamiento del grid, sino de la capacidad de infraestructura de red.

⁴⁶ G\$ es el acrónimo para Grid Units por Entidad de Procesamiento, la cual es una medida de costo estándar definida dentro del modelo de operación de GridSim

⁴⁷ Los Elementos de Procesamiento es la unidad mínima de procesamiento de información, los cuales se utilizan para medir la cantidad de información que puede procesar un nodo de informática distribuida. Esta unidad es definida en Buyya Rajkumar. (2005). *Grid Computing Info Centre*. Recuperado el 11 de junio del 2005, de <http://www.GridComputing.com>.

Cuadro N° 28 OSC Linux Cluster Tiempo de transporte modelo tiempo compartido

OSC Linux - Tiempo Compartido				
Tiempo de transporte de Información				
Usuario	Tiempo de Ida y Retorno (segundos)	Número de Saltos	Ancho Banda (bits)	Tiempo Envío Recepción de Trabajos
Usuario 00	0,07280	3,00000	1.000.000,00	18,882000
Usuario 01	0,07280	3,00000	1.000.000,00	18,892000
Usuario 02	0,07280	3,00000	1.000.000,00	18,892000

Fuente: Elaboración propia con resultados de las simulaciones en GridSim

Cuadro N° 29 OSC Linux Cluster Tiempo de transporte modelo espacio compartido

OSC Linux - Espacio Compartido				
Tiempo de transporte de Información				
Usuario	Tiempo de Ida y Retorno (segundos)	Número de Saltos	Ancho Banda (bits)	Tiempo Envío Recepción de Trabajos
Usuario 00	0,07280	3,00000	1.000.000,00	18,422000
Usuario 01	0,07280	3,00000	1.000.000,00	18,432000
Usuario 02	0,07280	3,00000	1.000.000,00	18,442000

Fuente: Elaboración propia con resultados de las simulaciones en GridSim

En el Cuadro N° 30, Cuadro N° 31 y Cuadro N° 21, se muestran los tiempos que dura la atención de las tareas dentro del grid. Los resultados revelan que los tiempos de atención del modelo de tiempo compartido y del modelo de espacio compartido son los mismos. Cabe recordar que la infraestructura de componentes del grid son los mismos, lo único que cambia es el modelo de asignación de recursos; por tanto, es de esperar que los estos tiempos de atención sean iguales.

Cuadro N° 30 OSC Linux Cluster tiempo de duración modelo espacio compartido

OSC Linux - Tiempo Compartido				
Duración Usuario 00				
Entidad	Tiempo Entrada	Tiempo Salida	Duración	Ancho Banda
Usuario 00	18,80520	18,80920	0,00400	1.000.000,00
Enrutador 01	18,81920	18,81960	0,00040	10.000.000,00
Enrutador 02	18,82960	18,83360	0,00400	1.000.000,00
Recurso 01	18,84360	18,84760	0,00400	1.000.000,00
Enrutador 02	18,85760	18,85800	0,00040	10.000.000,00
Enrutador 01	18,86800	18,87200	0,00400	1.000.000,00
Usuario 00	18,88200	N/A		N/A

Fuente: Elaboración propia con resultados de las simulaciones en GridSim

Cuadro N° 31 OSC Linux Cluster tiempo de duración modelo espacio compartido

OSC Linux - Espacio Compartido				
Duración Usuario 00				
Entidad	Tiempo Entrada	Tiempo Salida	Duración	Ancho Banda
Usuario 00	18,34520	18,34920	0,00400	1.000.000,00
Enrutador 01	18,35920	18,35960	0,00040	10.000.000,00
Enrutador 02	18,36960	18,37360	0,00400	1.000.000,00
Recurso 01	18,38360	18,38760	0,00400	1.000.000,00
Enrutador 02	18,39760	18,39800	0,00040	10.000.000,00
Enrutador 01	18,40800	18,41200	0,00400	1.000.000,00
Usuario 00	18,42200	N/A	N/A	N/A

Fuente: Elaboración propia con resultados de las simulaciones en GridSim

En el Cuadro N° 32, se muestra el costo de los trabajos de procesamiento para cada uno de los modelos de asignación: tiempo compartido y espacio compartido . La unidad de costo utilizada es G\$; los resultados indican que el costo de procesamiento es más alto si se utiliza un esquema de asignación de recursos basado en el modelo de tiempo compartido, específicamente el costo de este modelo es 4,5 G\$ más alto, lo cual representan un costo adicional de 2,34%.

Cuadro N° 32 OSC Linux Cluster Comparación de Costos

OSC Linux Cluster Costos Usuario 00					
Id Trabajo	Estado	Recurso	Costo Tiempo Compartido	Costo Espacio Compartido	
0	Exitoso	10	37,50	39,00	
1	Exitoso	10	40,38	38,88	
2	Exitoso	10	39,95	38,45	
3	Exitoso	10	39,83	38,33	
4	Exitoso	10	39,40	37,90	
Total			197,05	192,55	

Fuente: Elaboración propia con resultados de las simulaciones en GridSim

De igual manera, se realizaron simulaciones con más tareas de procesamiento. El resultado de estas pruebas se muestra en el Cuadro N° 33 y es consistente con los resultados anteriores. De nuevo, el costo del modelo de tiempo compartido es superior, específicamente 4,369.73 G\$, lo cual indica que el costo de aplicación de este modelo es 39.82% superior al de espacio compartido.

Cuadro N° 33 OSC Linux Cluster Comparación de Costos Tareas Personalizadas

OSC Linux - Tiempo Compartido					
Costos Tareas Personalizadas					
Id Trabajo	Estado	Recurso	Costo Tiempo Compartido	Costo Espacio Compartido	
1	Exitoso	5	291,35	183,00	
2	Exitoso	5	303,00	183,00	
3	Exitoso	5	529,57	363,00	
4	Exitoso	5	993,80	813,00	
5	Exitoso	5	548,30	363,00	
6	Exitoso	5	733,58	540,11	
7	Exitoso	5	1.855,48	1.623,00	
8	Exitoso	5	351,07	183,00	
9	Exitoso	5	477,00	243,00	
10	Exitoso	5	1.059,79	813,00	
11	Exitoso	5	785,29	543,00	
12	Exitoso	5	802,13	540,04	
13	Exitoso	5	532,12	273,00	
14	Exitoso	5	804,59	543,00	
15	Exitoso	5	360,00	183,00	
16	Exitoso	5	801,59	543,00	
17	Exitoso	5	360,00	183,00	
18	Exitoso	5	372,23	153,00	
19	Exitoso	5	612,23	273,00	
20	Exitoso	5	2.770,73	2.433,00	
Total			15.343,87	10.974,14	

Fuente: Elaboración propia con resultados de las simulaciones en GridSim

Los resultados de las simulaciones realizadas con el Sistema OSC Linux Cluster, muestran que la escogencia de un adecuado modelo asignación y calendarización de recursos incide directamente en los costos relacionados con su procesamiento. Por tanto, es tarea de los administradores del grid elegir un modelo de operación adecuado.

3. Cálculo del Índice de Gestión Económica (IGE)

En este apartado, se realiza el cálculo del Índice de Gestión Económica con base en los indicadores estudiados hasta el momento. Para realizar el cálculo de este índice, se utiliza el procedimiento especificado en el apartado de Análisis de Indicadores del Capítulo de Metodología.

En el Cuadro N° 34, se muestra el procedimiento de cálculo del Índice de Gestión Económica. La información está compuesta de la siguiente forma:

- En la columna “Resultado del Indicador”, para cada indicador se muestra el resultado obtenido por SDSC Blue y OSC Linux en cada uno de los años de estudio. Para ver el detalle de cada uno de estos indicadores, refiérase a la sección anterior.
- La columna Max se utiliza como medio de normalización de los resultados de los indicadores. La normalización es necesaria debido a que los indicadores poseen unidades de medida distinta y para calcular el IGE, resulta preciso que posean una base común.
- La columna “Resultado del Indicador Normalizado” incluye el resultado de la normalización del indicador con base en valor de la columna Max.
- IGE Bruto corresponde a la suma de los valores normalizados de cada uno de los indicadores. Debido a que se asignó un peso Max de 10 el valor máximo que puede alcanzar el IGE Bruto, es de 60 puntos.
- IGE Porcentual corresponde al porcentaje del total de puntos obtenidos por un sistema de grid computing en un año dado.
- IGE Normalizado corresponde a la normalización de los puntos obtenidos con base el peso asignado al IGE dentro del cálculo del IBECOD. Para efectos de cálculo, en este estudio se le asignó al IGE un peso de 33% del total de IBECOD; sin embargo, este parámetro podría variarse dependiendo de la importancia que se le desee asignar a cada componente del IBECOD.

Cuadro N° 34 Cálculo del Índice de Gestión Económica

Indicador	Unidad Medida	Resultado del Indicador						Resultado del Indicador Normalizado						
		SDSC Blue			OSC Linux			Max	SDSC Blue			OSC Linux		
		2000	2001	2002	2000	2001	2002		2000	2001	2002	2000	2001	2002
Ingresos en la empresa (promedio por tarea procesada)	\$ *	40.28	56.82	51.41	320.78	851.67	1,443.23	10	1.26	0.67	0.36	10.00	10.00	10.00
Evolución de los ingresos de las empresas a través de los años	%	100%	64%	-11%	100%	62%	43%	10	10.00	6.40	(1.10)	10.00	6.20	4.30
Porcentaje de cambio en la eficiencia a través de los años (tareas no atendidas)	%	25%	20%	27%	0%	0%	0%	10	2.50	2.00	2.70	-	-	-
Tareas atendidas	%	75%	80%	73%	100%	100%	100%	10	7.50	8.00	7.30	10.00	10.00	10.00
Cambio en los costos debido a la aplicación de distintos modelos de asignación	G\$ **	4.31	4.31	4.31	4,369.73	4,369.73	4,369.73	10	10.00	10.00	10.00	10.00	10.00	10.00
Porcentaje de cambio en los costos debido a la aplicación de distintos modelos de asignación	%	2.24%	2.24%	2.24%	39.82%	39.82%	39.82%	10	10.00	10.00	10.00	10.00	10.00	10.00
IGE Bruto								60.00	41.26	37.07	29.26	50.00	46.20	44.30
IGE Porcentual								100.00	69%	62%	49%	83%	77%	74%
IGE Normalizado								33.33	22.92	20.59	16.25	27.78	25.66	24.61

Fuente: Elaboración Propia

En el Cuadro N° 34, se muestra el resultado del cálculo del IGE Normalizado. Para el año 2000, el sistema SDSC Blue obtuvo un IGE de 22,92 mientras que OSC Linux presenta un IGE de 27,78. Por tanto, durante ese año la gestión económica de OSC Linux fue superior, Esto es consistente con el resultado de los indicadores individuales, en los cuales se aprecia que OSC Linux posee ingresos superiores y tasas de atención de tareas exitosas de 100%. Este comportamiento se mantiene durante el resto de los años. OSC Linux posee un IGE superior, lo cual indica que -según el modelo planteado en este trabajo- la gestión económica de SDSC Blue debe mejorarse y para realizarlo es necesario estudiar cada uno de los indicadores individualmente y tomar las medidas necesarias para mejorarlos.

C. Índice de Gestión Operacional (IGO)

1. Introducción

El análisis de gestión operacional considera sistemas de informática distribuida representativa y utiliza las bitácoras de trabajo recolectadas por Dror Feitelson (2006). La fuente de datos utilizada para realizar este análisis se basa en el Standard Workload Format especificado por Chapin y Cirne (1999). Este modelo fue definido con el fin de facilitar el uso de bitácoras de trabajo de centros de procesamiento de datos. Este estándar posee las siguientes características:

- Las cargas de trabajo son almacenadas en un archivo de texto.
- Cada carga de trabajo es representada por una línea dentro del archivo.
- Se utiliza el mismo formato, tanto para registrar las cargas de trabajo y como para describir las características de los recursos que posee la infraestructura del grid.

En este análisis, se incluirán las siguientes infraestructuras de grid:

- SDSC Blue Horizon (San Diego Supercomputer Center)
- Sistema OSC Linux Cluster (Ohio Supercomputer Center).

Los Indicadores de Gestión Operacional permiten determinar la eficiencia con que se están utilizando los recursos de hardware y software que componen el grid, su tiempo de operación y el tiempo de respuesta. De esta manera, se puede determinar cuál es la calidad de servicio brindado (tiempo de atención y respuesta de los procesos enviados al grid, disponibilidad de los recursos para ser utilizados por los clientes, tiempo de espera para que la carga de información recibida sea atendida, entre otros). El Cuadro N° 35 resume los indicadores que componen el Índice de Gestión Operacional (IGO).

Cuadro N° 35 Índice de Gestión Operacional (IGO)

Indicador	Unidad de medida
Indicador de tiempo de espera por mes	Segundos
Indicador tiempo de espera por día de la semana	Segundos
Indicador de tiempo de ejecución de las tareas por mes	Segundos
Indicador cantidad de trabajos atendidos por mes	EP ⁴⁸ /Proceso
Indicador de memoria utilizada por mes	Megabytes
Indicador cantidad de trabajos recibidos por día	EP/Día
Indicador cantidad de trabajos atendidos según hora del día	EP/Hora

Fuente: Elaboración propia.

La descripción de cada uno de estos indicadores se presenta a continuación.

- **Indicador de tiempo de espera por mes (ITEM).** Mide cuánto tiempo, expresado en segundos, tiene que esperar una tarea desde que es recibida en el grid hasta que se inicia su atención. En este índice se toma en segundos como unidad de trabajo.
- **Indicador tiempo de espera por día de la semana (ITEDS).** Permite identificar los patrones de uso del grid y, principalmente, conocer cuál es el día de la semana cuando las colas de procesamiento aumentan. Si se conoce este comportamiento, será posible crear estrategias que permitan distribuir la carga de procesamiento de forma equitativa durante la semana.
- **Indicador de tiempo de ejecución de las tareas por mes (TETM).** Permite identificar cuánto tiempo tardan las tareas en ser procesadas dentro del grid; de esta forma, se pueden tomar las medidas necesarias para aumentar la eficiencia de ejecución de las tareas.

⁴⁸ EP (Elementos de Procesamiento) es la medida con la cual se define qué tan exigente -en cuanto a procesamiento- es un proceso determinado. Generalmente están dados en diferentes unidades de velocidad tales como MIPS o SPEC

- **Indicador cantidad de trabajos atendidos por mes (CTAM).** Posibilita determinar cuáles son los meses más ocupados y -de esa forma- preparar el grid para atender de forma adecuada las tareas enviadas en los meses de mayor actividad.
- **Indicador de memoria utilizada por mes (CMUM).** Por medio de él, se determina en cuáles meses se requiere que el grid de procesamiento posea más o menos memoria, de forma tal que se puedan atender –adecuadamente- las tareas.
- **Indicador cantidad de trabajos recibidos por día (CTRD).** A partir de este, se establece los días cuando se presenta mayor demanda por parte de los consumidores. Con este indicador se puede establecer estrategias adecuadas para que las cargas de procesamiento sean atendidas adecuadamente en aquellos días que se presentan mayor actividad.
- **Indicador cantidad de trabajos atendidos según hora del día (CTAHD).** Permite fijar cuál es el comportamiento de las tareas recibidas y saber cuáles son los periodos críticos de atención de estas.

A continuación, se presenta el análisis de indicadores de gestión operacional para cada uno de los sistemas de grid computing sujetos de estudio.

2. SDSC Blue Horizon (San Diego Supercomputer Center)

El sistema SDSC Blue Horizon -en promedio- atiende entre 4.087,00 y 8.108,00 tareas de procesamiento por mes, las cuales -en promedio- requieren 10.165,33 megabytes de memoria. Desde el momento cuando las tareas son enviadas al grid, transcurre un periodo de 9 horas para que inicie su procesamiento.

A continuación se analiza, en detalle, cada uno de los indicadores de Eficiencia en Infraestructura.

a) Indicador de tiempo de espera por mes

El Cuadro N° 49 incluye los datos de tiempo de espera del sistema SDSC Blue. Se incluyen los datos del periodo del año 2000 al 2002. Se observa que en el año 2001 se dio una reducción de un 21% en el tiempo de espera; sin embargo, para el 2002 el tiempo de espera se incrementó en un 90%, a pesar de que el promedio de las tareas se mantuvo prácticamente igual: de 97,290 a 96,980. Este aumento del tiempo de espera puede estar ligado a que las tareas enviadas son más complejas desde un punto de vista de cálculo. Por lo tanto, sería recomendable evaluar una posible actualización de la infraestructura.

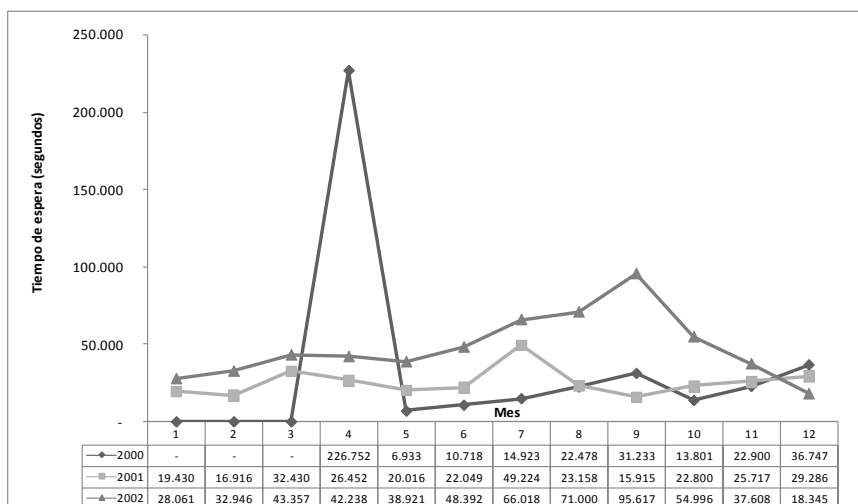
Cuadro N° 36 SDSC Blue Tiempo de espera promedio por mes

Mes	2000	2001	2002
Enero	-	19,430	28,061
Febrero	-	16,916	32,946
Marzo	-	32,430	43,357
Abril	226,752	26,452	42,238
Mayo	6,933	20,016	38,921
Junio	10,718	22,049	48,392
Julio	14,923	49,224	66,018
Agosto	22,478	23,158	71,000
Setiembre	31,233	15,915	95,617
Octubre	13,801	22,800	54,996
Noviembre	22,900	25,717	37,608
Diciembre	36,747	29,286	18,345
Promedio anual	32,207	25,283	48,125

Fuente: Elaboración propia con datos de las bitácoras de SDSC Blue.

El Gráfico N° 23 muestra cómo el tiempo de espera se incrementó durante el año 2002 e, incluso, se indica un punto alto en setiembre de ese año. Si se mantiene esa tendencia, implica que este grid debe preparar su infraestructura; así, puede crecer su capacidad de procesamiento durante esa parte del año y se bajará este índice de espera.

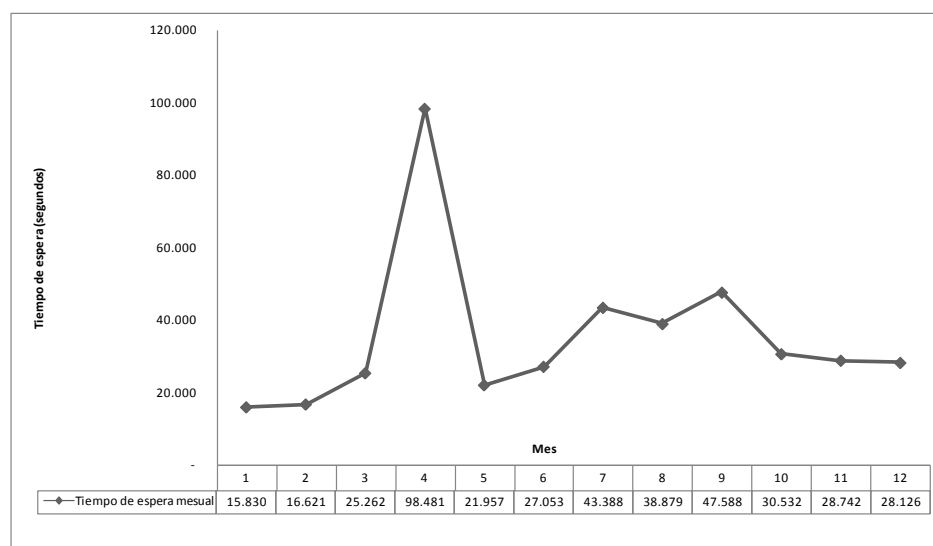
Gráfico N° 23 SDSC Blue Tiempo de espera promedio por mes



Fuente: Elaboración propia con datos de las bitácoras de SDSC Blue.

Para el grid de SDSC Blue, -en promedio- en los meses de junio y julio se presenta un aumento en el tiempo de espera en un 23% y un 60% respectivamente; en setiembre se da un aumento del 22%.

En general, durante estos meses, se producen los aumentos más altos en tiempo de espera, lo cual está muy relacionado con el hecho de que en esos meses se presenta un aumento del 47% en la cantidad de trabajos enviados al grid para su procesamiento. Esto indica que el grid no está preparado para administrar grandes aumentos en cuanto a las tareas que deben procesarse.

Gráfico N° 24 SDSC Blue Tiempo global de espera promedio por mes

Fuente: Elaboración propia con datos de las bitácoras de SDSC Blue.

b) Indicador tiempo de espera por día de la semana

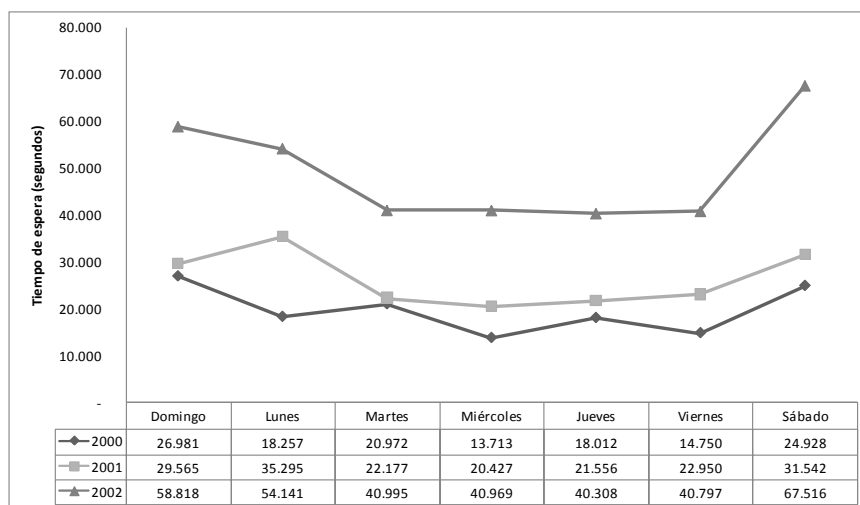
En cuanto SDSC, el fin de semana es el periodo en el cual las colas de atención crecen: específicamente, el sábado y domingo (Ver Cuadro N° 37). Por ejemplo, en el año 2002, el tiempo de espera es de 67,516 segundos y 58,818 segundos respectivamente.

Cuadro N° 37 SDSC Blue Tiempo de espera por día de la semana

Día de la semana	2000	2001	2002
Domingo	26,981	29,565	58,818
Lunes	18,257	35,295	54,141
Martes	20,972	22,177	40,995
Miércoles	13,713	20,427	40,969
Jueves	18,012	21,556	40,308
Viernes	14,750	22,950	40,797
Sábado	24,928	31,542	67,516
Promedio semanal	19,659	26,216	49,078

Fuente: Elaboración propia con datos de las bitácoras de SDSC Blue.

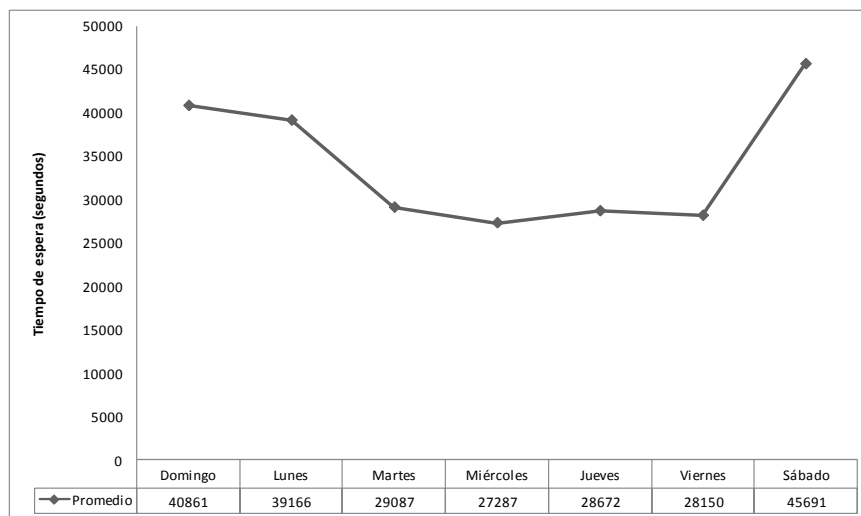
Gráfico N° 25 SDSC Blue Tiempo de espera por día de la semana



Fuente: Elaboración propia con datos de las bitácoras de SDSC Blue.

El aumento de tiempo de espera durante los fines de semana es una tendencia que se ha mantenido desde el año 2000 (Ver Gráfico N° 25); sin embargo, el tiempo de espera ha aumentado conforme pasan los años. En el año 2001, el tiempo de espera -los domingos- aumentó un 10% y el de los viernes un 56%, lo cual afecta de forma significativa la calidad de servicio que se da a los clientes.

Gráfico N° 26 SDSC Blue Tiempo de espera promedio por día de la semana



Fuente: Elaboración propia con datos de las bitácoras de SDSC Blue.

c) Indicador de tiempo de ejecución de las tareas por mes

En cuanto al sistema SDSC Blue, el tiempo de duración de las tareas se mantiene estable, tal y como se observa en el Cuadro N° 38. En el año 2001, crece un 6% y en el 2002 únicamente un 2%. A pesar de que estos aumentos son bajos, lo ideal sería que tiendan a bajar en lugar de subir; pues si logra mayor eficiencia, se reducen los tiempos de ejecución- lo cual repercute en un aumento de las ganancias del grid.

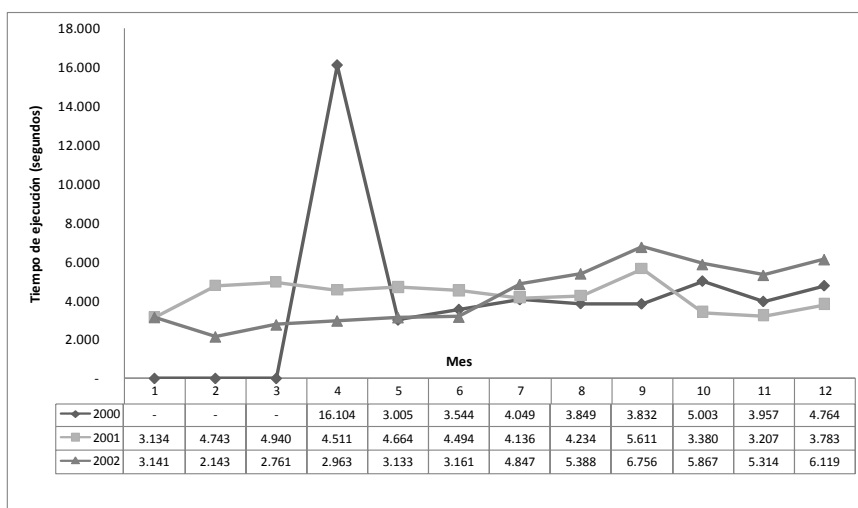
Cuadro N° 38 SDSC Blue Duración de las tareas dentro del grid (en segundos)

Mes	2000	2001	2002
Enero	-	3,134	3,141
Febrero	-	4,743	2,143
Marzo	-	4,940	2,761
Abril	16,104	4,511	2,963
Mayo	3,005	4,664	3,133
Junio	3,544	4,494	3,161
Julio	4,049	4,136	4,847
Agosto	3,849	4,234	5,388
Setiembre	3,832	5,611	6,756
Octubre	5,003	3,380	5,867
Noviembre	3,957	3,207	5,314
Diciembre	4,764	3,783	6,119
Promedio anual	4,009	4,236	4,299

Fuente: Elaboración propia con datos de las bitácoras de SDSC Blue.

El Gráfico N° 27 muestra cómo la duración del cumplimiento de las tareas se mantiene estable alrededor de los 2,000 y 6,000 segundos por tarea.

Gráfico N° 27 SDSC Blue Tiempo de duración de las tareas dentro del grid

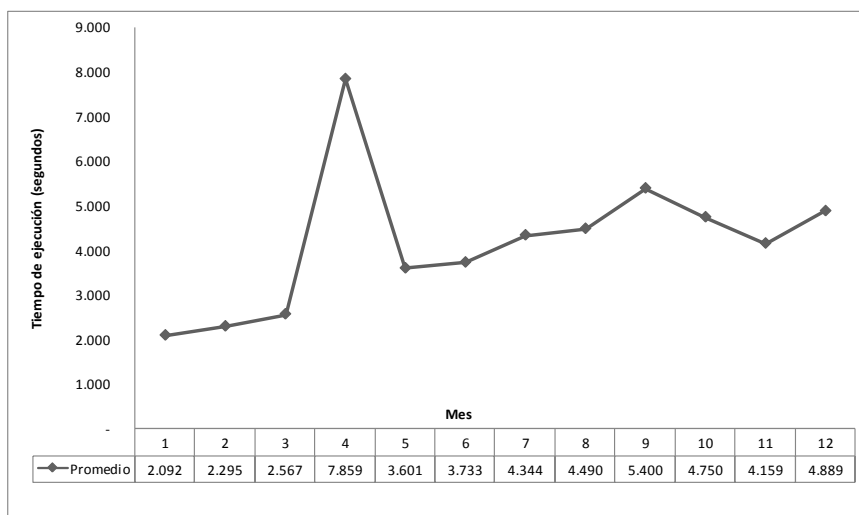


Fuente: Elaboración propia con datos de las bitácoras de SDSC Blue.

Otro comportamiento que se observa en el Gráfico N° 28, consiste en que el tiempo de procesamiento de las tareas es de 2,000 segundos en enero y va aumentando poco a poco hasta llegar a los 5,400 segundos en setiembre y 6,119 segundos en diciembre.

Este comportamiento se transforma en una indicación para los administradores de grid, con el propósito de que en esos meses logre optimizar el uso de la infraestructura o, si es posible, realizar ajustes mediante los cuales se logre agregar más recursos al grid durante esos meses. Estos recursos pueden ser propios o se realizan alianzas estratégicas con otros grids -de forma tal- que se baje el tiempo de atención de las tareas. De esta manera, los clientes se ven beneficiados y se podría crear un sentido de lealtad hacia el grid, pues los clientes verán que éste es capaz de procesar la información eficientemente.

Gráfico N° 28 SDSC Blue Tiempo promedio de duración de las tareas dentro del grid



Fuente: Elaboración propia con datos de las bitácoras de SDSC Blue.

d) Indicador cantidad de trabajos atendidos por mes e Indicador de memoria utilizada por mes

El sistema SDSC Blue atiende más de 90,000 tareas al año. Tal y como se observa en el Cuadro N° 39, el promedio de tareas mensual fue de 8,108 en el 2001 y de 8,082 en el 2002, lo cual es muy estable. De la misma forma, la memoria requerida es de 10,665 en promedio mensual en el 2001 y de 10,426 en el 2002. Así, la administración del grid puede saber la cantidad de tareas que debe atender por año y adecuar su infraestructura para que sea eficiente.

Una tarea importante que debe emprender la administración de este grid, es la atracción de tareas -siempre y cuanto- se cuente con la capacidad necesaria para atender el crecimiento de la carga.

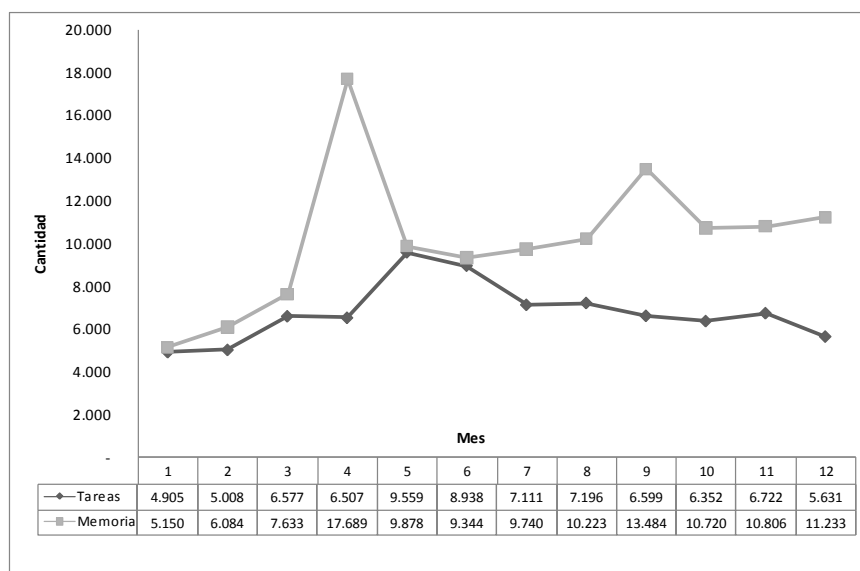
Cuadro N° 39 SDSC Blue Cantidad de memoria y trabajos por mes

Mes	2000		2001		2002	
	Tareas	Memoria	Tareas	Memoria	Tareas	Memoria
Enero	-	-	6,632	7,635	8,082	7,814
Febrero	-	-	7,591	11,228	7,434	7,025
Marzo	-	-	6,939	13,727	12,791	9,171
Abril	12	33,308	7,881	11,360	11,627	8,399
Mayo	8,630	10,598	9,068	10,294	10,979	8,742
Junio	7,842	9,948	8,660	10,088	10,313	7,997
Julio	6,534	8,567	8,069	10,118	6,729	10,536
Agosto	6,628	9,081	8,636	9,472	6,325	12,115
Setiembre	5,935	8,867	8,630	17,915	5,231	13,669
Octubre	4,666	10,819	8,488	8,753	5,902	12,588
Noviembre	5,004	10,993	8,964	8,260	6,198	13,164
Diciembre	3,793	10,684	7,732	9,127	5,369	13,887
Promedio	4,087	9,405	8,108	10,665	8,082	10,426

Fuente: Elaboración propia con datos de las bitácoras del SDSC Blue.

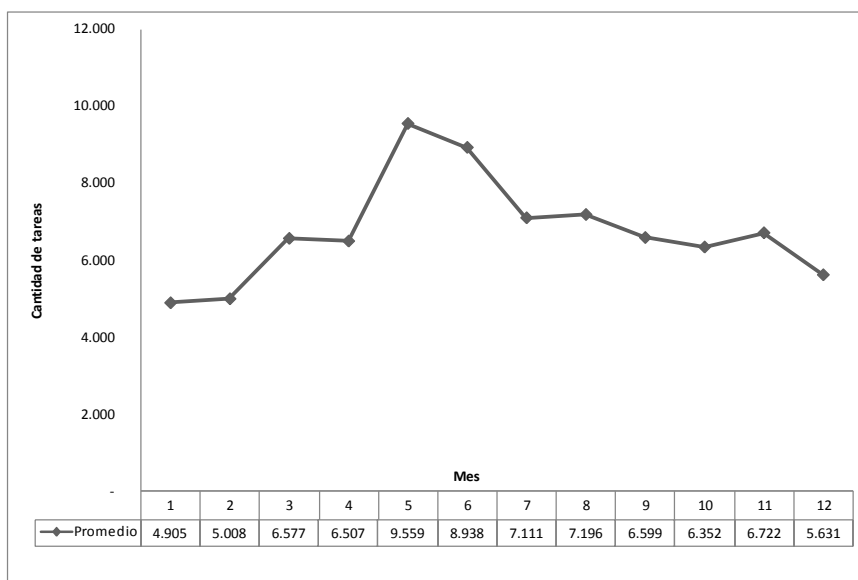
En el Gráfico N° 29, se observa la relación existente entre la cantidad de trabajos atendidos y la memoria requerida. En los meses de abril y setiembre, se presenta un aumento en la cantidad de memoria utilizada, a pesar de que la cantidad de tareas baja. Esto podría deberse a que -durante esos meses- se atienden tareas que requieren gran cantidad de almacenamiento temporal para ser procesadas. Una opción sería que la administración del grid agregue más memoria durante esos periodos de alta demanda, de forma tal, que no se sufran demoras en la atención de las cargas de información, especialmente entre los meses de mayo y junio que son los meses durante los cuales se atienden más tareas por mes, como se observa en el Gráfico N° 30.

Gráfico N° 29 SDSC Blue Relación entre memoria requerida y tareas atendidas por mes



Fuente: Elaboración propia con datos de las bitácoras de SDSC Blue.

Gráfico N° 30 SDSC Blue cantidad de tareas atendidas por mes



Fuente: Elaboración propia con datos de las bitácoras de SDSC Blue.

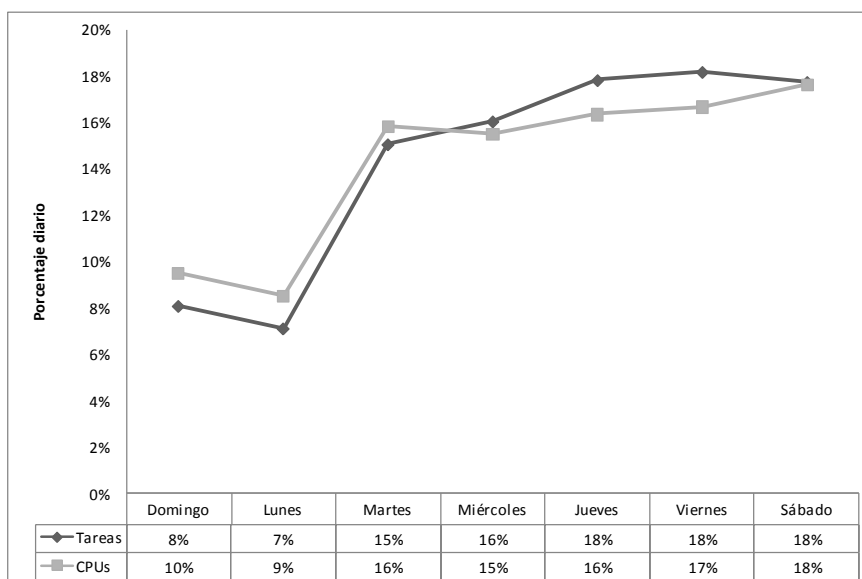
e) Indicador cantidad de trabajos recibidos por día

Los datos sobre la cantidad de trabajos recibidos por día en el sistema SDSC Blue, se muestran en el Cuadro N° 40 y en el Gráfico N° 31. Se puede apreciar que existe una tendencia muy marcada, en la cual el 85% de las tareas se reciben entre el martes y el sábado de cada semana. Esto permite que la administración del grid establezca previsiones necesarias, pues para disminuir la cantidad de nodos disponibles los días domingos y lunes e incentivar el uso del grid durante esos días.

Cuadro N° 40 SDSC Blue Cantidad de trabajos recibidos por día

Día	2000 Tareas	2001 Tareas	2002 Tareas
Domingo	3.651	7.582	8.389
Lunes	3.347	6.950	6.960
Martes	7.471	14.876	14.277
Miércoles	7.826	16.372	14.834
Jueves	8.806	17.203	17.363
Viernes	9.073	16.659	18.502
Sábado	8.870	17.648	16.655
Promedio	7.006	13.899	13.854

Fuente: Elaboración propia con datos de las bitácoras de SDSC Blue.

Gráfico N° 31 SDSC Blue Porcentaje de tareas recibidas por día y procesadores utilizados por día

Fuente: Elaboración propia con datos de las bitácoras de SDSC Blue.

f) Indicador cantidad de trabajos atendidos según hora del día

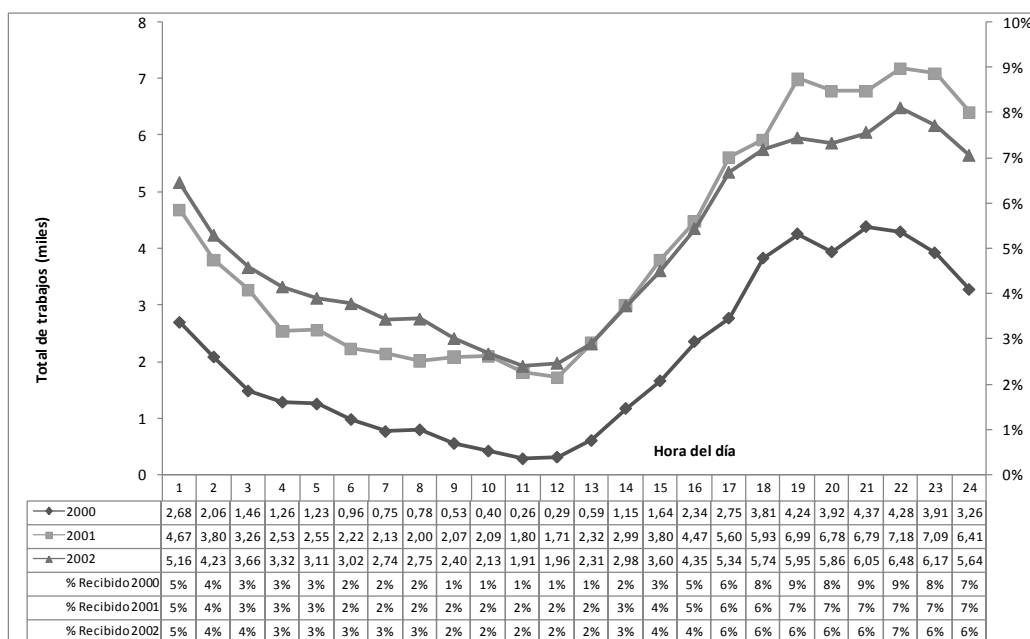
En el caso del sistema SDSC Blue, cuya información se muestra en el Cuadro N° 41, las horas en la cuales se recibe mayor cantidad de trabajo es entre las doce medio día y las doce media noche; esto representa el 67% de las tareas atendidas por día (Ver Gráfico N° 32 y Gráfico N° 33). La administración del grid debe tomar las medidas necesarias para que se asegure la continuidad del negocio durante estas horas.

Cuadro N° 41 SDSC Blue Cantidad de trabajos atendidos según hora del día

Hora del día	2000	2001	2002
-	2,681	4,676	5,167
1	2,069	3,802	4,234
2	1,467	3,267	3,661
3	1,269	2,532	3,320
4	1,236	2,555	3,119
5	964	2,220	3,024
6	751	2,131	2,747
7	781	2,009	2,750
8	539	2,077	2,406
9	405	2,093	2,137
10	268	1,803	1,915
11	298	1,715	1,969
12	592	2,324	2,312
13	1,159	2,996	2,984
14	1,646	3,803	3,605
15	2,342	4,479	4,351
16	2,755	5,607	5,348
17	3,817	5,931	5,748
18	4,247	6,999	5,957
19	3,927	6,785	5,864
20	4,375	6,796	6,054
21	4,282	7,185	6,488
22	3,911	7,094	6,173
23	3,263	6,411	5,647

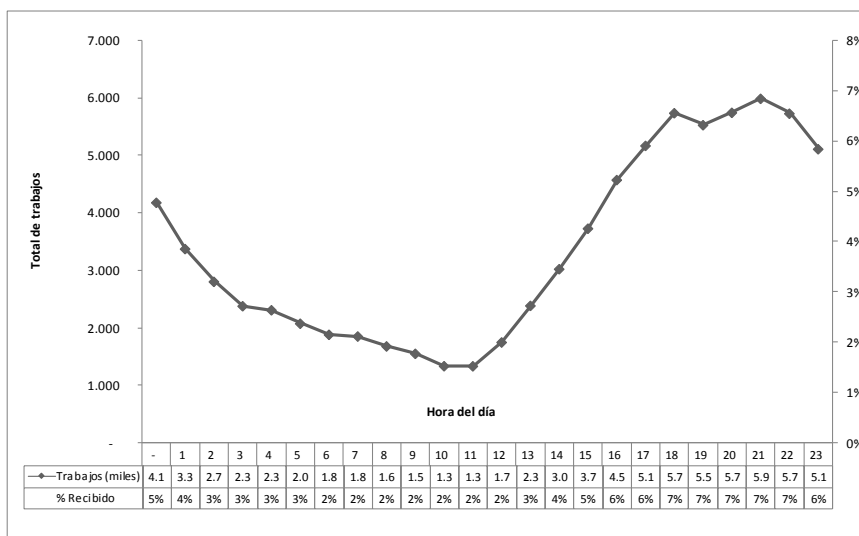
Fuente: Elaboración propia con datos de las bitácoras de SDSC Blue.

Gráfico N° 32 SDSC Blue Cantidad de trabajos atendidos según hora del día



Fuente: Elaboración propia con datos de las bitácoras de SDSC Blue.

Gráfico N° 33 SDSC Blue Promedio de cantidad de trabajos atendidos según hora del día



Fuente: Elaboración propia con datos de las bitácoras de SDSC Blue.

3. Sistema OSC Linux Cluster (Ohio Supercomputer Center)

a) Indicador de tiempo de espera por mes

El Cuadro N° 42 muestra los tiempos de espera promedio de OSC Linux Cluster para los años 2001 al 2002. Con respecto a este sistema de informática distribuida, se presenta un aumento promedio del 113% por ciento en el tiempo de espera.

Cuadro N° 42 OSC Linux Cluster Tiempo de espera por mes

Mes	2000	2001	2002
Enero	447	4,794	2,621
Febrero	4,160	4,297	4,229
Marzo	5,004	26,868	15,936
Abril	5,739	36,032	20,886
Mayo	10,839	14,846	12,843
Junio	6,084	13,915	10,000
Julio	10,441	5,909	8,175
Agosto	5,773	6,570	6,172
Setiembre	3,428	12,611	8,020
Octubre	2,281	12,491	7,386
Noviembre	8,672	2,019	5,346
Diciembre	2,898	8,265	5,581
Promedio	5,481	12,385	8,933

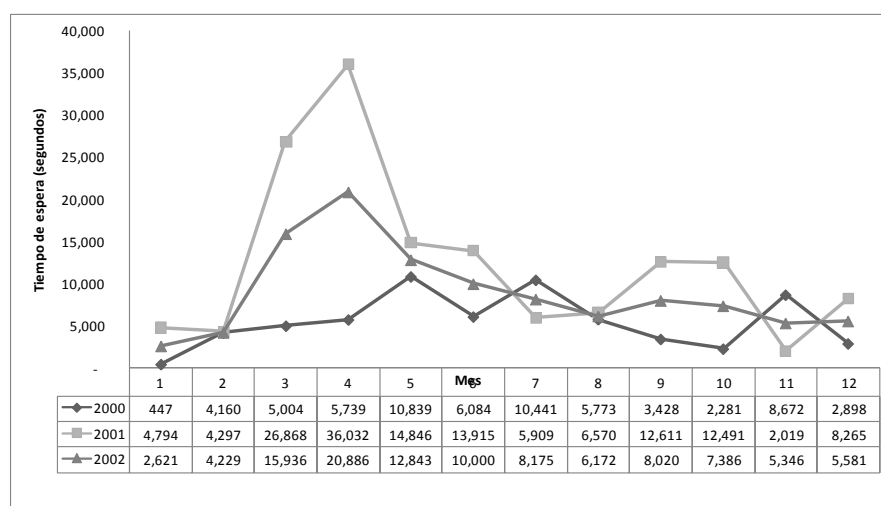
Fuente: Elaboración propia con datos de las bitácoras de OSC Linux Cluster.

En el año 2001, se presentaron grandes aumentos en el tiempo de espera (Ver Gráfico N° 34) especialmente durante los meses de marzo y abril: estos son de un 35% y 34% respectivamente, para bajar un 59% en el mes de mayo y se mantiene sin muchas alteraciones hasta el final del año. Esto podría deberse a un aumento de un 59% en la

cantidad de tareas recibidas, lo cual provoca que los recursos del grid sean insuficientes para atender la demanda. La misma situación de aumento de tiempo de espera se presenta en un 11% durante agosto del 2001 y un 92% en setiembre, debido a un aumento del 88% en la cantidad de trabajos recibidos por el grid en el mes de agosto.

Este comportamiento también indica que cuando se acrecienta de forma abrupta la cantidad de trabajos recibidos para su procesamiento en el grid, la infraestructura colapsa y los trabajos empiezan a ser atendidos de forma más lenta. Sin embargo, una vez que se han atendido estos trabajos extraordinarios, el grid vuelve a tiempos de respuesta promedio. Con estos datos, se puede determinar que una vez que se ha detectado un aumento importante en la cantidad de trabajos recibidos, es necesario que el grid se ajuste automáticamente y se agregue mayor poder computacional a este, lo cual puede lograrse mediante la adición temporal de nodos al grid durante tiempos de alta demanda.

Gráfico N° 34 OSC Linux Cluster Tiempo de espera por mes

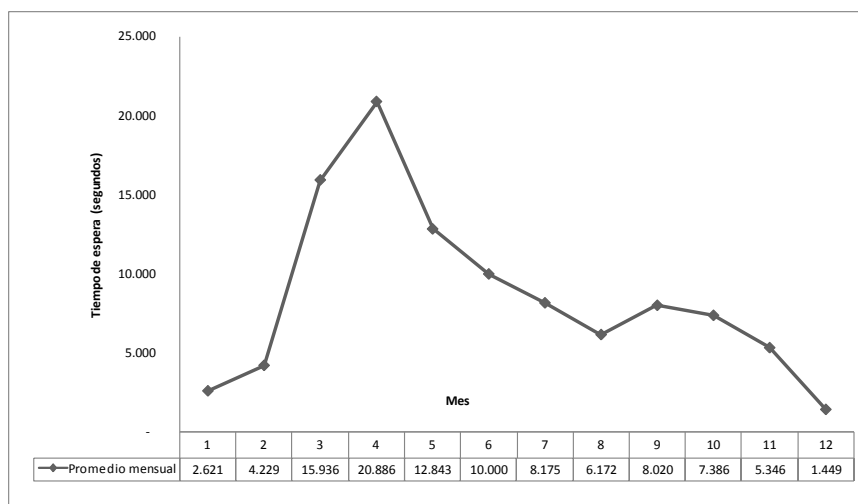


Fuente: Elaboración propia con datos de las bitácoras OSC Linux Cluster.

El comportamiento creciente de tiempo de espera se muestra en el Gráfico N° 35. En general, se presentan dos puntos importantes de saturación del grid en cuanto a

procesamiento: el primero en abril y el segundo en setiembre. Si se identifica esta estacionalidad de la demanda, será posible que se prepare la infraestructura para hacer frente a estas alzas repentinas de la demanda.

Gráfico N° 35 OSC Linux Cluster Tiempo de espera promedio por mes



Fuente: Elaboración propia con datos de las bitácoras de OSC Linux Cluster.

b) Indicador tiempo de espera por día de la semana

En el Cuadro N° 43, se muestra el tiempo de espera promedio del sistema OSC Linux Cluster según el día de la semana. En este sistema, los tiempos de espera del 2001 aumentaron con respecto al 2000 en un 93% en promedio y los días más críticos han sido los domingo, miércoles y sábado con un 280%, 270% y un 154%, respectivamente; a pesar de que la cantidad de tareas disminuyó entre un 17% y 16%.

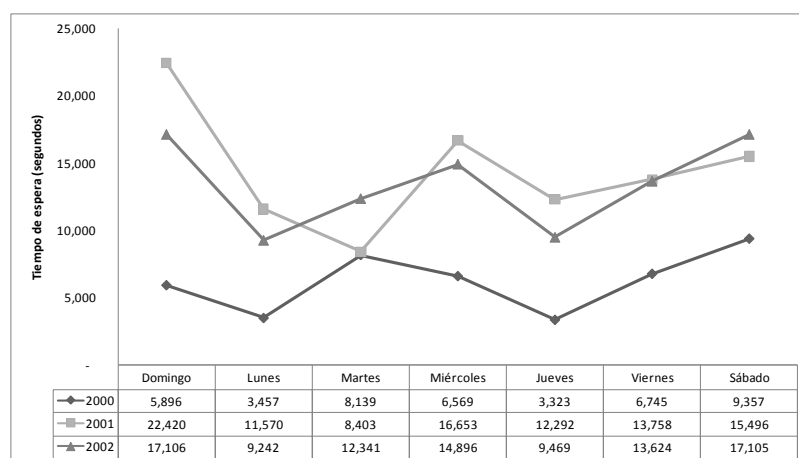
Esto indica que se está presentado un problema en cuanto al manejo de los recursos de infraestructura del grid, pues -a pesar de que se atiende menos tareas- el hardware que se posee no tiene la capacidad suficiente para atender de forma adecuada las solicitudes de los clientes.

Cuadro N° 43 OSC Linux Cluster Tiempo de espera por día de la semana

Día de la semana	2000	2001	2002
Domingo	5,896	22,420	17,106
Lunes	3,457	11,570	9,242
Martes	8,139	8,403	12,341
Miércoles	6,569	16,653	14,896
Jueves	3,323	12,292	9,469
Viernes	6,745	13,758	13,624
Sábado	9,357	15,496	17,105
Promedio	6,212	14,370	13,397

Fuente: Elaboración propia con datos de las bitácoras de OSC Linux Cluster.

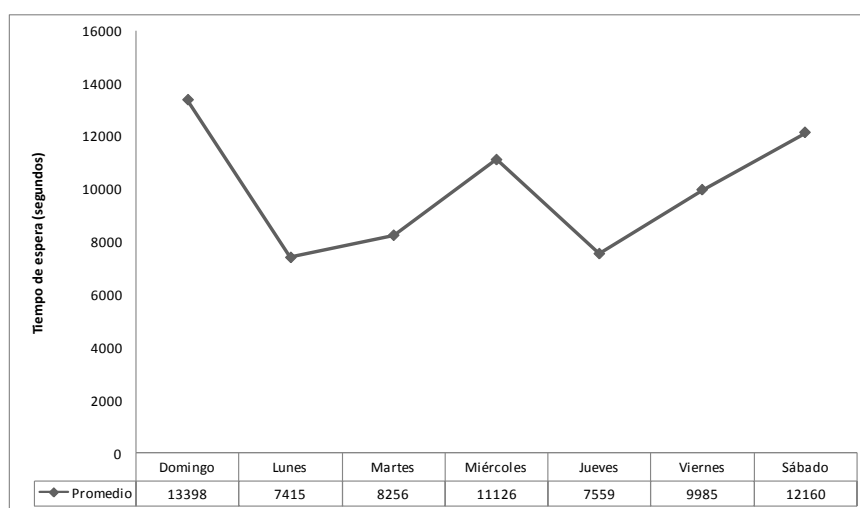
El fenómeno de aumento de tiempo de espera se muestra en el Gráfico N° 36, en el cual se compara este indicador para el año 2000 y 2001. En estas gráficas, se muestra cómo el comportamiento de la atención cambió; porque en el 2000, el tiempo de espera era sumamente inferior. Este comportamiento puede originarse en que las tareas enviadas al grid para su procesamiento son más complejas; por tanto, la infraestructura debería haberse ajustado a estos nuevos requerimientos.

Gráfico N° 36 OSC Linux Cluster Tiempo de espera por día de la semana

Fuente: Elaboración propia con datos de las bitácoras de OSC Linux Cluster.

El Gráfico N° 37 muestra el promedio de atención de los dos años. Los administradores y la administración del grid debe poner especial interés en el estudio de las tareas que son enviadas los días domingo, miércoles y sábado -de forma tal- que se pueda bajar el tiempo de espera en esos días.

Gráfico N° 37 OSC Linux Cluster Tiempo promedio de espera por día de la semana



Fuente: Elaboración propia con datos de las bitácoras de OSC Linux Cluster.

c) Indicador de tiempo de ejecución de las tareas por mes

El sistema OSC Linux en el año 2001 presenta un gran aumento en el tiempo de ejecución de las tareas, lo cual se puede ver en el Cuadro N° 44 y Gráfico N° 38. En este año, el tiempo de ejecución pasó de un promedio anual de 13,436 segundos a 42,378 segundos. Esto representa un aumento del 215%, a pesar de que las tareas pasaron de 1,618 en el año 2000 a 1,390 en el año 2001. Este aumento en la cantidad de segundos necesarios para procesar las tareas le resta eficiencia al grid en general, lo cual impacta directamente en la percepción de calidad de servicio.

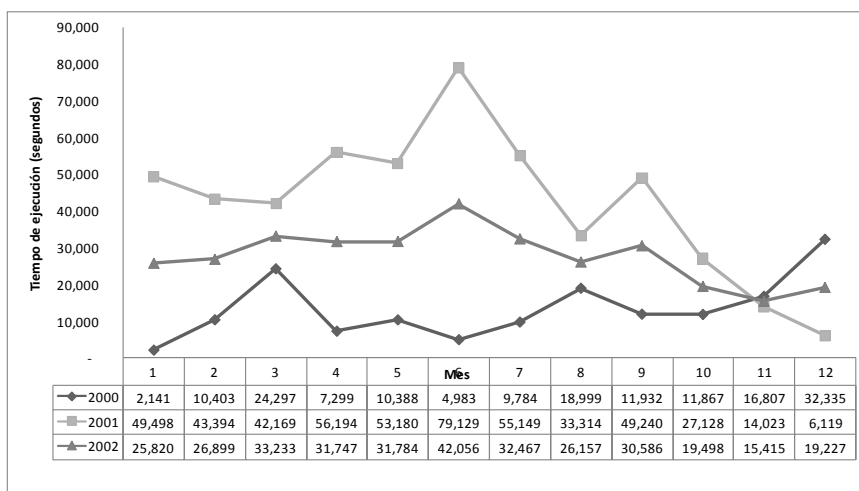
Este acrecentamiento tan pronunciado en el tiempo de procesamiento de las tareas podría deberse a dos causas posibles: primero, las tareas recibidas en el año 2000 son más complejas, por lo tanto se requiere mayor capacidad de procesamiento; segundo, la infraestructura del grid ha sufrido pérdidas de eficiencia ya sea por falta de recursos o por su mal uso. En cualquiera de estos casos es responsabilidad de la administración del grid, encontrar la causa y determinar si se tiene que actualizar la infraestructura, si los algoritmos de calendización deben ser optimizados o ambos.

Cuadro N° 44 OSC Linux Cluster tiempo de ejecución de las tareas

Mes	2000	2001	2002
Enero	2,141	49,498	25,820
Febrero	10,403	43,394	26,899
Marzo	24,297	42,169	33,233
Abril	7,299	56,194	31,747
Mayo	10,388	53,180	31,784
Junio	4,983	79,129	42,056
Julio	9,784	55,149	32,467
Agosto	18,999	33,314	26,157
Setiembre	11,932	49,240	30,586
Octubre	11,867	27,128	19,498
Noviembre	16,807	14,023	15,415
Diciembre	32,335	6,119	19,227
Promedio	13,436	42,378	27,907

Fuente: Elaboración propia con datos de las bitácoras de OSC Linux

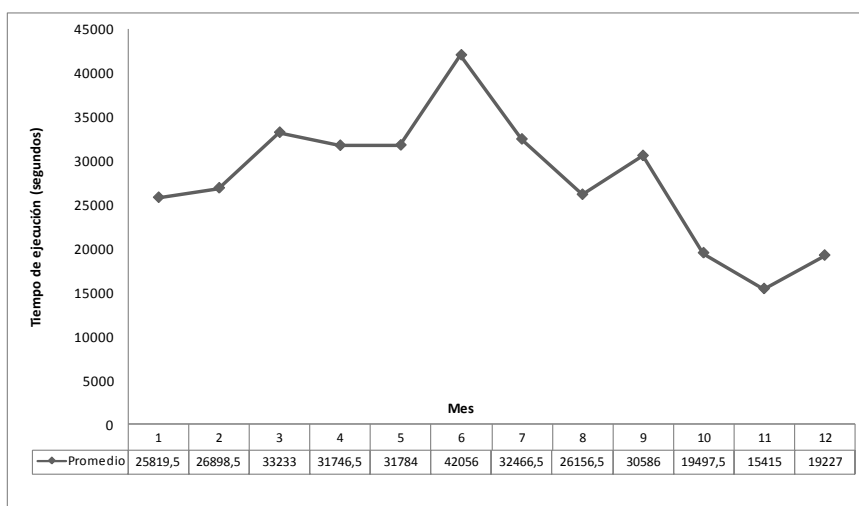
Gráfico N° 38 OSC Linux Cluster tiempo de las tareas dentro del grid



Fuente: Elaboración propia con datos de las bitácoras de OSC Linux Cluster.

Un aspecto importante es que el OSC Linux presenta los tiempos de ejecución más altos en el mes de junio y julio (Ver Gráfico N° 39). Igual que los casos anteriores, es necesario analizar las causas de este comportamiento y tomar las medidas correctivas adecuadas, de forma tal, que los tiempos de ejecución no sean tan altos.

Gráfico N° 39 OSC Linux Cluster. Tiempo promedio de duración de las tareas dentro del grid



Fuente: Elaboración propia con datos de las bitácoras de OSC Linux Cluster.

d) Indicador cantidad de trabajos atendidos por mes e Indicador de memoria utilizada por mes

Las bitácoras de trabajos del sistema OSC Linux Cluster no registran la cantidad de memoria utilizada por las tareas, por lo tanto, no se realiza el análisis de dicho indicador y esta sección se concentra en analizar la cantidad de trabajos atendidos por mes.

El sistema OSC Linux Cluster, en el año 2000, presentó mayor actividad en el periodo que se extiende de abril a julio, tal y como se observa en el Cuadro N° 45. Este comportamiento no se mantuvo puesto que en el año 2001, los meses de mayor actividad fueron agosto y octubre.

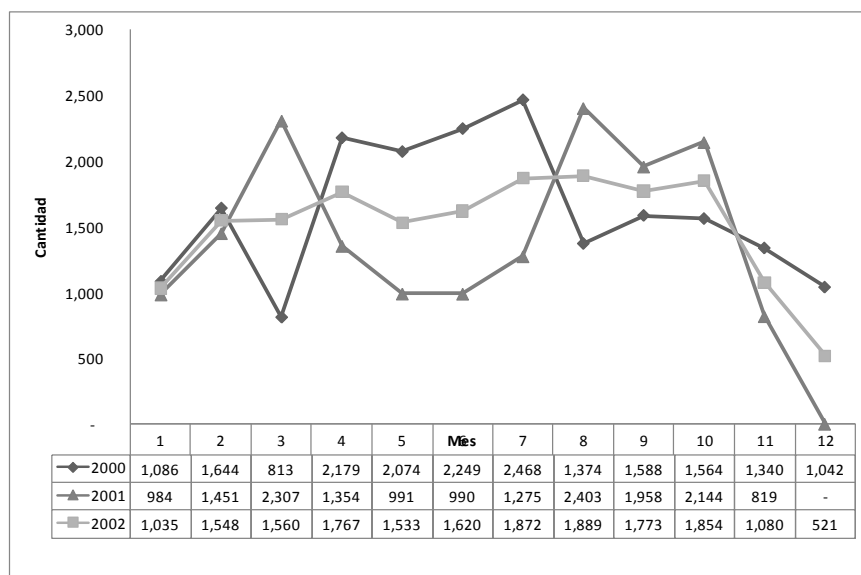
Cuadro N° 45 OSC Linux Cluster Tareas atendidas por mes

Mes	2000 Tareas	2001 Tareas	2002 Tareas
Enero	1,086	984	1,035
Febrero	1,644	1,451	1,548
Marzo	813	2,307	1,560
Abril	2,179	1,354	1,767
Mayo	2,074	991	1,533
Junio	2,249	990	1,620
Julio	2,468	1,275	1,872
Agosto	1,374	2,403	1,889
Setiembre	1,588	1,958	1,773
Octubre	1,564	2,144	1,854
Noviembre	1,340	819	1,080
Diciembre	1,042	-	521
Promedio	1,618	1,390	1,504

Fuente: Elaboración propia con datos de las bitácoras de OSC Linux Cluster.

Tal y como se observa en el Gráfico N° 40, los periodos de más uso del grid han cambiado de un año al otro; esto hace más difícil las tareas de administración, pues se trata de una tarea compleja el predecir cuáles serán los meses de mayor actividad y, así, preparar el grid para que procese las cargas de trabajo de forma eficiente.

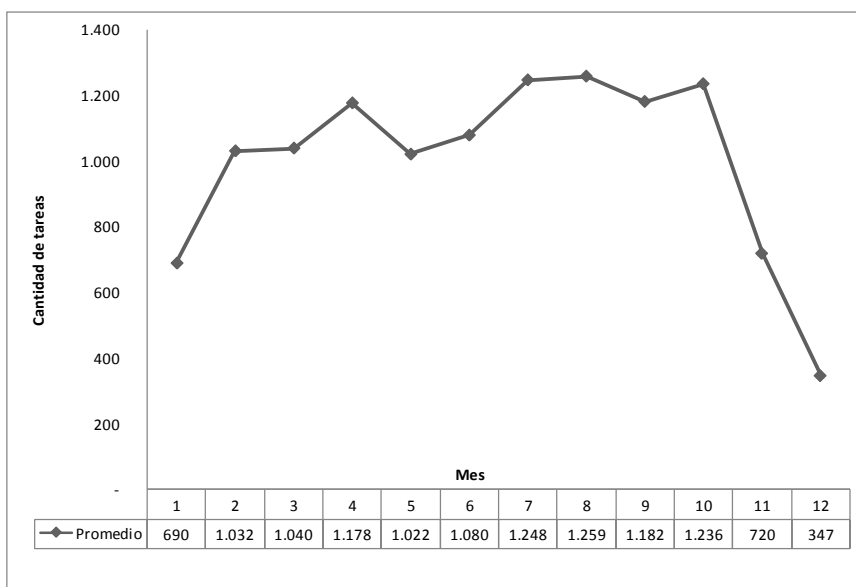
Gráfico N° 40 OSC Linux Cluster Cantidad de tareas atendidas por mes



Fuente: Elaboración propia con datos de las bitácoras de OSC Linux Cluster.

Si se utiliza un promedio de las cargas de trabajo atendidas en los años revisados, se obtienen los resultados que se muestran en el Gráfico N° 41. De este análisis se desprende que, en la mayor parte del año, el grid se mantiene ocupado atendiendo tareas de procesamiento. Sin embargo, en los meses de noviembre y diciembre se presenta una gran baja en la cantidad de trabajos recibidos. Este comportamiento merece que la administración tome medidas necesarias para aumentar la cantidad de trabajos recibidos en el periodo señalado. Una estrategia por seguir consiste en bajar el precio durante estas épocas con el fin de atraer más clientes.

Gráfico N° 41 OSC Linux Cluster Cantidad promedio de tareas atendidas por mes



Fuente: Elaboración propia con datos de las bitácoras de OSC Linux Cluster.

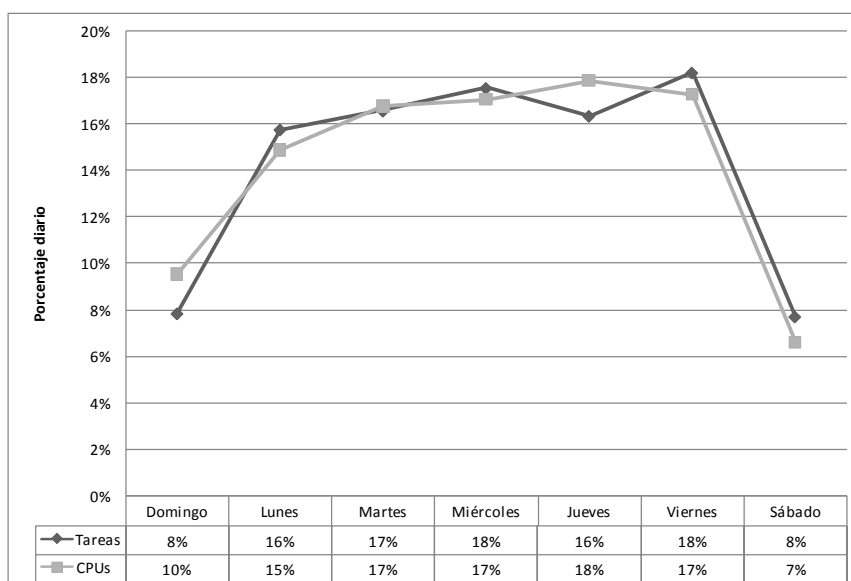
e) Indicador cantidad de trabajos recibidos por día

Los datos sobre la cantidad de trabajos atendidos por día en el sistema OSC Linux Cluster, se muestran en el Cuadro N° 46. Este sistema presenta un comportamiento similar al sistema SDSC Blue, pues durante los fines de semana se presenta menor actividad dentro del grid: se reciben, únicamente, 8% en cada uno de estos días. Este comportamiento se muestra en el Gráfico N° 42.

Cuadro N° 46 OSC Linux Cluster Cantidad de trabajos atendidos por día

Día	2000 Tareas	2001 Tareas	2002 Tareas
Domingo	1,544	1,284	1,414
Lunes	2,914	2,776	2,845
Martes	3,327	2,656	2,992
Miércoles	3,473	2,864	3,169
Jueves	3,115	2,788	2,952
Viernes	3,538	3,039	3,289
Sábado	1,510	1,269	1,390
Promedio	2,774	2,382	2,578

Fuente: Elaboración propia, con datos de las bitácoras de OSC Linux Cluster.

Gráfico N° 42 OSC Linux Cluster Porcentaje de tareas recibidas por día y procesadores utilizados por día

Fuente: Elaboración propia, con datos de las bitácoras de OSC Linux Cluster.

f) Indicador cantidad de trabajos atendidos según hora del día

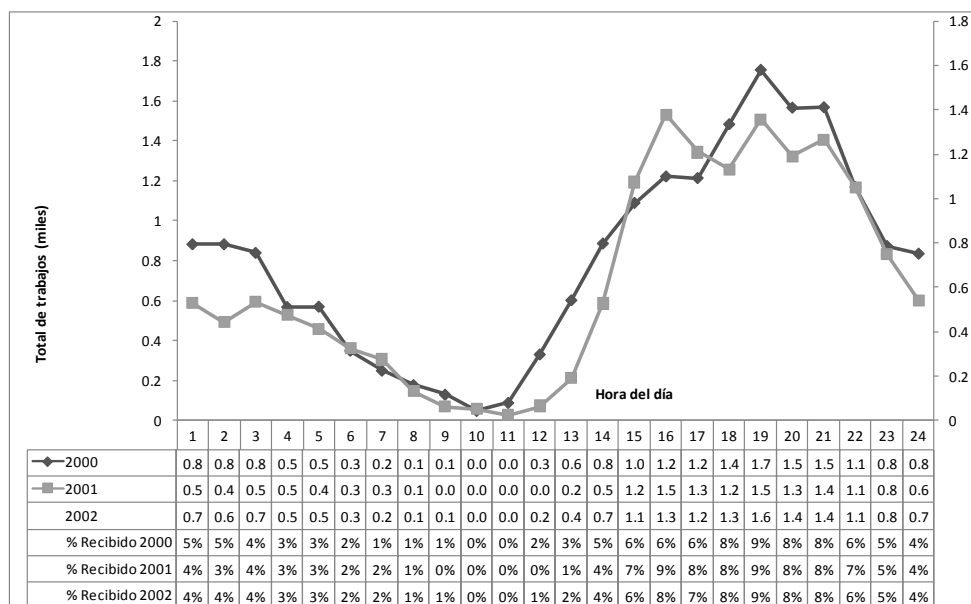
Al igual que en el sistema SDSC Blue, el sistema OSC Linux Cluster (Ver Cuadro N° 47) presenta mayor actividad en el periodo comprendido entre las doce medio día y las doce media noche, ya que se reciben el 75% de las tareas durante ese periodo. En los Gráfico N° 43 y Gráfico N° 44, se puede observar esta tendencia.

Cuadro N° 47 OSC Linux Cluster Cantidad de trabajos atendidos según hora del día

Hora del día	2000	2001	2002
-	884	589	737
1	884	493	689
2	841	594	718
3	570	528	549
4	572	459	516
5	352	360	356
6	252	305	279
7	180	146	163
8	132	71	102
9	51	58	55
10	91	28	60
11	333	72	203
12	604	214	409
13	888	586	737
14	1,091	1,195	1,143
15	1,223	1,532	1,378
16	1,215	1,345	1,280
17	1,484	1,260	1,372
18	1,756	1,507	1,632
19	1,567	1,325	1,446
20	1,569	1,406	1,488
21	1,170	1,169	1,170
22	875	832	854
23	837	602	720

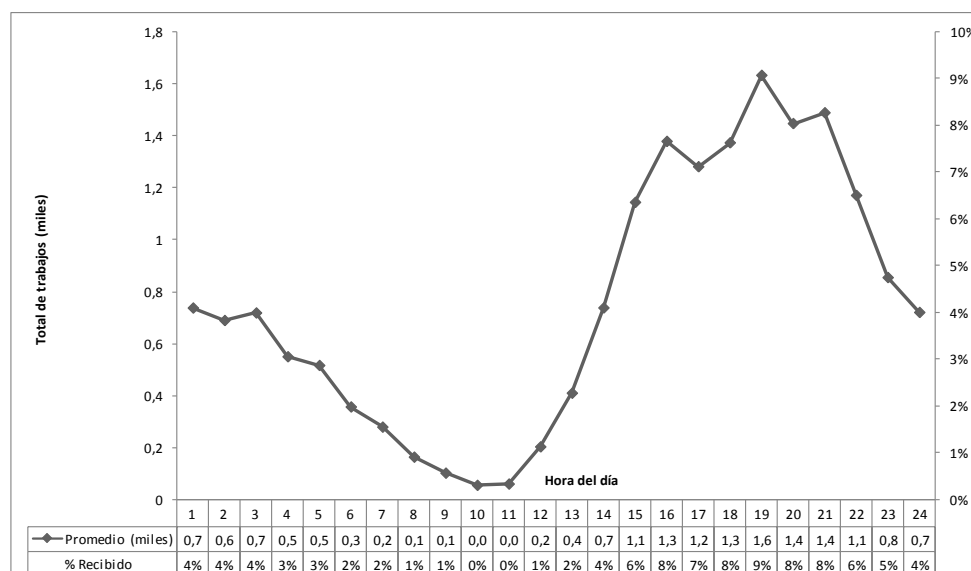
Fuente: Elaboración propia con datos de las bitácoras de OSC Linux Cluster.

Gráfico N° 43 OSC Linux Cluster Cantidad de trabajos recibidos por hora del día



Fuente: Elaboración propia con datos de las bitácoras de OSC Linux Cluster.

Gráfico N° 44 OSC Linux Cluster Cantidad promedio de trabajos recibidos por hora del día



Fuente: Elaboración propia con datos de las bitácoras de OSC Linux Cluster.

En esta sección, se han estudiado distintos indicadores que permiten determinar cuál es la eficiencia operacional de distintos sistemas de informática distribuida. Con esta información, los administradores pueden tomar las medidas necesarias para mejorar la eficiencia y de esa forma aumentar las ganancias obtenidas gracias al procesamiento de cargas de información.

4. Cálculo del Índice de Gestión Operacional (IGO)

En este apartado, se realiza el cálculo del Índice de Gestión Operacional con base en los indicadores operacionales estudiados hasta el momento. Para realizar el cálculo de este índice, se utiliza el procedimiento especificado en el apartado de Análisis de Indicadores del Capítulo de Metodología.

En el Cuadro N° 48, se muestra el procedimiento de cálculo del Índice de Gestión Operacional. La información está compuesta de la siguiente forma:

- En la columna “Resultado del Indicador”, para cada indicador, se muestra el resultado obtenido por SDSC Blue y OSC Linux en cada uno de los años de estudio. Para ver el detalle de cada uno de estos indicadores refiérase a la sección anterior.
- La columna Max se utiliza como medio de normalización de los resultados de los indicadores. La normalización es necesaria debido a que los indicadores poseen unidades de medida distintos y para calcular el IGE es necesario que posean una base común.
- La columna “Resultado del Indicador Normalizado” incluye el resultado de la normalización del indicador con base en valor de la columna Max.
- IGO Bruto corresponde a la suma de los valores normalizados de cada uno de los indicadores. Debido a que se asignó un peso Max de 10, el valor máximo que puede alcanzar el IGO Bruto es de 60 puntos.

- IGO Porcentual corresponde al porcentaje del total de puntos obtenidos por un sistema de grid computing en un año dado.
- IGO Normalizado corresponde a la normalización de los puntos obtenidos con base en el peso asignado al IGO dentro del cálculo del IBECOD. Para efectos de cálculo, en este estudio se le asignó al IGO un peso de 33% del total de IBECOD, sin embargo, este parámetro podría variarse dependiendo de la importancia que se le desee asignar a cada componente del IBECOD.

Cuadro N° 48 Cálculo del Índice de Gestión Operacional

Indicador	Unidad Medida	Resultado del Indicador						Max	Resultado del Indicador Normalizado					
		SDSC Blue			OSC Linux				SDSC Blue			OSC Linux		
		2000	2001	2002	2000	2001	2002		2000	2001	2002	2000	2001	2002
Indicador de tiempo de espera por mes (segundos)	Seg	32,207.00	25,283.00	48,125.00	5,481.00	11,696.00	8,933.00	10.00	1.70	4.63	1.86	10.00	10.00	10.00
Indicador tiempo de espera por día de la semana (segundos)	Seg	19,659.00	26,216.00	49,078.00	6,212.00	14,370.00	13,397.00	10.00	3.16	5.48	2.73	10.00	10.00	10.00
Indicador de tiempo de ejecución de las tareas por mes (segundos)	Seg	4,009.00	4,236.00	4,299.00	13,436.00	42,378.00	27,907.00	10.00	2.98	1.00	1.54	10.00	10.00	10.00
Indicador cantidad de trabajos atendidos por mes (EPs)	Eps	4,087.00	8,108.00	8,082.00	1,618.00	1,390.00	1,504.00	10.00	10.00	10.00	10.00	3.96	1.71	1.86
Indicador de memoria utilizada por mes (Megabytes)	MB	9,405.00	10,665.00	10,426.00	2,346.10	2,015.50	2,180.80	10.00	2.49	1.93	2.09	10.00	10.00	10.00
Indicador cantidad de trabajos atendidos según hora del día (EP/Hora)	Cant	7,006.00	13,899.00	13,854.00	2,774.00	2,382.00	2,578.00	10.00	10.00	10.00	10.00	3.96	1.71	1.86
IGO Bruto								70.00	40.34	43.04	38.22	51.88	45.14	45.58
IGO Porcentual								100.00	0.58	0.61	0.55	0.74	0.64	0.65
IGO Normalizado								33.33	19.21	20.49	18.20	24.70	21.49	21.70

Fuente: Elaboración Propia

En el Cuadro N° 48, se muestra el resultado del cálculo del IGO Normalizado. Para el año 2000, el sistema SDSC Blue obtuvo un IGO de 19,21 mientras que OSC Linux presenta un IGO de 24,70. Por tanto, durante ese año la gestión operacional de OSC Linux fue superior. Lo anterior es consistente con el resultado de los indicadores individuales. En estos se aprecia que OSC Linux posee menores tiempos de espera. Este comportamiento se mantiene durante el resto de los años. OSC Linux posee un IGO superior, lo cual indica que, según el modelo planteado en esta investigación, la gestión operacional de SDSC Blue debe mejorarse y para realizarlo es preciso estudiar cada uno de los indicadores individualmente y tomar las medidas necesarias para perfeccionarlos. De manera específica, se debe

investigar por qué los tiempos de espera son tan elevados en relación con la cantidad de trabajos atendidos.

D. Índice de Eficiencia en Infraestructura (IEI)

El índice de eficiencia en infraestructura y manejo de recursos será investigado con respecto a los efectos operacionales tales como cantidad de datos manipulados en el grid y los tiempos de transmisión de los datos que deben ser procesados, entre otros. Estos parámetros son de gran importancia debido a que influyen en la calidad del servicio prestado y en los costos- tanto de los dueños de la información, que debe ser procesada, como de los administradores del grid.

El análisis instrumental de estos índices implica incluir otros factores como cantidad de carga que debe ser transportada para procesarse, capacidad de la red de transporte de datos, entre otros. La relación de estos parámetros ayudan a definir un modelo matemático que explica la importancia de la eficiencia en la administración de la infraestructura de los recursos computacionales.

1. Eficiencia de procesamiento y la importancia del tiempo de estadía de la información en los nodos del grid

Desde el punto de vista de costos y rendimiento, es lógico pensar que el aumento de la velocidad de procesamiento de datos en los grids incide directamente en el índice de eficiencia IBECOD.

Según Santibáñez (2004): “las mayores dificultades que se le presentan al investigador, son encontrar ciertas relaciones, que reducidas a una fórmula o ecuación, permitan configurar un cierto modelo matemático, que nos permita la aplicación de algoritmos u otras

herramientas que entreguen certezas y resultados. Además, debemos demostrar de manera rigurosa y exacta, que un aumento en la velocidad de manipulación de carga, se traduce en un beneficio, expresable en cualquier unidad adecuada: toneladas adicionales de carga (para carga a granel), TEUS (para carga contenedorizada), porcentaje de rentabilidad (sobre las inversiones) o simplemente, utilidades de operación.”

El planteamiento original de Santibáñez se aplica al transporte de carga, pero puede ser perfectamente adaptado al transporte y procesamiento de paquetes de información: caso que nos interesa en esta investigación.

Para realizar el análisis presentado en esta investigación, se recurre a bases matemáticas que permiten diseñar relaciones o ecuaciones, las cuales sirven para modelar el problema de estudio y brinda el punto de partida de lo que se desea proponer, medir, evaluar e investigar. En esta sección, se desarrolla un procedimiento que inicia con varias restricciones con el fin de establecer un modelo general, para que se pueda entender la dinámica del movimiento de información en la informática distribuida y, de esa forma, asignar una medida a la eficiencia de los proveedores de servicios.

El modelo que se mostrará fue creado, inicialmente, en el ámbito del transporte y puertos (Santibáñez, 2004), y ha sido adaptado y extendido para medir la eficiencia en infraestructura de los nodos de procesamiento de grid computing.

2. Modelo para definir la eficiencia de los algoritmos de calendarización en informática distribuida

En esta sección, se analiza la aplicación del modelo definido en el Capítulo VI con el fin de medir la eficiencia en infraestructura (IEI) de los algoritmos de calendarización en la

informática distribuida. Cabe destacar que el IEI es calculado utilizando la Ecuación 11, cuyos parámetros y definición se muestran a continuación.

Tabla N° 21 Parámetros del modelo de medición de eficiencia en infraestructura

Siglas Utilizadas	Significado de la Variable del Modelo Matemático
NV	Número de viajes necesarios para transportar la información
K	Cantidad de información por transportar.
Mbp	Cantidad de información que puede manejar la red de transmisión de datos.
Nvp	Número de viajes en periodo de actividad (ida y retorno).
Kp	Cantidad de información transportable en un periodo de actividad y en ambos sentidos.
Tv	Tiempo que es ocupada la red en un viaje completo de información
Tpg	Tiempo empleado por los paquetes de información en su operación dentro del grid.
Tng	Tiempo empleado por la información desde su salida hasta recibir los resultados del procesamiento.
Tg	Tiempo de actividad de los nodos del grid
Kmg	Cantidad de información manejada en el grid (carga de información y generación de resultados).
Vp	Velocidad de procesamiento de la información dentro del grid.
QoS	Nivel de calidad de servicio.

Fuente: Elaboración Propia.

$$Kp = \left(\frac{2Mbp * Tg}{\frac{Kmg}{Vp} + Tng} \right) * QoS$$

Ecuación 11 Medición de la eficiencia de procesamiento de información

Esta última ecuación permite conocer la cantidad de información, transportada y procesada, durante un periodo de actividad determinada, con base en las variaciones que se den en la velocidad de manipulación de información. Por tanto, si los algoritmos de distribución de carga son eficientes, el valor de Kmg (Cantidad de información manejada en el grid) aumentará. En esta investigación, se analizará cómo cambia este parámetro dependiendo de la cantidad de información procesada en un tiempo dado utilizando un algoritmo de distribución determinado.

Antes de aplicar el modelo de cálculo de IEI en esta investigación, se realiza un análisis sobre el comportamiento del modelo en distintas situaciones. Este se presenta en la siguiente sección.

a) Análisis de sensibilización del modelo de cálculo de IEI

Para sensibilizar el efecto que existe entre los tiempos de espera dentro del grid (que afecta al índice de básico de eficiencia de computación distribuida - IBECOD) y la velocidad de procesamiento de cargas de información y con el fin de diseñar una red de transmisión de datos óptima en el transporte de cargas de información, esta investigación demuestra- mediante una simulación- el efecto que tiene la eficiencia en el procesamiento de cargas de información sobre el desempeño global de las empresas que brindan servicios de procesamiento de datos.

Para fines de la simulación, se ha considerado cargas de información constantes y se variará la capacidad de procesamiento que tienen sus nodos de procesamiento. En el Cuadro N° 49, se muestran los parámetros utilizados durante la simulación.

Cuadro N° 49 Información base del modelo

Variables de operación	Siglas	Información de cada variable
Cantidad de información que puede manejar la red de transmisión de datos.	Mbp	1,024 Megabits
Tiempo de actividad de los nodos del grid.	Tg	8,760 horas anuales (7x24)
Cantidad de información manejada en el grid (carga de información y generación de resultados).	Kmg	512 Megabits
Tiempo empleado por la información desde su salida hasta recibir los resultados del procesamiento.	Tng	4 horas en promedio
Costo mantener nodo.		1,000 US\$
Ganancia por instrucción procesada ciclos CPU.		0.0004 US\$

Fuente: Elaboración propia.

El propósito de la simulación es investigar la forma cómo se comportarán las variables de medición conforme se aumente o disminuya la eficiencia del grid de procesamiento de datos; además, de la cantidad de información que puede procesarse dentro del grid

conforme varía la velocidad promedio de procesamiento de datos dentro de los nodos del grid.

Para efectuar un análisis paramétrico, se considera una variación en la velocidad de procesamiento de carga que va desde 10 EPs (Entidades de Procesamiento) hasta 1500 EPs por segundo.

Desarrollo

Se utiliza el desarrollo de la Ecuación 11 Índice de eficiencia del grid computing , y en esta simulación se asume un valor de calidad de servicio igual a uno.

$$Kp = \left(\frac{\frac{2Mbp * Tg}{Kmg} + Tng}{Vp} \right) * QoS$$

Ecuación 11 Índice de eficiencia del grid computing

Cuando se evalúa la ecuación con los datos conocidos para la simulación, queda como sigue:

$$Kp = \left(\frac{2 \bullet (1,024) * 8,760}{\frac{512}{Vp} + 4} \right) * 1 \text{ EPs por segundo}$$

$$Kp = \left(\frac{17,940,480.00}{\frac{512}{Vp} + 4} \right) * 1 \text{ EPs por segundo}$$

Esta información puede ser presentada matricialmente (forma tabular) para el análisis, de forma tal que facilite la comprensión y nos proporciona las siguientes cifras.

Cuadro N° 50 Simulación 1: Velocidad de manipulación de cargas de datos variable

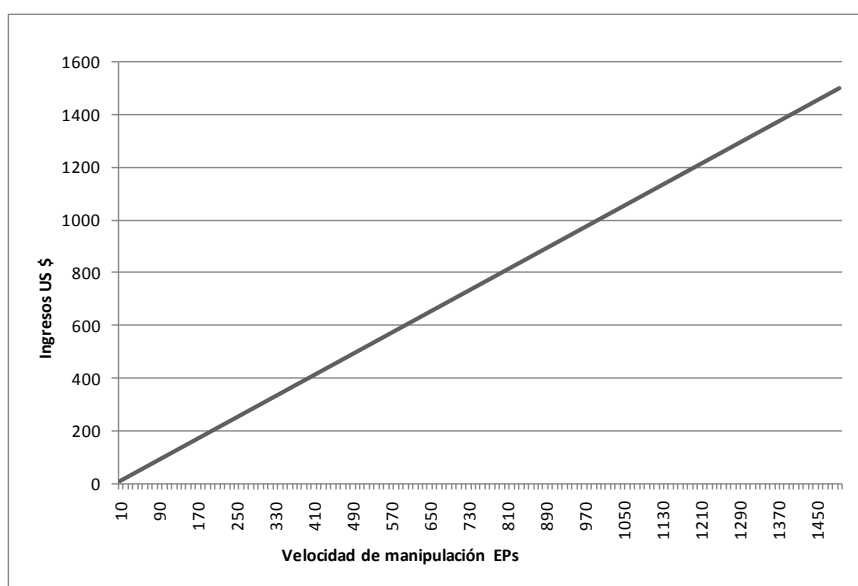
Velocidad de manipulación de datos V_p	Capacidad de la red $2 * Mbp * T_g$	Tiempo procesamiento $(Kmg/V_p) + T_n$ g	Total de Información procesada K_p	Ciclos de procesamiento	Aumento ⁴⁹
10	17,940,480.00	55.20	325,008.70	2.48	0
20	17,940,480.00	29.60	606,097.30	4.63	2.15
30	17,940,480.00	21.07	851,605.06	6.50	1.87
40	17,940,480.00	16.80	1,067,885.71	8.15	1.65
50	17,940,480.00	14.24	1,259,865.17	9.61	1.47
60	17,940,480.00	12.53	1,431,421.28	10.92	1.31
70	17,940,480.00	11.31	1,585,648.48	12.10	1.18
80	17,940,480.00	10.40	1,725,046.15	13.17	1.06
90	17,940,480.00	9.69	1,851,655.05	14.13	0.97
100	17,940,480.00	9.12	1,967,157.89	15.01	0.88
200	17,940,480.00	6.56	2,734,829.27	20.87	0.42
300	17,940,480.00	5.71	3,143,775.70	23.99	0.24
400	17,940,480.00	5.28	3,397,818.18	25.93	0.16
500	17,940,480.00	5.02	3,570,955.41	27.25	0.11
1000	17,940,480.00	4.51	3,976,170.21	30.35	0.03
1100	17,940,480.00	4.47	4,017,615.64	30.66	0.03
1200	17,940,480.00	4.43	4,052,819.28	30.93	0.03
1300	17,940,480.00	4.39	4,083,092.44	31.16	0.02
1400	17,940,480.00	4.37	4,109,403.14	31.36	0.02
1500	17,940,480.00	4.34	4,132,481.57	31.54	0.02

Fuente: Elaboración propia, con base en simulaciones realizadas.

⁴⁹ El aumento se calcula como la diferencia entre los ciclos de procesamiento actuales (n) y los ciclos de procesamiento inmediatamente anterior ($n-1$)

Puede observarse, la gran influencia que ejerce la velocidad de manipulación de datos V_p , sobre la cantidad de información que es procesada dentro del grid. Basta que aumente la velocidad de procesamiento de información V_p de 10 a 40 para que el total de información procesable por el grid suba de 325,008.70 EPs a 1, 067,885.71 EPs. Con esto, si se supone una tarifa de \$0.0004⁵⁰, representaría un aumento en los ingresos del grid de \$2,971.51 por segundo. En el Gráfico N° 45, se muestra el comportamiento de los ingresos con relación al aumento de eficiencia en el procesamiento de información. Esta relación es de tipo exponencial; sin embargo, esto ocurriría únicamente en un ambiente ideal y óptimo, lo cual es muy distinto a la realidad.

Gráfico N° 45 Comportamiento de los ingresos relacionados con el aumento en la eficiencia de procesamiento de datos



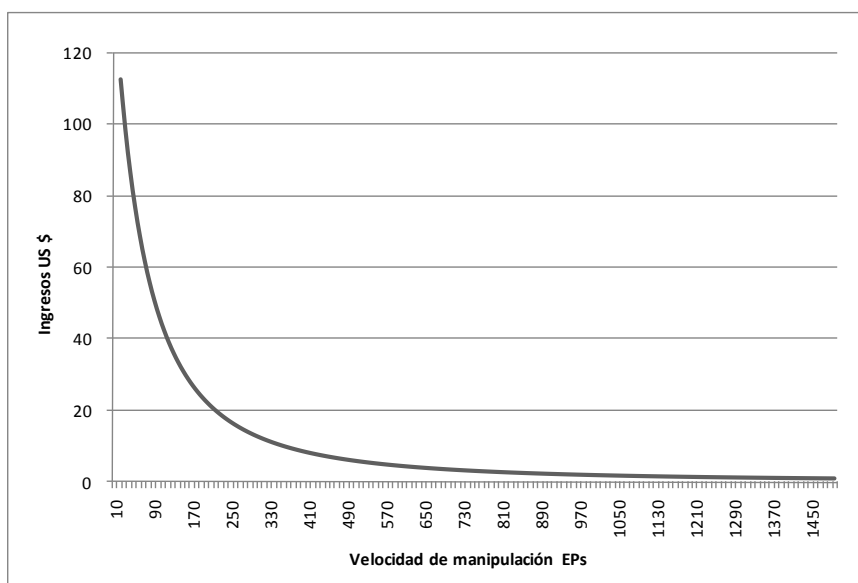
Fuente: Elaboración propia, con base en simulaciones realizadas.

Como se ha mencionado anteriormente, el aumento en los ingresos -producto de un acrecentamiento en la eficiencia de manipulación de datos- no crece de forma exponencial,

⁵⁰ Tipo de moneda utilizada: dólares de Estados Unidos de América

tal y como se aprecia en el Gráfico N° 46. El aumento en la eficiencia, al principio, genera grandes crecimientos en los ingresos. Sin embargo, llega el momento cuando los aumentos de velocidad de procesamiento no generan grandes cambios en los ingresos percibidos. De esta forma, es necesario que las empresas proveedoras de servicios de informática distribuida, realicen análisis de forma tal que se encuentre el punto de equilibrio entre aumento de eficiencia y aumento de los ingresos. Esto se debe a que todo aumento en eficiencia tiene un costo relacionado, el cual -no necesariamente- contribuirá a un incremento en los ingresos percibidos.

Gráfico N° 46 Relación entre los ingresos y aumento de procesamiento de información



Fuente: Elaboración propia, con base en simulaciones realizadas.

3. Análisis de sensibilidad con cargas de información ilimitada.

En el análisis de sensibilidad presentado anteriormente, se ha demostrado la influencia de la velocidad de procesamiento de datos, V_p , en la cantidad de información que puede ser enviada por los clientes hacia los proveedores de servicios. Sin embargo, en este caso se ha considerado que la cantidad de información que necesita procesarse es ilimitada o, más bien,

que los proveedores de servicios de procesamiento cuentan siempre con clientes dispuestos a enviar sus datos para que sean procesados.

La existencia de una cantidad ilimitada de datos por procesar se presenta únicamente en casos especiales, donde se cuenta con una fuente de datos ilimitada. Un ejemplo de estos es el proyecto SETI que está procesando las señales provenientes del espacio exterior, por lo tanto, tiene una fuente de datos ilimitada.

No obstante que la disponibilidad de carga es limitada, los proveedores de servicios de informática distribuida deben contar siempre con el equipo computacional listo para operar en caso de que se presenten picos de demanda de procesamiento. El estudio del impacto que se tiene en los costos debido a poseer equipo ocioso, se deja como un problema abierto, el cual podría incluirse en trabajos futuros.

4. Diseño de un conjunto de nodos de procesamiento y su relación con la velocidad de procesamiento y administración de carga

Con el aumento en la velocidad de procesamiento y administración de carga, pueden optimizarse los costos de operación de un proveedor de informática distribuida, pues se eliminan nodos de procesamiento, lo cual no modifica la cantidad de cargas de información que puede estar atendándose pero ahora con una menor cantidad de nodos y, por consiguiente, con menores costos de operación.

Cuando se desarrolla la expresión K_p , cantidad de información procesable, y se extiende al uso de tres, cuatro, cinco y seis nodos de procesamiento, se obtiene el siguiente cuadro comparativo.

Cuadro N° 51 Simulación: Comparación de información procesada

Velocidad de manipulación	Información procesada por 3 nodos	Información procesada por 4 nodos	Información procesada por 5 nodos	Información procesada por 6 nodos
80	5,175,138	6,900,185	8,625,231	10,350,277
90	5,554,965	7,406,620	9,258,275	11,109,930
100	5,901,474	7,868,632	9,835,789	11,802,947
110	6,218,864	8,291,818	10,364,773	12,437,728
120	6,510,658	8,680,877	10,851,097	13,021,316
130	6,779,833	9,039,777	11,299,721	13,559,665
140	7,028,919	9,371,893	11,714,866	14,057,839
150	7,260,086	9,680,115	12,100,144	14,520,173
160	7,475,200	9,966,933	12,458,667	14,950,400
170	7,675,877	10,234,502	12,793,128	15,351,753
180	7,863,522	10,484,696	13,105,870	15,727,044
190	8,039,366	10,719,155	13,398,943	16,078,732
380	10,065,033	13,420,044	16,775,055	20,130,066

Fuente: Elaboración propia, con base en simulaciones realizadas.

En el Cuadro N° 51, obsérvense las cantidades de información remarcadas, las cuales representan alrededor de 10,000,000 de Eps de carga de información procesada. De estos datos, se desprende que esta información puede ser atendida por 3, 4, 5 o 6 nodos de procesamiento, según sea la velocidad de manipulación de carga en los grids de procesamiento que atienden a los consumidores finales. En el Cuadro N° 51, se muestra la comparación de este comportamiento.

Cuadro N° 52 Simulación: Relación entre la cantidad de nodos y la velocidad de procesamiento

Cantidad de nodos requeridos para procesar la información "N"	Velocidad de procesamiento de Eps "Vp"	Cantidad de información procesable "Kp"
3 nodos	380	10,065,033
4 nodos	170	10,234,502
5 nodos	110	10,364,773
6 nodos	80	10,350,277

Fuente: Elaboración propia, con base en simulaciones realizadas.

Cuadro N° 53 Simulación: Relación entre la cantidad de nodos y la velocidad de procesamiento

Cantidad de nodos requeridos para procesar la información "N"	Velocidad de procesamiento de Eps "Vp"	Cantidad de información procesable "Kp"
3 nodos	1060	12,005,624.24
4 nodos	260	12,021,971.13
5 nodos	150	12,100,143.88
6 nodos	110	12,437,727.73

Fuente: Elaboración propia, con base en simulaciones realizadas.

En el Cuadro N° 52 y Cuadro N° 53, se aprecia que cuantos más nodos de procesamiento estén disponibles en el grid, menor será la velocidad de procesamiento necesaria para atender la misma cantidad de información. De esta forma, si una empresa -debido a restricciones de software o hardware- no puede aumentar la velocidad de procesamiento

individual de los nodos, entonces podrá optar por agregar más nodos con el fin de procesar cantidades mayores de información.

En el Gráfico N° 47, se observa la forma cómo varía la cantidad de información procesada con respecto a la cantidad de nodos de procesamiento disponibles y su velocidad de procesamiento. Ante el supuesto de procesar 4,000,000 de EPs, las velocidades de manipulación requeridas para atender dicha carga varían según la cantidad de nodos que se posean. El detalle de estos requerimientos se pueden ver en el Cuadro N° 54 y el Gráfico N° 47.

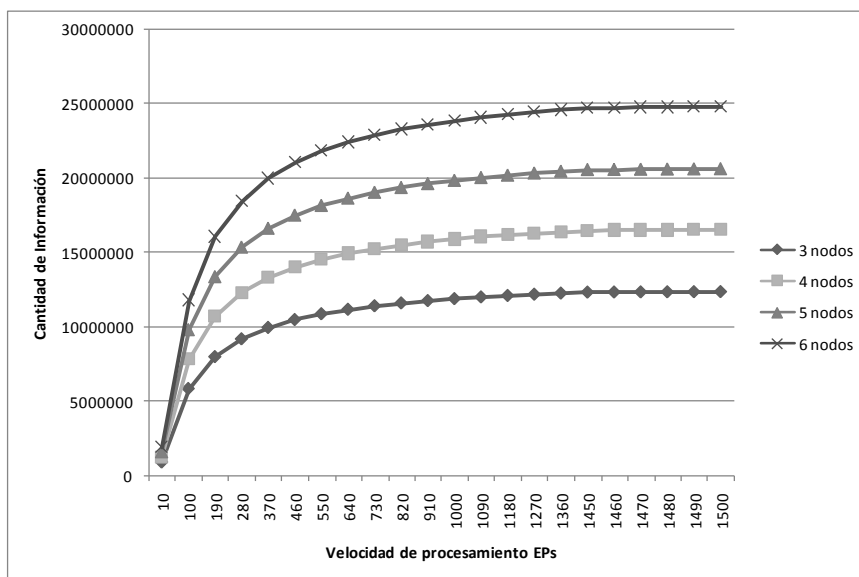
Cuadro N° 54 Resultado de simulación y análisis de cantidad de nodos

Cantidad de nodos requeridos “N”	Velocidad de procesamiento de información “Vp”
6 nodos	20
5 nodos	30
4 nodos	40
3 nodos	60
1 nodo	1060

Fuente: Elaboración propia, con base en simulaciones realizadas.

Si se considera que el costo diario de mantener un nodo de procesamiento compuesto por 128 computadores es de alrededor de \$768, la eliminación de un nodo gracias a un aumento de eficiencia representa un impacto económico importante en ahorros de \$280,320.00 dólares anuales, esto sin contar la inversión inicial realizada para poner en operación un nodo adicional.

Gráfico N° 47 Relación entre cantidad de nodos disponibles y la cantidad de información atendida

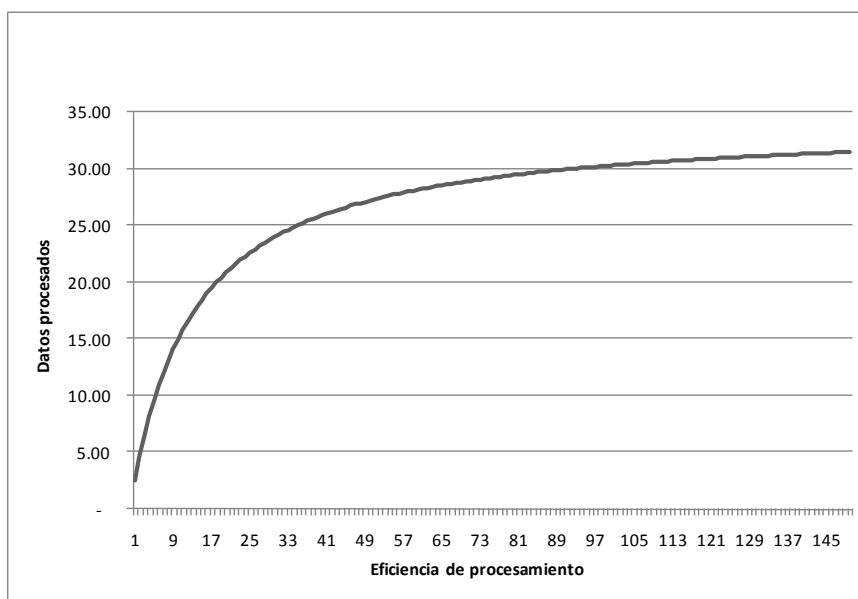


Fuente: Elaboración propia, con base en simulaciones realizadas.

Un análisis ligero de los resultados anteriores podría llegar a concluir que cuanto más se aumente la eficiencia de procesamiento, menos nodos se requerirán para atender la misma cantidad de información. Esto es en parte verdad; sin embargo, tal y como se muestra en el Gráfico N° 48, el aumento eficiencia tiene una asíntota, de forma tal el aumento de eficiencia de los nodos individuales tiene un límite.

Por tanto, cuando una empresa proveedora de informática distribuida llega al punto máximo de eficiencia individual de los nodos, la única manera de aumentar su capacidad de procesamiento global es agregando nuevos nodos de procesamiento al grid.

Gráfico N° 48 Relación entre eficiencia del grid y capacidad de procesamiento

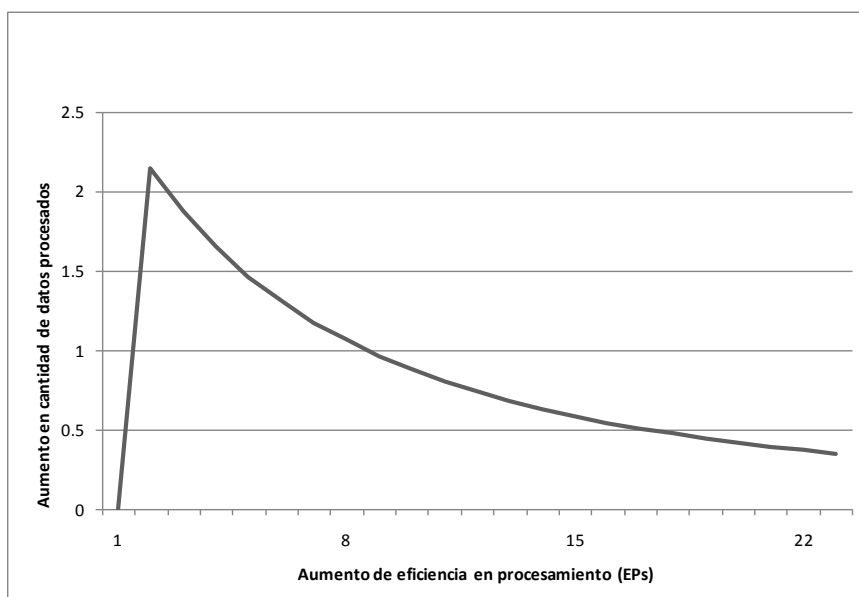


Fuente: Elaboración propia, con base en simulaciones realizadas.

El planteamiento anterior de una asíntota de capacidad se confirma en el Gráfico N° 49, en el cual se observa que inicialmente un pequeño aumento de eficiencia individual del nodo implica un aumento significativo en la cantidad de datos procesados.

Sin embargo, este aumento de eficiencia tiene un límite. En el Gráfico N° 49, se observa que cuando la eficiencia individual es superior a 22, el incremento en la cantidad de datos procesados no es significativo. Por lo tanto, la inversión requerida para aumentar la eficiencia no se recuperará en un plazo razonablemente corto.

Gráfico N° 49 Relación de la eficiencia de procesamiento y la información atendida



Fuente: Elaboración propia, con base en simulaciones realizadas.

5. Análisis adicionales para el índice de eficiencia en infraestructura

Con respecto a las utilidades resultantes, si se aumenta la velocidad de manipulación de las cargas de información atendidas en los nodos de la informática distribuida (V_p), se deduce de este análisis que V_p es un factor muy importante en el contexto de operación y administración.

La tarea de optimizar la velocidad de atención y procesamiento de información en la informática distribuida, es responsabilidad de los dueños del grid y para hacerlo deben tomar en cuenta lo siguiente:

- Se requiere una continua innovación tecnológica, debido a que la infraestructura del grid influye directamente en la obtención de un índice de eficiencia alto y competitivo.
- Es necesario determinar cuáles son los algoritmos de distribución de carga que permitan mejorar el índice de eficiencia y que maximice el uso de los recursos disponibles.
- Si se identifican tendencias de uso de los recursos del grid, será posible prepararse para atender -de la forma más eficiente- a los clientes que solicitan procesamiento de tareas.

Otro factor importante que debe señalarse es el tiempo que una tarea permanece esperando para ser atendida, lo cual afecta directamente la calidad de servicio y provoca retrasos a los consumidores, pues sus operaciones podrían estar paralizadas debido a que están esperando los resultados.

Un aumento en la productividad del grid provoca un alto impacto en la calidad de servicio percibida por los clientes y, además, se aumentan los ingresos de los dueños del grid. Entonces, la evaluación de resultados en los factores presentados por el investigador ponen claro que existe una excelente oportunidad para inducir los cambios necesarios, los cuales logren un aumento en la productividad y la capacidad de los nodos de informática distribuida.

6. Cálculo del Índice de Eficiencia en Infraestructura

Para realizar el cálculo del Índice de Eficiencia en Infraestructura, se aplica la Ecuación 11 definida en este apartado. Los resultados de este índice se muestran en el Cuadro N° 55. En general, el índice de eficiencia en infraestructura nos permite identificar que el sistemas SDSC Blue no realiza un uso eficiente de su infraestructura. Se puede apreciar que los tiempos de espera son muy elevados y -a pesar de que atiende gran cantidad de tareas- no hace uso eficiente de su infraestructura. Este comportamiento se aprecia mejor en el Cuadro N° 56, en donde se observa que SDSC Blue posee muchos más procesadores que OCS Linux y -a pesar de esto- factura menos horas por procesador y tiempos de espera superiores, lo cual implica un uso poco eficiente de la infraestructura instalada.

Cuadro N° 55 Cálculo del Índice de Eficiencia en Infraestructura IEI

Indicador	SDSC Blue			OSC Linux		
	2000	2001	2002	2000	2001	2002
Tiempo de espera anual	32,207.08	25,282.75	48,124.92	5,480.50	12,384.71	8,932.60
Tiempo de espera diario	19,659.00	26,216.00	49,078.00	6,212.00	14,370.00	13,397.00
Tiempo de ejecución	4,009.00	4,236.00	4,299.00	13,436.00	42,378.00	27,907.00
Cantidad de tareas atendidas al año	49,044.00	97,290.00	96,980.00	19,421.00	16,676.00	18,048.50
Índice de eficiencia en infraestructura	6.21	7.90	4.15	36.48	16.15	22.39

Fuente: Elaboración propia, con base en el modelo de eficiencia definido en esta investigación.

Cuadro N° 56 Detalles del cálculo del IEI

Variable	SDSC Blue			OSC Linux		
	2000	2001	2002	2000	2001	2002
Total segundos ocupados	196,617,396.00	412,120,440.00	416,917,020.00	260,940,556.00	706,695,528.00	503,679,489.50
Total horas facturadas	54,615.94	114,477.90	115,810.28	72,483.49	196,304.31	139,910.97
Total procesadores	1,152.00	1,152.00	1,152.00	178.00	178.00	178.00
Horas facturas por procesador	47.41	99.37	100.53	407.21	1,102.83	786.02
Numerador Ecuación 11	200,000.00	200,000.00	200,000.00	200,000.00	200,000.00	200,000.00
Denominador Ecuación 11	32,219.32	25,305.72	48,147.48	5,481.95	12,385.10	8,933.25
Resultado	6.21	7.90	4.15	36.48	16.15	22.39

Fuente: Elaboración propia, con base en el modelo de eficiencia definido en esta investigación.

En el Cuadro N° 57, se muestra el IEI normalizado. Esta normalización es necesaria para calcular IBECOD; para efectos de cálculo al IEI, se le asigna un peso de 33%. Cabe destacar que en la escala normalizada OSC Linux posee un comportamiento superior a OSC Blue.

Cuadro N° 57 Índice de Eficiencia en Infraestructura Normalizado

Indicador	Resultado del Indicador							Resultado del Indicador Normalizado					
	SDSC Blue			OSC Linux			Max	SDSC Blue			OSC Linux		
	2000	2001	2002	2000	2001	2002		2000	2001	2002	2000	2001	2002
IEI Bruto	6.21	7.90	4.15	36.48	16.15	22.39	10.00	1.70	3.53	1.86	10.00	10.00	10.00
IEI Porcentual							100.00	17%	35%	19%	100%	100%	100%
IEI Normalizado							33.33	5.67	11.77	6.18	33.33	33.33	33.33

Fuente: Elaboración propia, con base en el modelo de eficiencia definido en esta investigación.

7. Cálculo del Índice Básico de Eficiencia de Computación Distribuida (IBECOD)

En el capítulo de Metodología, se define IBECOD como a aquel que resume la eficiencia de los grid de computadoras en cuanto al procesamiento de cargas de trabajo y que está compuesto de la siguiente forma:

$$IBECOD = IGE + IEI + IGO + \xi$$

IGE: Índice de Gestión Económica.

IEI: Índice de Eficiencia en Infraestructura.

IGO: Índice de Gestión Operacional.

ξ : Margen de error.

A continuación, se procede al cálculo del IBECOD con base en los resultados de los índices calculados en este apartado. En el Cuadro 58, se observa el detalle del cálculo de IBECOD, el cual consiste en la combinación de los índices normalizados de gestión económica,

gestión operacional y de eficiencia en infraestructura. En este cálculo se le asignó un peso igual a cada uno de estos índices; sin embargo, el modelo puede ser ajustado para hacer énfasis en algún índice específico dependiendo de las necesidades de análisis que se presenten y del tipo de sistemas de grid computing que se esté analizando.

El resultado de IBECOD es con base en una puntuación máxima de 100. En el Cuadro 58, se observa que el sistema SDSC está utilizando sus recursos e infraestructura en un 50%, lo cual implica que se está haciendo un uso no eficiente de sus recursos y que se debe investigar -en detalle- cada uno de los indicadores para determinar la mejor forma de optimizar el índice IBECOD. En cambio, el sistema OSC Linux Cluster presenta un IBECOD de aproximadamente un 82%, lo cual indica que se está haciendo un uso superior de los recursos computacionales; aunque siempre es posible hacer mejoras dentro de este sistema para subir este índice. Se debe recordar que la máxima nota del IBECOD es 100% y sería ideal contar con sistema que posea esta puntuación.

Cuadro 58 Cálculo de IBECOD

Indicador	Índice Básico de Eficiencia de Computación Distribuida					
	SDSC Blue			OSC Linux		
	2000	2001	2002	2000	2001	2002
Índice de Gestión Económica	22.92	20.59	16.25	27.78	25.66	24.61
Índice de Gestión Operacional	19.21	20.49	18.20	24.70	21.49	21.70
Índice de Eficiencia en Infraestructura	5.67	11.77	6.18	33.33	33.33	33.33
Índice Básico de Eficiencia de Computación Distribuida	47.80	52.85	40.63	85.81	80.49	79.64

Fuente: Elaboración Propia, con base en el análisis realizado.

En general, en este apartado se ha mostrado la factibilidad de cálculo del IBECOD y su aplicabilidad para analizar la posición de un sistema grid con respecto a otros, con lo cual la administración puede tomar las medidas necesarias para mejorar su eficiencia global.

E. Comprobación de hipótesis

En esta sección, se vuelven a listar las hipótesis presentadas en el Capítulo II con el fin de relacionar éstas con la investigación realizada a través de fuentes primarias y los sujetos de información. Cada una de ellas, va seguida por comentarios, los cuales muestran la ubicación de los elementos que hacen sustentables cada una de las hipótesis.

Se procede a sintetizar la hipótesis general de investigación y los resultados obtenidos mediante la aplicación de las distintas técnicas de recolección y análisis de información, los cuales han sido orientados por los indicadores y los respectivos instrumentos de medición; estos, en su conjunto, contribuyeron al desarrollo sistemático de esta tesis doctoral.

La hipótesis general de trabajo es la siguiente: “La eficiencia de los sistemas de informática distribuida (IBECOD) puede determinarse mediante el análisis de la gestión económica (IGE), de la eficiencia en infraestructura (IEI) y de la gestión operacional (IGO) del grid computing”.

Durante el desarrollo esta investigación, para la hipótesis de trabajo, se determinó que los índices independientes están representadas por la gestión económica, la gestión operacional y la eficiencia en infraestructura. Cada uno de estos índices se representa por una serie de indicadores, los cuales son presentados a continuación.

El índice independiente “gestión económica” fue documentada mediante cinco indicadores: ingresos de la empresa, evolución de los ingresos a través de los años, cambio en los costos debido a la aplicación de distintos modelos de asignación y porcentaje de cambios en costos gracias a la aplicación de diferentes modelos de asignación. Este índice fue estudiado en el Capítulo de Análisis de Resultados en el apartado de Índice de Gestión Económica IGE.

El índice de “gestión operacional” es calculado mediante siete indicadores: tiempo de espera por mes, tiempo de espera por día de la semana, tiempo ejecución de las tareas por mes, cantidad de trabajos atendidos por mes, memoria utilizada por mes, cantidad de trabajos recibidos por día y cantidad de trabajos atendidos según hora del día. Este índice es estudiado en el Capítulo de Análisis de Resultados en el apartado de Índice de Gestión Operacional IGO.

El índice “eficiencia en infraestructura” es definido mediante los siguientes indicadores: cantidad de información a transportar, capacidad de la red de transmisión de datos, número de viajes para realizar el transporte de información y resultados, cantidad de información transportable, tiempo empleado por los paquetes de información en su operación dentro del grid, tiempo empleado por la información desde su salida hasta recibir los resultados del procesamiento, tiempo de actividad de los nodos del grid, cantidad de información manejada por el grid y velocidad de procesamiento de la información dentro del grid. Este índice fue estudiado en el Capítulo de Análisis de Resultados en el apartado de Índice de Eficiencia en Infraestructura IEI.

Con base en los resultados expuestos en el Capítulo de Análisis de Resultados y la síntesis referenciada en la hipótesis de trabajo, se comprueba la validez del sistema de índices planteado inicialmente para el desarrollo de esta investigación y que fue presentado esquemáticamente por medio del siguiente enunciado:

$$IBECOD = IGE + IEI + IGO + \xi$$

IGE: Índice de Gestión Económica.

IEI: Índice de Eficiencia en Infraestructura.

IGO: Índice de Gestión Operacional.

ξ : Margen de error.

La validez de este enunciado queda comprobada en el Capítulo de Análisis de Resultados, ya que cada una de los índices e indicadores planteados fueron calculados de forma exitosa y los resultados obtenidos permiten analizar las características económicas, operacionales y de infraestructura de los sistemas de grid computing. Incluso fue posible representar la eficiencia global de cada grid mediante el IBECOD, lo cual permite que las empresas determinen su posición relativa en el mercado y con la información arrojada por los indicadores individuales pueden tomar las medidas necesarias para mejorar su eficiencia global.

De igual forma, el IBECOD puede ser utilizado no sólo para conocer la situación con respecto a la competencia, sino que cuál ha sido la evolución de la empresa a través del tiempo. Esto es evidente en el siguiente cuadro, en el cual se muestra que el cálculo de IBECOD es posible y permite el análisis de la situación de la empresa. Por ejemplo, el sistema SDSC Blue necesita mejorar su eficiencia, porque está por debajo de la eficiencia del sistema OSC Linux.

También el cálculo de IBECOD permite comparar la eficiencia del grid a través de los años. Por ejemplo, OSC Linux -en el año 2000- posee un IBECOD de 85.81; sin embargo, éste baja en el año 2001 y 2002. Por tanto, la administración obtiene de forma rápida un índice que le permite determinar si la empresa está mejorando o no a través de los años, y de esta forma si se desea analizar las causas de un buen o mal rendimiento es posible analizar cada uno de los componentes de IBECOD e, incluso, los indicadores individuales y así llegar a determinar el origen del cambio en eficiencia de la empresa.

Cuadro 58 Cálculo de IBECOD

Indicador	Índice Básico de Eficiencia de Computación Distribuida					
	SDSC Blue			OSC Linux		
	2000	2001	2002	2000	2001	2002
Índice de Gestión Económica	22.92	20.59	16.25	27.78	25.66	24.61
Índice de Gestión Operacional	19.21	20.49	18.20	24.70	21.49	21.70
Índice de Eficiencia en Infraestructura	5.67	11.77	6.18	33.33	33.33	33.33
Índice Básico de Eficiencia de Computación Distribuida	47.80	52.85	40.63	85.81	80.49	79.64

Fuente: Elaboración Propia, con base en el análisis realizado.

Específicamente, la aplicación del Índice IBECOD permitió determinar qué tan eficientes son los sistemas grid computing. En esta investigación se calculó el IBECOD para los sistemas SDSC Blue y OSC Linux Cluster, lo cual da como resultado que el sistema OSC Linux es el más eficiente. De esta forma, se demostró que este índice permite la compra de sistemas de informática distribuida que poseen características de infraestructura y de calendarización distintas, de forma tal que los administradores de un sistema grid en particular pueden compararse con otros con el fin de determinar si son competitivos o no.

De igual manera, en el Capítulo de Análisis de Resultados, se sensibilizó el Índice IBECOD por medio de simulaciones para demostrar que se puede medir –efectivamente- la relación que existe entre los tiempos de espera dentro del grid (índice de eficiencia de la informática distribuida) y la velocidad de procesamiento de cargas de información.

Igualmente, se demostró que el Índice IBECOD modela -de forma adecuada- el comportamiento de los indicadores influyentes en la eficiencia del sistema de informática distribuida.

El impacto económico e impacto en eficiencia que tienen los algoritmos de distribución de procesos y asignación de recursos en un ambiente de grid computing, fueron estudiados en el Capítulo de Análisis de Resultados en el apartado de “Cambio en los costos debido a la

aplicación de distintos modelos de asignación (CAMCOSTO) y Porcentaje de Cambio en los costos debido a la aplicación de distintos modelos de asignación (PORCAMCOSTO)”.

Para sustentar esta afirmación, se realizaron experimentos y pruebas bajo dos modelos económicos de calendarización distintos: tiempo compartido y espacio compartido. En el modelo de tiempo compartido, todos los consumidores del mercado comparten -de forma equitativa- todos los recursos; en cambio -en el modelo de espacio compartido- los primeros consumidores que solicitan los recursos, tienen prioridad sobre los consumidores que lo hacen después.

El impacto fue analizado para los sistemas de grid computing -sujeto de estudio- y los resultados obtenidos determinaron que, efectivamente, existe un impacto económico cuando se cambia el algoritmo de distribución de procesos y asignación de recursos; específicamente, tanto en el sistema SDSC Blue como en el OSC Linux Cluster la aplicación de los algoritmos de tiempo compartido representan un costo superior comparado con los algoritmos de espacio compartido. El detalle de este análisis y el fundamento de estas afirmaciones pueden ser consultadas en el Capítulo de Análisis de Resultados.

Con respecto a las hipótesis planteadas para esta investigación y con base en los estudios presentados en el trabajo de investigación, tanto práctico como teórico, nos llevan a ratificar los parámetros con los cuales se definió el Índice Básico de Eficiencia de Computación Distribuida (IBECOD).

- Una infraestructura adecuada y acorde con las tendencias de la demanda de los consumidores hace posible lograr mayor eficiencia en el procesamiento de cargas de información.

- La eficiencia operacional o productividad, determinada por los indicadores que evalúen los rendimientos del grid, incluirán una actuación permanente sobre la demanda de los servicios de procesamiento.
- Una gestión económica eficiente de los sistemas de administración, que simplifiquen los procedimientos y faciliten los procesos de comercio.

VIII. Conclusiones

Objetivo General. Definir un modelo económico matemático que permita medir la eficiencia de la distribución de procesos y asignación de recursos en un ambiente de grid computing.

1. El indicador de eficiencia en manejo de recursos influye directamente en los efectos operacionales tales como: cantidad de datos manipulados en el grid y los tiempos de espera de datos que deben ser procesados. Estos parámetros son de gran importancia debido a que influyen en la calidad del servicio prestado y en los costos tanto de los dueños de la información que debe ser procesada como de los administradores del grid. Además, si se desea obtener un máximo retorno de la inversión realizada en la infraestructura del grid, es vital que éste sea eficiente y se disminuyan los costos de operación.
2. La aplicación del Índice Básico de Eficiencia de Computación Distribuida (IBECOD), el cual resume la eficiencia de los grid de computadoras en cuanto al procesamiento de cargas de trabajo, puede ser utilizado como marco de referencia para comparar los modelos de administración y procesamiento de distintos grids. Por ejemplo, en esta investigación se demostró que el IBECOD del sistema de informática distribuida SDSC Blue y de OSC Linux puede ser calculados y comparados entre ellos. Con esta información, los administradores de SDSC y de OSC Linux pueden iniciar el análisis de los distintos factores que están provocando una eficiencia determinada y, de esa manera, tomar las medidas correctivas correspondientes con el fin de ser más competitivos en el mercado.
3. Con base en las simulaciones realizadas, se puede concluir -que desde el punto de vista de costos y rendimiento- un aumento en la velocidad de procesamiento de

datos en los grids incide directamente en los indicadores de eficiencia IBECOD. Así, el índice de atención de cargas de trabajo, el cual se refiere al número de cargas de trabajo atendidas por día; el índice de estadía de las cargas de trabajo en el grid, que se mide en minutos y resume la composición de otros índices dependientes como tiempo de espera. Estos indicadores sirven como un punto de partida para el análisis operacional del grid, de forma tal, que puedan ser mejorados con el fin de utilizar la infraestructura del grid de forma eficiente y, por lo tanto, repercutiendo directamente en las ganancias obtenidas del procesamiento de información.

4. La velocidad de manipulación de datos dentro de un sistema de informática distribuida, ejerce gran influencia sobre la cantidad de información procesada dentro del grid. Basta con que aumente la velocidad de procesamiento de información en unos puntos para que el total de información procesable por el grid aumente considerablemente.
5. El comportamiento de los ingresos en relación con el aumento de eficiencia en el procesamiento de información no es de tipo exponencial, ya que existe una asíntota en la curva de crecimiento.

El aumento en la eficiencia, al principio, genera grandes aumentos en los ingresos. Sin embargo, llega el momento en que los aumentos de velocidad de procesamiento no generan grandes cambios en los ingresos percibidos. De esta forma, es necesario que las empresas proveedoras de servicios de informática distribuida, realicen análisis de forma que se encuentre el punto de equilibrio entre aumento de eficiencia y aumento de los ingresos, debido a que todo aumento en eficiencia tiene un costo relacionado, el cual -no necesariamente- contribuirá a un incremento en los ingresos percibidos.

Por lo tanto, las empresas administradoras de grids deben aumentar su eficiencia hasta que se llegue al punto máximo de eficiencia con la infraestructura existente, pues si se intenta seguir aumentando la eficiencia con el fin de mejorar las ganancias y disminuir los costos, se llegará a un punto en el cual la inversión realizada para mejorar la eficiencia no se verá reflejada con aumentos de ganancias o disminución de costos.

Para que una empresa que brinda servicios de informática distribuida encuentre ese punto de equilibrio entre eficiencia y ganancias, se recomienda la realización de un análisis periódico de los distintos indicadores de eficiencia que se sugieren en esta investigación. De esta forma, siempre se tendrá información detallada con la cual se pueden tomar las medidas correctivas necesarias para lograr un uso óptimo de la infraestructura del grid.

6. En esta investigación, se ha demostrado que el aumento de eficiencia tiene una asíntota, de forma tal el aumento de eficiencia de los nodos individuales tiene un límite. Por tanto, cuando una empresa proveedora de informática distribuida llega al punto máximo de eficiencia individual de los nodos, la única manera de aumentar su capacidad de procesamiento global es agregando nuevos nodos de procesamiento al grid.

De los análisis realizados, se desprende que el aumento de eficiencia tiene un límite. Cuando se alcanza que el límite de eficiencia individual sea mayor o igual al 90%, el incremento en la cantidad de tareas atendidas no es significativo. Por lo tanto, la inversión requerida para aumentar la eficiencia no se recuperará en un plazo razonablemente corto.

Objetivo específico 1. Analizar el impacto económico e impacto en eficiencia que tienen los algoritmos de distribución de procesos y asignación de recursos en un ambiente de grid computing.

1. Los resultados de los análisis realizados arrojan que el algoritmo de distribución no incide en el tiempo de transporte de información. Este fenómeno se debe a que los dos modelos utilizan la misma infraestructura de comunicación; por tanto, el tiempo de transporte de información no depende de los algoritmos internos de procesamiento del grid, sino de la capacidad de infraestructura de red.
2. De la misma forma, con respecto a los tiempos de duración de atención de las tareas dentro del grid, los resultados muestran que no existe una diferencia entre los tiempos de atención utilizando el modelo de tiempo compartido o el modelo de espacio compartido. Cabe recordar que la infraestructura de componentes del grid son los mismos, lo único que cambia es el modelo de asignación de recursos.
3. El modelo de asignación de recursos afecta, directamente, sus costos de operación. Esto quedó demostrado en el Capítulo de Análisis de Resultados, en donde se determina que el modelo de tiempo compartido implica costos más altos. Por tanto, los algoritmos de calendarización de las empresas de grid computing influyen directamente en los costos.

Objetivo específico 2. Analizar la forma en la cuál está siendo utilizada la infraestructura existente en las empresas que brindan servicios de grid computing y su aporte en la eficiencia.

1. Una de las principales ventajas de la tecnología de informática distribuida es que cuantos más nodos de procesamiento estén disponibles en el grid, menor será la velocidad de procesamiento necesaria para atender la misma cantidad de

información. De esta forma, si una empresa -debido a restricciones de software o hardware- no puede aumentar la velocidad de procesamiento individual de los nodos, entonces podrá preferir por agregar más nodos con el fin de procesar cantidades mayores de información, debido a que en la mayoría de los nodos de procesamiento se utilizan sistemas operativos de código abierto, con lo cual el precio de licenciamiento es muy bajo o cero; por lo tanto, el costo de agregar nuevos nodos es relativamente bajo.

Objetivo específico 3. Analizar la gestión operacional de los sistemas de grid computing con el fin de determinar su impacto en la eficiencia.

1. La tarea de optimizar la velocidad de atención y procesamiento de tareas en la informática distribuida, es responsabilidad de los administradores del grid. Para lograr esta meta se requiere una continua innovación tecnológica, debido a que la infraestructura del grid influye directamente en la obtención de un índice de eficiencia alto y competitivo. Además, es necesario determinar cuáles son los algoritmos de distribución de carga que permitan mejorar el índice de eficiencia y que maximice el uso de los recursos disponibles.

Un aspecto importante es que si se identifican tendencias de uso de los recursos del grid, será posible prepararse para atender -de la forma más eficiente- a los clientes que solicitan procesamiento de tareas.

Otro factor significativo es el tiempo que una tarea permanece esperando para ser atendido, lo cual afecta directamente la calidad de servicio y provoca retrasos a los

consumidores ya que sus operaciones podrían estar paralizadas debido a la espera de resultados del procesamiento

Objetivo específico 4. Analizar el comportamiento de indicadores económicos de los sistemas de grid computing comprendidos dentro esta investigación.

1. Con base en el análisis del indicador de ingresos de la empresa, se determinó que cuanto más alto sea el promedio de uso de CPU de las tareas, los ingresos percibidos por la empresa son superiores.
2. Con base en los datos analizados en esta investigación, el escenario ideal para una empresa de grid computing se presenta cuando los trabajos atendidos utilizan mayor cantidad de procesador, pues los ingresos dependen de esa variable. Sin embargo, para optimizar el uso de la infraestructura se debe aumentar la tasa de atención de tareas concurrentes.
3. Cuando se analiza la evolución de los ingresos a través del tiempo y su comportamiento es sostenido, si se muestra una disminución en este sentido se debe a que la cantidad de tiempo de CPU utilizada por los procesos atendidos es inferior. Por tanto, lo más importante para un sistema de grid computing es la cantidad de CPU que es consumido por los clientes y no tanto la cantidad de tareas recibidas para su procesamiento.
4. Un alto porcentaje de tareas no atendidas implica que los clientes pierden confiabilidad en la capacidad de la empresa de grid computing para realizar el procesamiento oportuno de tareas. Por consiguiente, podría implicar que éstos clientes busquen otros proveedores de servicios que les brinden un nivel de asistencia más alto.

5. La atención no oportuna de las tareas afecta la imagen de la empresa de grid computing ante sus clientes, también implica dinero no percibido para la empresa. Esto se debe a que la tarea fue enviada por los clientes al grid; posteriormente, esta tarea fue calendarizada y su ejecución se inicia. Sin embargo, debido a que dicha tarea no es completada dentro de los términos especificados por el cliente se cancela. Por tanto, si esto ocurre, el cliente no paga ninguna tarifa a la empresa de grid computing. Esta es una situación crítica, pues la tarea cancelada utilizó tiempo de CPU que no fue cobrado y que podría haber sido utilizado para atender a otros clientes. Por ejemplo, el sistema SDSC Blue, en el año 2000, tuvo impacto económico negativo de \$653.062,73 debido al no atender –oportunamente- las tareas de procesamiento recibidas.

IX. Recomendaciones

Objetivo General. Definir un modelo económico matemático que permita medir la eficiencia de la distribución de procesos y asignación de recursos en un ambiente de grid computing.

1. En los ambientes de grid computing, los productores (dueños de los recursos) y los consumidores (usuarios de los recursos) tienen metas, objetivos, estrategias y patrones de oferta y demanda distintos. Debido a la naturaleza de las relaciones de los productores y los consumidores se hace necesario utilizar un enfoque económico para la administración de recursos dentro de ambientes grid computing.
2. Un enfoque económico posee las bases para la administración de la descentralización y heterogeneidad del grid, pues estos factores están presentes en las economías humanas. Los modelos económicos de competitividad tienen algoritmos, políticas y herramientas que se pueden aplicar en el manejo de recursos (compartir, asignar) de los sistemas de grid computing.

Objetivo específico 1. Analizar el impacto económico e impacto en eficiencia que tienen los algoritmos de distribución de procesos y asignación de recursos en un ambiente de grid computing.

1. El factor principal de un modelo económico se basa en que el precio de los bienes se establece con base en la demanda y oferta; por lo tanto, en un ambiente de grid computing los usuarios son entes dentro del mercado que compiten con otros para utilizar los recursos.

2. Específicamente, un modelo económico de grid computing debe poseer las siguientes características:
 - Ha de asumir un modelo para establecer el valor de los recursos de los productores.
 - El modelo tiene que poseer mecanismos para regir sus precios y publicación.
 - Los mecanismos de negociación de precios deben ser contemplados dentro del modelo.
 - Tiene que existir un lugar, en el cual se puedan localizar las entidades participantes en el grid; de forma tal, que los usuarios puedan buscar los recursos que necesitan.
3. La calidad del servicio debe ser tomada en cuenta durante el proceso de calendarización de tareas.

Objetivo específico 2. Analizar la forma en la cuál está siendo utilizada la infraestructura existente en las empresas que brindan servicios de grid computing y su aporte en la eficiencia.

1. La mayoría de los sistemas de informática distribuida incluyen calendarizadores de trabajos, pero a medida que el campo de acceso geográfico del grid se expende, existirá la necesidad de meta calendarizadores que puedan administrar los distintos conjuntos de grids. Estos deben desarrollarse para considerar múltiples factores y no, únicamente, la capacidad de CPU. Se debe tomar en cuenta aspectos que permitan aumentar las ganancias económicas, tanto para los consumidores como para los dueños de los grids.
2. Deberá plantearse un estándar que consienta regir el mercado de la informática distribuida. De la misma manera que en la economía convencional existen tratados de libre comercio y tratados bilaterales, es necesario que existan tratados, los cuales

permitan regular el mercado de compra y venta de recursos de procesamiento de información.

3. Del estudio sobre grid computing realizado por Platform Computing, se puede concluir que la industria está interesada en adoptar la tecnología de informática distribuida, principalmente debido a que es posible incrementar la productividad de las empresas con tasas de retorno de inversión favorables desde el punto de vista económico.
4. A pesar de que existen muchas iniciativas de grid computing, cada una tiene su propia estructura de operación. Para reducir costos y facilitar la integración, será necesario establecer estándares aceptados por la industria.

Objetivo específico 3. Analizar la gestión operacional de los sistemas de grid computing con el fin de determinar su impacto en la eficiencia.

1. Si una empresa proveedora de servicios de grid computing puede lograr un aumento en la velocidad de procesamiento y administración de carga, podrá optimizar los costos de operación, debido a que se pueden eliminar nodos de procesamiento ociosos, sin que esto modifique la cantidad de cargas de información que puede estar atendiendo por ahora, con una menor cantidad de nodos y, por consiguiente, con menores costos de operación.
2. Además, si se identifican nodos ociosos es posible que estos puedan ser arrendados a otros proveedores de grid computing, este es un ambiente posible debido a los progresos que se han dado en la construcción de canales de cooperación y comunicación entre distintos grids distribuidos a nivel mundial. Por ejemplo, en el Research National Park (2006) un grupo de científicos de Estados Unidos y Japón,

demonstraron que es probable crear interoperabilidad automática entre dos grids con infraestructuras totalmente diferentes y separadas de manera geográfica. Este mecanismo permite que el potencial de procesamiento y comunicación de datos sea escalable y, gracias a las redes de alta velocidad, es permisible que la próxima generación de nodos de procesamiento de cargas de información crezca de forma exponencial.

3. Un aumento en la productividad del grid provoca un alto impacto en la calidad de servicio percibida por los clientes y, además, se acrecientan los ingresos de los dueños del grid. Entonces, la evaluación de resultados en los factores presentados por el investigador ponen claro que existe una excelente oportunidad para inducir los cambios necesarios que logren un aumento en la productividad y la capacidad los nodos de informática distribuida.

Objetivo específico 4. Analizar el comportamiento de indicadores económicos de los sistemas de grid computing comprendidos dentro esta investigación.

1. Cuando índice de tareas no atendidas (PORCAMEF) es alto, la empresa proveedora de grid computing debería revisar sus algoritmos de asignación, priorización y calendarización de recursos, ya que no se están atendiendo correctamente las tareas y se está perdiendo dinero. En este caso, la empresa debería establecer un mecanismo idóneo de detectar que el grid cuenta con la capacidad necesaria para ejecutar la tarea antes de iniciarla. De esta forma, se evitará la inversión de tiempo y recursos en tratar de ejecutar una tarea, cuyo procesamiento excede la capacidad de respuesta de la infraestructura actual del grid.

Sobre los indicadores de operación

1. Si un sistema de informática distribuida está compuesto por distintos conjuntos de nodos de procesamiento, es necesario distribuir los algoritmos de calendarización de forma equitativa y las cargas de trabajo, de forma tal, que se reduzca el tiempo de espera. Así, se disminuye este índice y se mejora la calidad de servicio.
2. En aquellos casos, cuando el sistema de informática distribuida presenta un índice de tiempo de espera alto con frecuencia estacional, los aumentos en los tiempos de espera se deben a que durante estos periodos se recibe una mayor cantidad de tareas o son tareas más complejas. Este comportamiento indica que es preciso analizar la carga de trabajo de estos periodos y determinar si es posible optimizar la infraestructura existente o, más bien, se debe aumentar la capacidad global del grid. Si se decide hacer esto último, será necesario realizar estudios de retorno de la inversión para determinar si la inversión realizada en infraestructura dará los resultados esperados.
3. Cuando se aumenta, de forma abrupta, la cantidad de trabajos recibidos para su procesamiento dentro del grid, la infraestructura colapsa y los trabajos empiezan a ser atendidos de forma más lenta. Sin embargo, una vez que se han atendido estos trabajos extraordinarios, generalmente el grid vuelve a tiempos de respuesta promedio. Con estos datos, se puede determinar que cuando se ha detectado un aumento importante en la cantidad de trabajos recibidos, es necesario que el grid se ajuste automáticamente y se agregue mayor poder computacional a este, lo cual puede lograrse mediante la adición temporal de nodos al grid, durante tiempos de alta demanda.
4. Si se detectan periodos de alta demanda de recursos, los administradores de grid deberán tomar las medidas necesarias para optimizar el uso de la infraestructura o,

si es posible, realizar ajustes que logren agregar más recursos al grid durante esos periodos. Estos recursos pueden ser propios o se pueden realizar alianzas estratégicas con otros grids, de forma tal, que se baje el tiempo de atención de las tareas, con lo cual los clientes se ven beneficiados y se podría crear un sentido de lealtad hacia el grid, puesto que los clientes verán que éste es capaz de procesar la información en forma eficiente.

5. Cuando un sistema de informática distribuida muestra un alza significativa en el uso de memoria, es necesario que la administración del grid cuente con las provisiones necesarias para suplir la demanda. Esto puede lograrse aumentando la memoria en todos los nodos del sistema; sin embargo, implicaría costos fijos muy elevados. Una opción más adecuada es agregar nodos adicionales únicamente durante este periodo, deben contar con la memoria suficiente para suplir la demanda, de esta forma el grid es dinámico y crece únicamente en el momento necesario, con lo cual se reducen los costos fijos.

X. Sugerencias.

En un ambiente de procesamiento masivo de datos, la existencia de una cantidad ilimitada de datos se presenta únicamente en casos especiales, en los cuales se cuenta con una fuente de datos ilimitada. Un ejemplo de estos es el proyecto SETI que está procesando las señales provenientes del espacio exterior, por lo tanto, tiene una fuente de datos ilimitada.

No obstante que la disponibilidad de carga es limitada, los proveedores de servicios de informática distribuida deben contar siempre con la infraestructura necesaria para operar en caso de que se presenten picos de demanda de procesamiento.

En trabajos futuros, se podría realizar el análisis detallado de la influencia que tiene el aumento o disminución de los nodos del grid sobre su eficiencia. A continuación se esboza dicho planteamiento, el cual es una adaptación del modelo propuesto por Santibañez (2006).

$$\Delta F = F \times \left(\frac{C_{me}}{C_p} - 1 \right)$$

Ecuación 26 Efectos del cambio en la cantidad de nodos en el gris.

Donde:

Parámetro	Descripción
ΔF	Incremento o disminución del grid
F	Cantidad de nodos actual
C_{me}	Carga de información promedio equivalente
C_p	Nivel límite de ocupación

La carga promedio equivalente se establece utilizando la media geométrica de las observaciones diarias de carga promedio por nodo, durante los días de evaluación de acuerdo con:

$$C_{me} = \sqrt{\frac{\sum Chp_i^2}{m}}$$

Ecuación 27 Cálculo de la Carga promedio

Donde:

Parámetro	Descripción
C_{me}	Carga promedio equivalente
Chp_i^2	Carga recibida durante el día i
m	Número de días evaluados

El valor de la carga promedio del día Chp_i se establecerá a partir del promedio aritmético de los nodos contabilizados en la hora y tiempo exacto por medio de la siguiente fórmula:

$$Chp_i^2 = \frac{\sum C_{ij}}{n_i}$$

Ecuación 28 Carga promedio equivalente diaria

Donde C_{ij} es la carga de cada nodo j de la ruta en el tiempo específica en el la pico día i y n_i es el total de nodos.

El costo generalizado de transporte, el cual es el valor promedio del tiempo de espera, del tiempo de acceso y del tiempo de viaje de la información desde el punto de inicio y el envío de los resultados se puede expresar:

$$C = (2T_v + T_e + T_c) \times V_{stv}$$

Ecuación 29 Costo generalizado de procesamiento

Donde:

Parámetro	Descripción
C	Costo generalizado de transporte
T_v	Tiempo utilizado durante el envío de la información desde el punto de origen hasta llegar al grid
T_e	Tiempo de espera para que la carga de información empiece a ser atendida
T_c	Tiempo de procesamiento de la carga
V_{stv}	Valor subjetivo del tiempo de espera

La aplicación de las ecuaciones formuladas anteriormente permite que se estime el impacto del aumento o disminución de nodos en el grid.

XI. Anexo I

A. Introducción

En este anexo, se incluye el modelo propuesto por el Dr. Leopoldo Santibáñez que fue utilizado como base para el desarrollo de la investigación realizada en este trabajo.

B. INDICADOR DE EFICIENCIA EN INFRAESTRUCTURA (IEI)

El **IEI** indicador de eficiencia en infraestructura será investigado a la luz de efectos operacionales en otros conceptos dependientes tales como manipulación de carga en puertos versus tiempos de espera de naves en puertos; ambos parámetros son de gran importancia en el transporte marítimo y la operación marítimo-portuaria. El análisis instrumental de estas variables nos lleva a incluir otras como: carga por transportar y capacidad de buques, entre otros. Estos determinaron un modelo matemático que explica la importancia de la infraestructura en los puertos, caracterizada y medida por variables linealmente dependientes, que afirman la determinación de nuestra variable dependiente.

$$\boxed{\text{Eficiencia Portuaria: } IEP \propto IEI}$$

EFICIENCIA PORTUARIA Y LA IMPORTANCIA DEL TIEMPO DE ESTADIA EN PUERTOS PARA EL TRANSPORTE MARÍTIMO

Parte del sentido común está en entender que es beneficioso, en términos de costos y rendimientos, aumentar la velocidad de manipulación de carga en los puertos, lo cual incide directamente en los indicadores de eficiencia portuaria (IEP), diseñados en esta investigación; sin embargo, el problema es: ¿Cómo demostrarlo?

Las mayores dificultades que se le presentan al investigador, son encontrar ciertas relaciones que, reducidas a una fórmula o ecuación, permitan configurar un cierto modelo

matemático, el cual posibilite la aplicación de algoritmos u otras herramientas que entreguen certezas y resultados. Además, debemos demostrar -de manera rigurosa y exacta- que un aumento en la velocidad de manipulación de carga, se traduce en un beneficio, expresable en cualquier unidad adecuada: toneladas adicionales de carga (para carga a granel), TEUS (para carga contenedorizada), porcentaje de rentabilidad (sobre las inversiones) o simplemente, utilidades de operación.

Siempre conviene en en estos casos, recurrir a bases físicas y/o matemáticas que nos deben llevar -inequívocamente- a diseñar las relaciones o ecuaciones, que con fundamentos, nos darán el punto de partida de lo que queremos medir, evaluar, investigar y proponer.

Estos procedimientos serán desarrollados, inicialmente, al asumir ciertas restricciones para entregar un modelo general, cuya ventaja está en comprender la dinámica del movimiento de naves y la atención de puertos; además, en extender el conocimiento que se quiere compartir, para que éste sea un vehículo en la búsqueda de soluciones alternativas.

Por lo demás, lo que aquí se mostrará, es de aplicación general en el ámbito del transporte. Es igualmente aplicable y valedero al transporte aéreo, marítimo o terrestre y por supuesto - como en este caso- a las operaciones portuarias.

FACTORES BÁSICOS DEL TRANSPORTE Y LA CONSTRUCCIÓN DE UN MODELO MATEMÁTICO PARA DEFINIR LA EFICIENCIA PORTUARIA

Las variables que se utilizarán en el modelo matemático, para definir la eficiencia portuaria, las describimos a continuación.

Siglas Utilizadas	Significado de la Variable del Modelo Matemático
NV =	Número de viajes.
K =	Cantidad de carga por transportar.

Siglas Utilizadas	Significado de la Variable del Modelo Matemático
D =	Cantidad de carga del buque.
Nva =	Número de viajes anuales (ida y retorno).
Ka =	Cantidad de carga transportable en un año y en ambos sentidos.
D =	Capacidad <u>útil</u> de carga del buque.
Tv =	Tiempo ocupado por un buque en un viaje completo (ida y retorno).
TPV =	Tiempo empleado por la nave en su operación de puertos.
TNV =	Tiempo empleado por la nave en la navegación desde el zarpe a la llegada al puerto de retorno.
Ta =	Tiempo de actividad
KMV =	Tonelaje operado en puerto (carga más descarga)
VM =	Velocidad de manipulación

Tres, son los elementos principales que aparecen e intervienen en todo proceso de transporte (asumiremos el transporte marítimo):

➤ *La cantidad de carga por transportar*

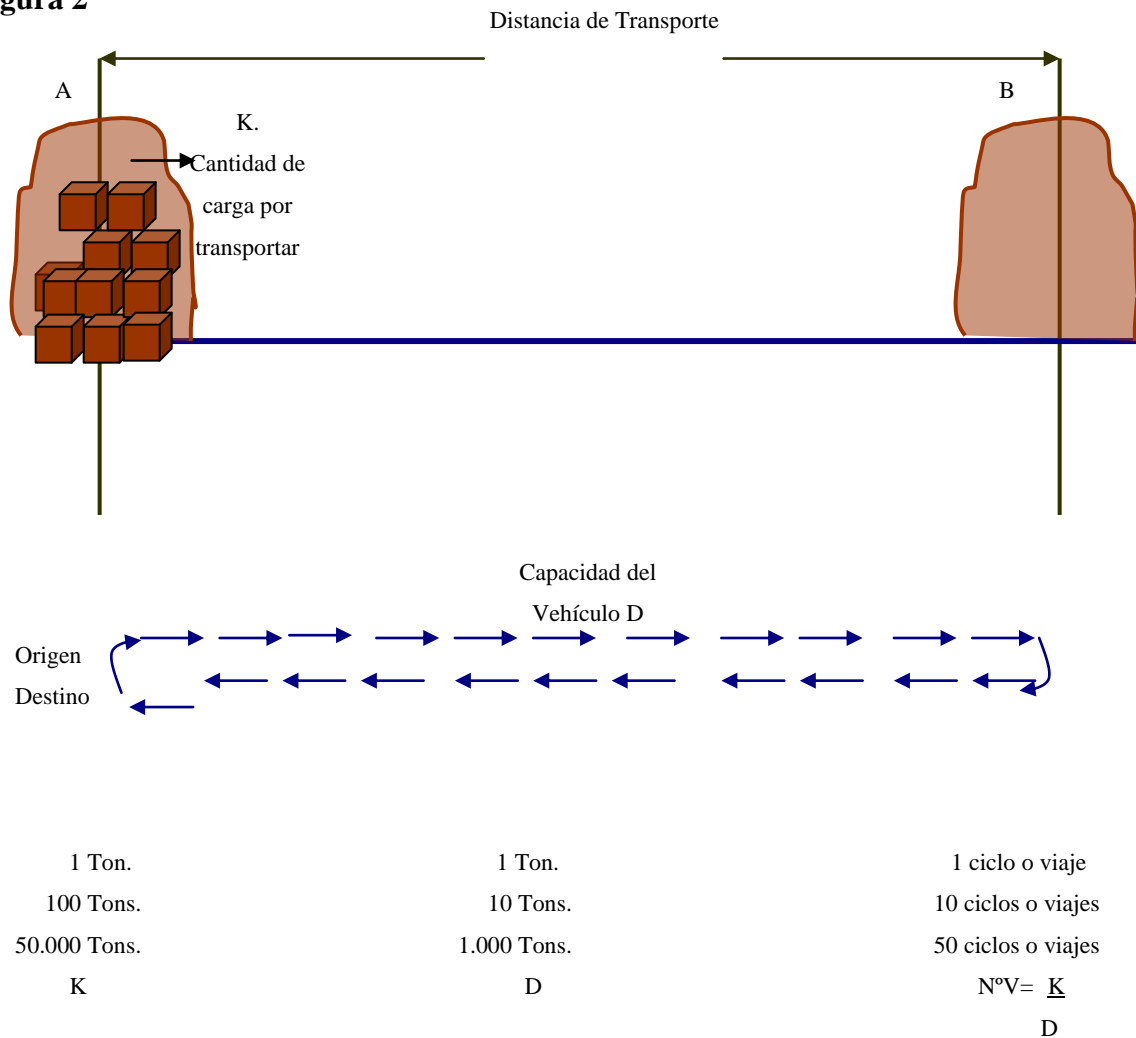
En cargas de distinto origen y composición: granel, granel líquido o contenedores, entre otras.

➤ *La distancia que deberá recorrer la nave con la carga por transportar: tráfico de comercio, líneas comerciales.*

➤ *La capacidad del vehículo utilizado en la operación*

Diferentes diseños de naves: multipropósitos MPCs, frigoríficos, graneleros, multitanques petroleros, entre otros.

Figura 2



Si la cantidad de carga que debe transportarse es igual a la capacidad de la nave utilizada, el transporte se hará de una sola vez. Pero, como suele suceder, si la capacidad de la nave puede transportar, solamente, una fracción de la cantidad de carga, será necesario efectuar varias veces la operación o ciclo, a fin de completar el transporte total de ella.

En cada ciclo de transporte (ida y retorno) o round trip (viaje redondo), el buque o medio de transporte tomará, de acuerdo con su capacidad, una porción de la carga en el punto de origen; recorrerá con ella la distancia hasta el punto de destino; dejará allí dicha carga y retornará vacío al punto de origen para iniciar un nuevo ciclo o viaje.

Este aspecto recurrente, repetitivo o cíclico es característico de todo proceso de transporte. Por ejemplo, se mencionarán algunas acciones cotidianas:

- El hombre compra y lleva el pan desde la pulpería a su casa.
- El furgón refrigerado transporta la fruta desde la finca en Guanacaste al puerto de embarque que es Caldera.
- El avión traslada los pasajeros desde San José a Miami.
- La nave acarrea miles de toneladas de petróleo, desde puerto Cabello, Venezuela a puerto Limón, Costa Rica.

El número de ciclos o viajes redondos, necesarios para efectuar el transporte, está dado (Véase figura 2) por la relación entre la cantidad de carga por transportar y el tamaño o capacidad del vehículo utilizado en la operación. Según la relación:

Número de ciclos o viajes = Cantidad de carga por transportar

Capacidad del vehículo empleado

NV = número de viajes

K = cantidad de carga por transportar

D = capacidad del buque

Y, reemplazando por los símbolos usados en la figura 1, se obtiene una primera relación:

$$NV = \frac{K}{N \cdot D} \quad (\text{transporte con carga en una dirección})$$

En el transporte marítimo entre dos puertos, el regreso vacío al puerto de origen no es usual, salvo en el transporte de petróleo. Generalmente, el buque de carga, que trabaja en una línea o tráfico, carga y descarga en ambos extremos del viaje; aunque, las cantidades de carga por transportar en ambos sentidos sean diferentes.

Para acotar nuestro diseño, asumiremos las siguientes restricciones:

- Se dispone de carga ilimitada.
- El buque esté a plena capacidad de carga (full carga) en ambos sentidos.

Entonces, el buque transporta D_1 toneladas en un sentido y D_2 toneladas en el retorno, o sea, que en un viaje redondo (A-B-A) conduce o transporta:

Pero $D_1 = D_2$ entonces, $2 D$ toneladas, modificando la ecuación a:

$$NVa = \frac{Ka}{2 D}$$

Se estimará un período de un año de actividad nominal en 355 días y se dejan 10 días de inactividad por si la nave requiere ciertas reparaciones o si -según la planificación- debe entrar a dique por carena (proceso de limpieza del casco de la nave) u otros tratamientos de pintura; por ese motivo, se asigna a las iniciales de nuestras variables el sufijo “a” (anual o anuales), que identifica esta consideración; por lo tanto, éstos deben leerse como sigue:

- NVa** : número de viajes anuales (ida y retorno).
Ka : cantidad de carga transportable en un año y en ambos sentidos.
D : capacidad útil de carga del buque.

EL FACTOR TIEMPO EN LA OPERACIÓN PORTUARIA

Orientando la búsqueda de relaciones para nuestro objetivo, que finalmente es la medición del tiempo de espera de los buques y su incidencia en la eficiencia portuaria, analizaremos el factor tiempo en la operación portuaria.

Como definición, estos procesos de transporte requieren de tiempo para sus ejecuciones:

- cargar en el puerto de origen,
- recorrer la distancia de transporte,
- descargar y cargar nuevamente en el puerto de destino,
- regresar al puerto inicial o de retorno, con lo cual se completa el ciclo o viaje.

El tiempo ocupado en la ejecución de un viaje completo estará compuesto por la suma del tiempo en puertos por viaje y del tiempo de navegación por viaje, medidos en días y expresado en la siguiente relación:

$$\text{TV} = \text{TPV} + \text{TNV} \qquad \text{días} \qquad (2)$$

TV: Tiempo ocupado por un buque en un viaje completo (ida y retorno).

TPV: Tiempo empleado por la nave en su operación de puertos.

TNV: Tiempo empleado por la nave en la navegación desde el zarpe a la llegada al puerto de retorno.

El tiempo que se dispone, anualmente, para la realización de los viajes y que fue explicado con anterioridad, se ha estimado en 355 días. Fue denominado “Ta” (tiempo de actividad).

Si dividimos este tiempo disponible para viajes “Ta”, entre el tiempo ocupado en la ejecución de un viaje “TV”, obtendremos el número de viajes “NVa” que puede efectuar un buque en un año de actividad.

$$NVa = \frac{Ta}{Tv} \quad (3)$$

Y reemplazando Tv según la ecuación (2) nos queda:

$$NVa = \frac{Ta/}{TPV + TNV} \quad (4)$$

Pero según nuestra ecuación (1) que vimos al comienzo, también:

$$NVa = \frac{Ka/}{2 D} \quad (1)$$

por lo tanto:

$$\frac{Ta/}{TPV + TNV} = \frac{Ka/}{2 D}$$

Y despejando “Ka” en la igualdad anterior obtenemos:

$$Ka = \frac{2 D * Ta}{TPV + TNV} \text{ toneladas} \quad (5)$$

ecuación que relaciona la cantidad anual de carga transportable “Ka”, con el tiempo de estadía en puertos “TPV”.

ANÁLISIS DEL TIEMPO DE ESTADÍA DE NAVES EN PUERTOS Y LA VELOCIDAD DE MANIPULACIÓN DE CARGA

El tiempo de permanencia o estadía en puertos, en cada viaje, y que se ha denominado “TPV”, representa el tiempo (expresado en días) que permanece el buque en los puertos, durante un viaje completo o circular. Se mide, generalmente, desde el momento del fondeo o maniobra de atraque del buque hasta el instante del zarpe.

Según lo anterior, es muy importante no confundir TPV, tiempo de permanencia del buque en puerto, con el tiempo ocupado por el buque en las faenas propias de carga y descarga.

Si asumimos que, durante la operación de un viaje con ida y retorno, todo lo que se carga a bordo en el puerto de origen debe ser descargado en el puerto de destino, se conseguirá que el tonelaje manipulado (cargado + descargado) por viaje será siempre el doble del tonelaje transportado “KV” en ese viaje, o sea:

$$\begin{aligned} \text{KMV} &= 2 \text{ KV} \quad \text{expresados en toneladas} \\ \text{KMV} &= \text{tonelaje operado en puerto (carga más descarga)} \end{aligned} \quad (6)$$

La velocidad de manipulación (de acuerdo con la física mecánica $v=d/t$) de la carga será equivalente, entonces, al total del tonelaje manipulado durante el viaje “KMV” por día de estadía en puertos del buque. Puede expresarse por la relación:

$$\text{VM} = \frac{\text{KMV}}{\text{TPV}}$$

VM = Velocidad de manipulación de los puertos y su unidad estará expresada en

Toneladas / día

(7)

despejando de la ecuación anterior el tiempo en puertos por viaje “TPV” se obtiene:

$$TPV = \frac{KMV}{VM} \text{ días} \quad (8)$$

Ahora, si se reemplaza “TPV” en la ecuación (5) que fue desarrollada anteriormente se tiene :

$$Ka = \frac{2 D * Ta}{TPV + TNV} \text{ En toneladas} \quad (5)$$

Finalmente se obtiene:

$$Ka = \frac{2 D * Ta}{\frac{KMV + TNV}{VM}} \text{ En toneladas} \quad (9)$$

Esta última relación nos permite conocer, asociar y medir cómo varía la cantidad de carga transportable por la línea, anualmente - “Ka” - al variar la velocidad de manipulación de carga VM en los puertos de la línea o compañía naviera.

XII. Anexo II

A continuación se presenta el código utilizado durante las simulaciones realizadas en esta investigación.

```

import gridsim.*;
import gridsim.net.*;
import gridsim.util.Workload;
import java.util.*;

/**
 * Test Driver class for grid simulation
 */
public class SchedulingSimulator
{
    /**
     * Creates main() to run this example
     */
    public static void main(String[] args)
    {
        try
        {
            // Step 1: Initialize the GridSim package. It should be
            // before creating any entities. We can't run this example
            // initializing GridSim first. We will get run-time exception
            // error.

            // number of grid user entities + any Workload entities.
            int num_user = 5;
            Calendar calendar = Calendar.getInstance();

            // a flag that denotes whether to trace GridSim events or
            boolean trace_flag = false;

            // Initialize the GridSim package
            System.out.println("Initializing GridSim package");
            GridSim.init(num_user, calendar, trace_flag);

            // Step 2: Creates one or more GridResource entities

```

```

// baud rate and MTU must be big otherwise the simulation
// runs for a very long period.
double baud_rate = 1000000;    // 1 Gbits/sec
double propDelay = 10;    // propagation delay in millisecond
int mtu = 100000;    // max. transmission unit in byte
int rating = 400;    // rating of each PE in MIPS
int i = 0;

// more resources can be created by
// setting totalResource to an appropriate value
int totalResource = 3;
ArrayList resList = new ArrayList(totalResource);
String[] resArray = new String[totalResource];

for (i = 0; i < totalResource; i++)
{
    String resName = "Res_" + i;
    GridResource res = createGridResource(resName, baud_rate,
rating);
                                propDelay, mtu,

    // add a resource into a list
    resList.add(res);
    resArray[i] = resName;
}

////////////////////////////////////
// Step 3: Get the list of trace files. The format should be:
// ASCII text, gzip or zip.

// In this example, I use the trace files from:
// http://www.cs.huji.ac.il/labs/parallel/workload/index.html
String[] fileName = {
(Nirvana)    "l_lanl_o2k.swf.zip",    // LANL Origin 2000 Cluster
              "l_sdsc_blue.swf.txt.gz", // SDSC Blue Horizon
};

String dir = "../";    // location of these files
String customFile = "custom_trace.txt"; // custom trace file
format

// total number of Workload entities
int numWorkload = fileName.length + 1; // including custom
trace

////////////////////////////////////
// Step 4: Creates one or more Workload trace entities.
// Each Workload entity can only read one trace file and
// submit its Gridlets to one grid resource entity.

int resID = 0;

```

```

Random r = new Random();
ArrayList load = new ArrayList();

for (i = 0; i < fileName.length; i++)
{
    resID = r.nextInt(totalResource);
    Workload w = new Workload("Load_"+i, baud_rate,
propDelay, mtu,
                                dir + fileName[i], resArray[resID],
rating);

    // add into a list
    load.add(w);
}

// for the custom trace file format
Workload custom = new Workload("Custom", baud_rate,
propDelay, mtu,
                                dir + customFile,
resArray[resID], rating);

// add into a list
load.add(custom);

// tells the Workload entity what to look for.
// parameters: maxField, jobNum, submitTime, runTime, numPE
custom.setField(4, 1, 2, 3, 4);
custom.setComment("#"); // set "#" as a comment

////////////////////////////////////
// Step 5: Creates one or more grid user entities.

// number of grid user entities
int numUserLeft = num_user - numWorkload;

// number of Gridlets that will be sent to the resource
int totalGridlet = 5;

// create users
ArrayList userList = new ArrayList(numUserLeft);
for (i = 0; i < numUserLeft; i++)
{
    // if trace_flag is set to "true", then this experiment
will
    // create User_i.csv where i = 0 ... (num_user-1)
    NetUser user = new NetUser("User_"+i, totalGridlet,
baud_rate,
                                propDelay, mtu, trace_flag);

    // add a user into a list
    userList.add(user);
}

```

```

////////////////////////////////////
// Step 6: Builds the network topology among entities.

// In this example, the topology is:
// user(s)      --1Gb/s-- r1 --10Gb/s-- r2 --1Gb/s--
GridResource(s)
//                |
// workload(s) --1Gb/s-- |

// create the routers.
// If trace_flag is set to "true", then this experiment will
create
// the following files (apart from sim_trace and sim_report):
// - router1_report.csv
// - router2_report.csv
Router r1 = new RIPRouter("router1", trace_flag); // router
1
Router r2 = new RIPRouter("router2", trace_flag); // router
2

// connect all user entities with r1 with 1Mb/s connection
// For each host, specify which PacketScheduler entity to
use.
NetUser obj = null;
for (i = 0; i < userList.size(); i++)
{
    // A First In First Out Scheduler is being used here.
    // SCFQScheduler can be used for more fairness
    FIFOScheduler userSched = new
FIFOScheduler("NetUserSched_"+i);
    obj = (NetUser) userList.get(i);
    r1.attachHost(obj, userSched);
}

// connect all Workload entities with r1 with 1Mb/s
connection
// For each host, specify which PacketScheduler entity to
use.
Workload w = null;
for (i = 0; i < load.size(); i++)
{
    // A First In First Out Scheduler is being used here.
    // SCFQScheduler can be used for more fairness
    FIFOScheduler loadSched = new
FIFOScheduler("LoadSched_"+i);
    w = (Workload) load.get(i);
    r1.attachHost(w, loadSched);
}

// connect all resource entities with r2 with 1Mb/s
connection
// For each host, specify which PacketScheduler entity to
use.

```

```

        GridResource resObj = null;
        for (i = 0; i < resList.size(); i++)
        {
            FIFOScheduler resSched = new
FIFOScheduler("GridResSched_"+i);
            resObj = (GridResource) resList.get(i);
            r2.attachHost(resObj, resSched);
        }

        // then connect r1 to r2 with 10 Gbits/s connection
        // For each host, specify which PacketScheduler entity to
use.
        baud_rate = 100000000;

        Link link = new SimpleLink("r1_r2_link", baud_rate,
propDelay, mtu);
        FIFOScheduler r1Sched = new FIFOScheduler("r1_Sched");
        FIFOScheduler r2Sched = new FIFOScheduler("r2_Sched");

        // attach r2 to r1
        r1.attachRouter(r2, link, r1Sched, r2Sched);

        ////////////////////////////////////////////////////
        // Step 7: Starts the simulation
        GridSim.startGridSimulation();

        ////////////////////////////////////////////////////
        // Final step: Prints the Gridlets when simulation is over

        // also prints the routing table
        r1.printRoutingTable();
        r2.printRoutingTable();

        GridletList glList = null;
        for (i = 0; i < userList.size(); i++)
        {
            obj = (NetUser) userList.get(i);
            glList = obj.getGridletList();
            printGridletList(glList, obj.get_name(), trace_flag);
        }

        // prints the Gridlets inside a Workload entity
        for (i = 0; i < load.size(); i++)
        {
            w = (Workload) load.get(i);
            w.printGridletList(trace_flag);
        }
    }
    catch (Exception e)
    {
        e.printStackTrace();
        System.out.println("Unwanted errors happen");
    }
}

```

```

}

/**
 * Creates one Grid resource. A Grid resource contains one or more
 * Machines. Similarly, a Machine contains one or more PEs
(Processing
 * Elements or CPUs).
 * <p>
 * In this simple example, we are simulating one Grid resource with
three
 * Machines that contains one or more PEs.
 * @param name          a Grid Resource name
 * @param baud_rate     the bandwidth of this entity
 * @param delay         the propagation delay
 * @param MTU           Maximum Transmission Unit
 * @param rating        a PE rating
 * @return a GridResource object
 */
private static GridResource createGridResource(String name,
                                               double baud_rate, double delay, int MTU, int rating)
{
    System.out.println();
    System.out.println("Starting to create one Grid resource with " +
                       "3 Machines");

    // Here are the steps needed to create a Grid resource:
    // 1. We need to create an object of MachineList to store one or
more
    //    Machines
    MachineList mList = new MachineList();
    //System.out.println("Creates a Machine list");

    // 2. A Machine contains one or more PEs or CPUs. Therefore,
should
    //    create an object of PEList to store these PEs before
creating
    //    a Machine.
    PEList peList1 = new PEList();
    //System.out.println("Creates a PE list for the 1st Machine");

    // 3. Create PEs and add these into an object of PEList.
Rating
    peList1.add( new PE(0,rating) ); // need to store PE id and MIPS
Rating
    peList1.add( new PE(1,rating) );
    peList1.add( new PE(2,rating) );
    peList1.add( new PE(3,rating) );
    //System.out.println("Creates 4 PEs with same MIPS Rating and put
them"+
    //    " into the PE list");

```

```

// 4. Create one Machine with its id and list of PEs or CPUs
mList.add( new Machine(0, peList1) ); // First Machine
//System.out.println("Creates the 1st Machine that has 4 PEs and
" +
//      "stores it into the Machine list");
//System.out.println();

// 5. Repeat the process from 2 if we want to create more
Machines
// NOTE: if you only want to create one Machine for one Grid
resource,
//      then you could skip this step.
PEList peList2 = new PEList();
//System.out.println("Creates a PE list for the 2nd Machine");

peList2.add( new PE(0, rating) );
peList2.add( new PE(1, rating) );
peList2.add( new PE(2, rating) );
peList2.add( new PE(3, rating) );
//System.out.println("Creates 4 PEs with same MIPS Rating and put
them"+
//      " into the PE list");

mList.add( new Machine(1, peList2) ); // Second Machine
//System.out.println("Creates the 2nd Machine that has 4 PEs and
" +
//      "stores it into the Machine list");
//System.out.println();

PEList peList3 = new PEList();
//System.out.println("Creates a PE list for the 3rd Machine");

peList3.add( new PE(0, rating) );
peList3.add( new PE(1, rating) );
//System.out.println("Creates 2 PEs with same MIPS Rating and put
them"+
//      " into the PE list");

mList.add( new Machine(2, peList3) ); // Third Machine
//System.out.println("Creates the 3rd Machine that has 2 PEs and
" +
//      "stores it into the Machine list");
//System.out.println();

// 6. Create a ResourceCharacteristics object that stores the
//      properties of a Grid resource: architecture, OS, list of
//      Machines, allocation policy: time- or space-shared, time
zone
//      and its price (G$/PE time unit).
String arch = "Sun Ultra"; // system architecture
String os = "Solaris"; // operating system
double time_zone = 9.0; // time zone this resource
located

```

```

        double cost = 3.0;                // the cost of using this
resource
        ResourceCharacteristics resConfig = new ResourceCharacteristics(
            arch, os, mList, ResourceCharacteristics.TIME_SHARED,
            time_zone, cost);

        //System.out.println("Creates the properties of a Grid resource
and " +
        //      "stores the Machine list");

        // 7. Finally, we need to create a GridResource object.
        long seed = 11L*13*17*19*23+1;
        double peakLoad = 0.0;           // the resource load during peak
hour
        double offPeakLoad = 0.0;       // the resource load during off-
peak hr
        double holidayLoad = 0.0;       // the resource load during holiday

        // incorporates weekends so the grid resource is on 7 days a week
        LinkedList Weekends = new LinkedList();
        Weekends.add(new Integer(Calendar.SATURDAY));
        Weekends.add(new Integer(Calendar.SUNDAY));

        // incorporates holidays. However, no holidays are set in this
example
        LinkedList Holidays = new LinkedList();
        GridResource gridRes = null;
        try
        {
            // creates a GridResource with a link
            gridRes = new GridResource(name,
                new SimpleLink(name + "_link", baud_rate, delay, MTU),
                seed, resConfig, peakLoad, offPeakLoad, holidayLoad,
                Weekends, Holidays);
        }
        catch (Exception e) {
            e.printStackTrace();
        }

        System.out.println("Finally, creates one Grid resource (name: " +
name +
            " - id: " + gridRes.get_id() + ")");
        System.out.println();

        return gridRes;
    }

    /**
     * Prints the Gridlet objects
     */
    private static void printGridletList(GridletList list, String name,
        boolean detail)

```



```

{
    int size = list.size();
    Gridlet gridlet = null;

    String indent = "    ";
    System.out.println();
    System.out.println("===== OUTPUT for " + name + "
=====");
    System.out.println("Gridlet ID" + indent + "STATUS" + indent +
        "Resource ID" + indent + "Cost");

    // a loop to print the overall result
    int i = 0;
    for (i = 0; i < size; i++)
    {
        gridlet = (Gridlet) list.get(i);
        System.out.print(indent + gridlet.getGridletID() + indent
            + indent);

        System.out.print( gridlet.getGridletStatusString() );

        System.out.println( indent + indent + gridlet.getResourceID()
+
            indent + indent + gridlet.getProcessingCost() );
    }

    if (detail == true)
    {
        // a loop to print each Gridlet's history
        for (i = 0; i < size; i++)
        {
            gridlet = (Gridlet) list.get(i);
            System.out.println( gridlet.getGridletHistory() );

            System.out.print("Gridlet #" + gridlet.getGridletID() );
            System.out.println(", length = " +
gridlet.getGridletLength()
                + ", finished so far = " +
                gridlet.getGridletFinishedSoFar() );

            System.out.println("=====\n");
        }
    }
}

} // end class

```


XIII. Glosario⁵¹

Acceso a Internet de Banda Ancha

Incluye conexiones a Internet de alta velocidad generalmente se refiere a la tecnología de Cable MODEM o de ADSL (Asymmetric Digital Subscriber Line).

ADSL Asymmetric Digital Subscriber Line

ADSL son las siglas de Asymmetric Digital Subscriber Line ("Línea de Abonado Digital Asimétrica"). Consiste en una línea digital de alta velocidad, apoyada en el par trenzado de cobre que lleva la línea telefónica convencional o línea de abonado.

Se trata de una tecnología de acceso a Internet de banda ancha, lo que implica capacidad para transmitir más datos, lo que, a su vez, se traduce en mayor velocidad.

Esto se consigue mediante la utilización de una banda de frecuencias más alta que la utilizada en el teléfono convencional (300-3.400 Hz) por lo que, para disponer de ADSL, es necesaria la instalación de un filtro (llamado splitter o discriminador) que se encarga de separar la señal telefónica convencional de la que usaremos para conectarnos con ADSL.

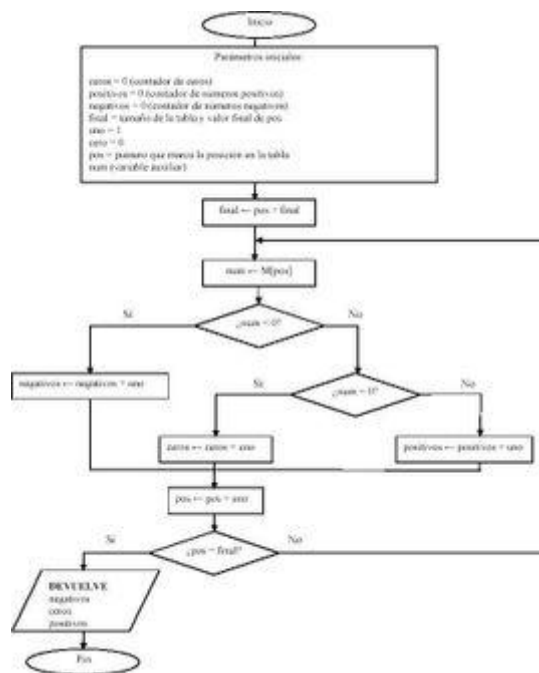
Esta línea se denomina asimétrica debido a que la velocidad de bajada y de subida de datos (entendiéndose por bajada la llegada de datos al usuario, y subida el envío de datos del usuario hacia la Red) no coinciden. Normalmente, la velocidad de bajada es mayor que la de subida.

En una línea ADSL se establecen tres canales de comunicación, que son el de envío de datos, el de recepción de datos y el de servicio telefónico normal.

⁵¹ Definiciones tomadas de Wikipedia en español

Algoritmo

Un algoritmo es un conjunto finito de instrucciones o pasos que sirven para ejecutar una tarea o resolver un problema. De un modo más formal, un algoritmo es una secuencia finita de operaciones realizables, no ambiguas, cuya ejecución da una solución de un problema en un tiempo finito.



Los Diagrama de flujo se usan habitualmente para representar algoritmos.

El término algoritmo no está exclusivamente relacionado con las matemáticas, ciencias de la computación o informática. En realidad, en la vida cotidiana empleamos algoritmos en multitud de ocasiones para resolver diversos problemas. Algunos ejemplos son el uso de una lavadora (se siguen las instrucciones), o la preparación de una comida (siguiendo los pasos de una receta). También existen ejemplos de índole matemática, como el algoritmo de la división para calcular el cociente de dos números, el algoritmo de Euclides para

calcular el máximo común divisor de dos enteros positivos, o incluso el método de Gauss para resolver sistemas de ecuaciones.

La definición de algoritmo aún no cuenta con la formalidad científica que podría ser ideal para ciencias como las matemáticas y las ciencias de la computación (donde los algoritmos son esenciales pero a falta de formalidad no pueden incluirse fácilmente en las demostraciones formales de estas ciencias). Sin embargo, si existe un concepto intuitivo de algoritmo.

Un algoritmo es un sistema por el cual se llega a una solución, teniendo en cuenta que debe de ser definido, finito y preciso. Por preciso entendemos que cada paso a seguir tiene un orden; finito implica que tiene un determinado número de pasos, o sea, que tiene un fin; y definido, que si se sigue el mismo proceso más de un vez llegaremos al mismo resultado.

Estructura Básica:

- inicio
- datos de entrada (operaciones básicas)
- procesamiento de los datos
- datos de salida
- fin

El análisis y estudio de los algoritmos es una disciplina de las ciencias de la computación, y en la mayoría de los casos, su estudio es completamente abstracto sin usar ningún tipo de lenguaje de programación ni cualquier otra implementación; por eso, en ese sentido, comparte las características de las disciplinas matemáticas. Así, el análisis de los algoritmos se centra en los principios básicos del algoritmo, no en los de la implementación particular. Una forma de plasmar (o algunas veces codificar) un algoritmo es escribirlo en

pseudocódigo o utilizar un lenguaje muy simple tal como Lexico cuyos códigos pueden estar en el idioma del programador.

Algunos escritores restringen la definición de algoritmo a procedimientos que deben acabar en algún momento, mientras que otros consideran procedimientos que podrían ejecutarse eternamente sin pararse, suponiendo el caso en el que existiera algún dispositivo físico que fuera capaz de funcionar eternamente. En este último caso, la finalización con éxito del algoritmo no se podría definir como la terminación de éste con una salida satisfactoria, sino que el éxito estaría definido en función de las secuencias de salidas dadas durante un periodo de vida de la ejecución del algoritmo. Por ejemplo, un algoritmo que verifica que hay más ceros que unos en una secuencia binaria infinita debe ejecutarse siempre para que pueda devolver un valor útil. Si se implementa correctamente, el valor devuelto por el algoritmo será válido, hasta que evalúe el siguiente dígito binario.

De esta forma, mientras evalúa la siguiente secuencia podrán leerse dos tipos de señales: una señal positiva (en el caso de que el número de ceros es mayor que el de unos) y una negativa en caso contrario. Finalmente, la salida de este algoritmo se define como la devolución de valores exclusivamente positivos si hay más ceros que unos en la secuencia, y en cualquier otro caso, devolverá una mezcla de señales positivas y negativas.

Arquitectura de software

En los inicios de la informática, la programación se consideraba un arte, debido a la dificultad que entrañaba para la mayoría de los mortales, pero con el tiempo se han ido desarrollando metodologías y fórmulas o trucos para conseguir nuestros propósitos. Y a todas estas técnicas se les ha dado en llamar Arquitectura Software.

Una Arquitectura Software, también denominada Arquitectura lógica, consiste en un conjunto de patrones y abstracciones coherentes que proporcionan el marco de referencia necesario para guiar la construcción del software para un sistema de información.

La arquitectura software establece los fundamentos para que analistas, diseñadores, programadores, etc. trabajen en una línea común que permita alcanzar los objetivos y necesidades del sistema de información.

Una arquitectura software se selecciona y diseña con base en unos objetivos y restricciones. Los objetivos son aquellos prefijados para el sistema de información, pero no solamente los de tipo funcional, también otros objetivos como la mantenibilidad, auditabilidad, flexibilidad e interacción con otros sistemas de información. Las restricciones son aquellas limitaciones derivadas de las tecnologías disponibles para implementar sistemas de información. Unas arquitecturas son más recomendables de implementar con ciertas tecnologías mientras que otras tecnologías no son aptas para determinadas arquitecturas. Por ejemplo, no es viable emplear una arquitectura software de tres capas para implementar sistemas en tiempo real.

La arquitectura software define, de manera abstracta, los componentes que llevan a cabo alguna tarea de computación, sus interfaces y la comunicación ente ellos. Toda arquitectura software debe ser implementable en una arquitectura física, que consiste simplemente en determinar qué computadora tendrá asignada cada tarea de computación.

La arquitectura de software, tiene que ver con el diseño y la implementación de estructuras de software de alto nivel. Es el resultado de ensamblar un cierto número de elementos arquitectónicos de forma adecuada para satisfacer la mayor funcionalidad y requerimientos de desempeño de un sistema, así como requerimientos no funcionales, como la confiabilidad, escalabilidad, portabilidad, y disponibilidad. —Kruchten, Philippe

Cable MODEM

Un cablemódem es un tipo especial de módem diseñado para modular la señal de datos sobre una infraestructura de televisión por cable. El término Internet por cable (o

simplemente cable) se refiere a la distribución de un servicio de conectividad a Internet sobre esta infraestructura de telecomunicaciones.

Cablemódem Motorola SurfBoard Los cablemodems no deben confundirse con antiguos sistemas LAN como 10base2 o 10base5 que utilizaban cables coaxiales -- y especialmente con 10broad36, el cual realmente utiliza el mismo tipo de cable que los sistemas CATV.

Los cablemodems se utilizan principalmente para distribuir el acceso a Internet de banda ancha, aprovechando el ancho de banda que no se utiliza en la red de TV por cable.

Los abonados de un mismo vecindario comparten el ancho de banda proporcionado por una única línea de cable coaxial. Por lo tanto, la velocidad de conexión puede variar dependiendo de cuanta gente este usando el servicio al mismo tiempo.

A menudo, la idea de una línea compartida se considera como un punto débil de la conexión a Internet por cable. Desde un punto de vista técnico, todas las redes, incluyendo los servicios DSL, comparten una cantidad fija de ancho de banda entre multitud de usuarios -- pero ya que las redes de cable tienden a abarcar áreas más grandes que los servicios DSL, se debe tener más cuidado para asegurar un buen rendimiento en la red.

Una debilidad más significativa de las redes de cable al usar una línea compartida es el riesgo de la pérdida de privacidad, especialmente considerando la disponibilidad de herramientas de hacking para cablemódems. De este problema se encarga el cifrado de datos y otras características de privacidad especificadas en el estándar DOCSIS ("Data Over Cable Service Interface Specification"), utilizado por la mayoría de cablemodems.

Existen dos estandares:

El DOCSIS y el EURODOCSIS mayormente utilizado en europa

Calendarizador

El calendarizador es un componente funcional muy importante de los sistemas operativos multitarea y multiproceso, y es esencial en los sistemas operativos de tiempo real. Su función consiste en repartir el tiempo disponible de un microprocesador entre todos los procesos que están disponibles para su ejecución.

Supóngase un ordenador que contiene un único microprocesador. Dicho microprocesador solamente puede ejecutar un programa en cada instante de tiempo. Además, cuando un programa está ejecutandose, nunca dejará de hacerlo por sí mismo. De manera que, en principio, cualquier programa monopoliza el microprocesador impidiendo que otros programas se ejecuten.

Por ello, la primera misión de un planificador es expulsar el programa en ejecución cuando decida que es pertinente. Esto se consigue de dos maneras, siempre con ayuda del propio hardware:

- Cuando expira un temporizador, que se activa a intervalos regulares de tiempo. En intervalos muy cortos, generalmente cada 250 milisegundos.
- Cuando el programa solicita una operación de entrada/salida. Dado que el programa no puede continuar hasta que termine dicha operación, es un buen momento para ejecutar otro programa.

En ambos casos, el control del microprocesador pasa a manos del calendarizador gracias a que el hardware genera una interrupción. En este proceso de expulsión, se salva el estado de ejecución del programa (programa y su estado se denomina proceso).

A continuación, el calendarizador decide cuál será el siguiente proceso en ejecutarse. Naturalmente, solamente se escogen procesos que estén listos para hacerlo. Si un proceso sigue esperando por una operación de entrada/salida no será candidato a ejecutarse hasta que finalice tal operación.

La selección del proceso sigue alguna política de planificación preestablecida. Una vez seleccionado un proceso, se procede a ejecutarlo. Para ello, el planificador restaura su estado de ejecución (previamente salvado) y abandona el uso del microprocesador cediéndoselo a dicho proceso.

Todo esto apenas dura unos pocos milisegundos.

Gracias a que el tiempo del microprocesador se reparte entre todos los procesos a intervalos muy cortos, el ordenador ofrece la sensación de que todos los procesos están ejecutándose a la vez.

Cuando un ordenador tiene varios microprocesadores (arquitectura NUMA y SMP), este esquema se repite para cada microprocesador.

Banda ancha

Se refiere al término técnico específico que deriva del inglés broadband. Para el método de acceso a Internet comunmente conocido como Banda Ancha véase Acceso a Internet de Banda Ancha.

En un sentido general, Banda Ancha se refiere a la transmisión de datos en el cual se envían simultáneamente varias piezas de información, con el objeto de incrementar la velocidad de transmisión efectiva. En ingeniería de redes este término se utiliza también para los métodos en donde dos o más señales comparten un medio de transmisión.

Algunas de las variantes de los servicios de línea de abonado digital (del inglés Digital Subscriber Line, DSL) son de banda ancha en el sentido en que la información se envía sobre un canal y la voz por otro canal, pero compartiendo el mismo par de cables. Los modems analógicos que operan con velocidades mayores a 600 bps también son técnicamente banda ancha, pues obtienen velocidades de transmisión efectiva mayores usando muchos canales en donde la velocidad de cada canal se limita a 600 baudios. Por ejemplo, un modem de 2400 bps usa cuatro canales de 600 baudios. Este método de transmisión contrasta con la transmisión en banda base, en donde un tipo de señal usa todo el ancho de banda del medio de transmisión, como por ejemplo Ethernet 100BASE-T.

Tecnologías de banda ancha

- Cablemódem
- DSL
- GPRS y UMTS (no suele ser con tarifa plana - de tiempo y megas -).
- Internet por satélite
- LMDS para zonas rurales y su derivación portátil para ciudad: WiMAX
- Power line communication (PLC)
- WiFi

Calendarización Round-robin

Round robin es un método para seleccionar todos los elementos en un grupo de manera equitativa y en un orden racional, normalmente comenzando por el primer elemento de la lista hasta llegar al último y empezando de nuevo desde el primer elemento.

Una forma sencilla de entender el round robin es imaginar una secuencia para "tomar turnos". En operaciones computacionales, un método para ejecutar diferentes procesos de manera simultánea, para la utilización equitativa de los recursos del equipo, es limitando cada proceso a un pequeño periodo de tiempo, y luego suspendiendo éste proceso para dar

oportunidad a otro proceso y así sucesivamente. A esto se le denomina comúnmente como Calendarización Round-Robin.

Ciencias de la computación

Las ciencias de la computación abarcan el estudio de las bases teóricas de la información y la computación y su implementación y aplicación en sistemas computacionales. Existen diversos campos dentro de la disciplina de las ciencias de la computación; algunos enfatizan los resultados específicos del cómputo (como los gráficos por computadora), mientras que otros (como la teoría de la complejidad computacional) se relacionan con propiedades de los algoritmos usados al realizar cómputos. Otros por su parte se enfocan en los problemas que requieren la implementación de cómputos. Por ejemplo, los estudios de la teoría de lenguajes de programación describen un cómputo, mientras que la programación de computadoras aplica lenguajes de programación específicos para desarrollar una solución a un problema computacional concreto.

Ciclos de procesamiento

El ciclo de procesamiento de información que realiza una computadora se puede resumir en las siguientes cuatro funciones básicas:

1. Aceptar entradas.

Una entrada es cualquier tipo de dato que introducimos a la computadora como: palabras y símbolos de un documento, números para realizar cálculos, instrucciones para realizar procesos, imágenes, sonidos, etc. Se utilizan diferentes dispositivos como el teclado, ratón y/o micrófono para que la computadora reciba las entradas; otros dispositivos pueden ser los lápices ópticos, scanner, cámaras, pantallas touch screen, entre otros.

2. Procesar datos.

Consiste en manipular los datos de diferentes maneras para: realizar cálculos, modificar palabras e imágenes, ordenar o alterar el orden de los datos. La computadora utiliza un dispositivo que se llama Unidad de Procesamiento.

Esta unidad contiene dos elementos principales:

- a) Unidad Central de Procesamiento (en Inglés: Central Processing Unit, CPU).
- b) Memoria del Sistema.

3. Almacenar datos e información.

La computadora debe almacenar datos para que estén disponibles durante el procesamiento. El lugar donde se almacenarán depende de cómo se utilicen los datos. Por ejemplo, en la Memoria del Sistema (dentro de la Unidad de Procesamiento) la computadora guarda lo que está usando. Hay otro tipo de almacenamiento, que es más permanente, en donde se guarda lo que no se necesita para procesar en ese momento (por ejemplo: disquetes, CD, disco duro).

4. Producir salidas.

Son los resultados generados por la computadora. Algunos ejemplos de resultados son: reportes, documentos, gráficas, imágenes, etc. Se utilizan dispositivos como el monitor, impresora, bocinas.

Probablemente, la pregunta que viene a la mente ahora es: ¿cómo sabe una computadora lo que tiene que hacer? Existen dos elementos esenciales en las computadoras, el hardware que son los componentes físicos y el software que son los programas. Sin software, lo único que podemos hacer con una computadora es encenderla y apagarla. El software es el conjunto de programas, o series de instrucciones, que le indican a la computadora cómo

debe realizar tareas específicas, por ejemplo: cómo debe interactuar con el usuario, cómo debe procesar los datos, etc.

Cluster de computadores

Cluster es un grupo de múltiples ordenadores unidos, mediante una red de alta velocidad, de tal forma que el conjunto es visto como un único ordenador más potente por los usuarios y las aplicaciones. Se espera de un cluster que presente combinaciones de los siguientes servicios:

- Alto rendimiento (High Performance)
- Alta disponibilidad (High Availability)
- Equilibrio de carga (Load Balancing)
- Escalabilidad (Scalability)

Los ordenadores pueden tener todos la misma configuración de Hardware y Sistema Operativo (cluster homogéneo), diferente rendimiento pero con arquitecturas y sistemas operativos similares (cluster semi-homogéneo) o tener diferente hardware y sistema operativo (cluster heterogéneo), lo que hace más fácil y económica su construcción.

Para que un cluster funcione como tal, no basta solo con conectar entre sí los ordenadores, sino que es necesario proveer un sistema de manejo del cluster, el cual se encargue de interactuar con el usuario y los procesos que corren en él para optimizar el funcionamiento.

Código ejecutable

Código ejecutable corresponde a las unidades de programas, donde el ordenador puede realizar las instrucciones compiladas mediante el compilador y el enlazador de librerías.

Generalmente se confunde con el código objeto, ya que al leer su estructura se comprende como símbolos. Pero en realidad, este código se encuentra empaquetado y listo para ser ejecutado en cualquier computadora. Generalmente vienen con la extensión EXE o COM, si son para ser ejecutados en computadoras con Sistema Operativo de Windows o con bits de marca que trae Linux para ser ejecutable.

Código fuente

El código fuente es un conjunto de líneas de código que conforman un bloque de texto que normalmente genera otro código mediante un compilador o intérprete para ser ejecutado por una computadora.

Normalmente se refiere a la programación de software. El programador escribe el código fuente en el lenguaje de programación elegido. Posteriormente en un proceso de compilación el código fuente se traduce en código objeto.

Componentes de un Cluster

Un cluster necesita de varios componentes de software y hardware para poder funcionar. A saber:

- Nodos (los ordenadores)
- Sistemas Operativos
- Conexiones de Red
- Middleware (capa de abstracción entre el usuario y los sistemas operativos)
- Protocolos de Comunicación y servicios.
- Aplicaciones (pueden ser paralelas o no)

Computación concurrente

La computación concurrente es la simultaneidad en la ejecución de múltiples tareas interactivas. Estas tareas pueden ser un conjunto de procesos o hilos de ejecución creados por un único programa. Las tareas se pueden ejecutar en una sola unidad central de proceso (multiprogramación), en varios procesadores o en una red de computadores distribuidos. La programación concurrente está relacionada con la programación paralela, pero enfatiza más la interacción entre tareas. Así, la correcta secuencia de interacciones o comunicaciones entre los procesos y el acceso coordinado de recursos que se comparten por todos los procesos o tareas son las claves de esta disciplina.

Los pioneros en este campo fueron Edsger Dijkstra, Per Brinch Hansen, y C. A. R. Hoare.

Computación de ciclos redundantes

El modelo de computación de ciclos redundantes, también conocido como computación zombi, es el empleado por aplicaciones como Seti@Home, consistente en que un servidor o grupo de servidores distribuyen trabajo de procesamiento a un grupo de computadoras voluntarias a ceder capacidad de procesamiento no utilizada. Básicamente, cuando dejamos nuestro ordenador encendido, pero sin utilizarlo, la capacidad de procesamiento se desperdicia por lo general en algún protector de pantalla, este tipo de procesamiento distribuido utiliza nuestra computadora cuando nosotros no la necesitamos, aprovechando al máximo la capacidad de procesamiento.

Al día de la fecha, ya hay virus que utilizan esta misma técnica para hacer envíos de correo electrónico masivo (spam) desde ordenadores en todo el mundo, sin el conocimiento de los usuarios.

Computación distribuida

La computación distribuida, computación en grilla o informática en rejilla, es un nuevo modelo para resolver problemas de computación masiva utilizando un gran número de

ordenadores organizados en racimos incrustados en una infraestructura de telecomunicaciones distribuida.

La informática en grilla consiste en compartir recursos heterogéneos (basadas en distintas plataformas, arquitecturas de equipos y programas, lenguajes de programación), situados en distintos lugares y pertenecientes a diferentes dominios de administración sobre una red que utiliza estándares abiertos. Dicho brevemente, consiste en virtualizar los recursos informáticos.

La computación en grilla ha sido diseñada para resolver problemas demasiado grandes para cualquier simple (Supercomputadora), mientras se mantiene la flexibilidad de trabajar en múltiples problemas más pequeños. Por lo tanto, la computación en grid es naturalmente un entorno multi-usuario; por ello, las técnicas de autorización segura son esenciales antes de permitir que los recursos informáticos sean controlados por usuarios remotos.

En términos de funcionalidad, las grillas se clasifican en computacionales (incluyendo grillas de barrido de la CPU) y en de datos.

La herramienta Globus ha emergido como el estándar de facto para la capa intermedia (middleware) de la grilla. Globus tiene recursos para manejar:

- La gestión de recursos (Protocolo de Gestión de Recursos en Rejilla o Grid Resource Management Protocol)
- Servicios de Información (Servicio de Descubrimiento y Monitorización o Monitoring and Discovery Service)
- Gestión y Movimiento de Datos (Acceso Global al Almacenamiento Secundario, Global Access to secondary Storage y FTP en grilla, GridFTP)

La mayoría de grillas que se expanden sobre las comunidades académicas y de investigación de Norteamérica y Europa están basadas en las herramienta Globus Toolkit como núcleo de la capa intermedia.

Conexiones de Red

Pueden conectarse mediante una simple red ethernet con placas comunes, o utilizarse tecnologías especiales de alta velocidad como Fast Ethernet, Gigabit Ethernet, Mirynet, SCI, etc.

Clustering

Otro método para crear sistemas de supercomputadoras es el clustering. Un cluster o racimo de computadoras consiste en un grupo de computadoras de relativo bajo costo conectadas entre sí mediante un sistema de red de alta velocidad (gigabit de fibra óptica por lo general) y un software que realiza la distribución de la carga de trabajo entre los equipos. Por lo general, éste tipo de sistemas cuentan con un centro de almacenamiento de datos único.

Cluster de computadores

En un cluster todos los nodos se encuentran en el mismo lugar, conectados por una red local y así englobar todos lo recursos, en cambio en un grid no tienen porque estar en el mismo espacio geográfico, pueden estar en diferentes puntos del mundo.

También suele presentarse que GRID se le llama cuando el resultado obtenido de el englobe de las maquinas da una supercomputadora, con un cluster solo se busca mejorar el rendimiento de las maquinas englobándolas en una sola.

DB2

DB2 es una marca comercial, propiedad de IBM, bajo la cual se comercializa el sistema de gestión de base de datos relacional de dicha compañía, así como otras herramientas relacionadas encuadradas en la misma línea de producto.

Incluye todo lo necesario para implementar una replicación de datos, permitiendo realizar gestiones sobre Bases de Datos y garantizando un cierto nivel de seguridad respecto a autorización y autenticación. Permite utilizar la conectividad que ofrece Java en su paquete JDBC.

Muchos expertos de la industria y usuarios han elogiado las nuevas herramientas que IBM desarrollo para facilitar la administración y uso de DB2 Universal Database, constituido en base a dos productos incluidos en el DB2 de AIX en 1994: el DB2 Common Server, que para propósitos generales incluía funciones avanzadas para el mercado de servidores de bases de datos con soporte de hardware SMP y OLTP; y el DB2 Parallel Edition, que fue desarrollado para soportar aplicaciones de gran escala, como el Data Warehousing y Data Minino y aplicaciones de negocios a nivel mundial como la SAP, People Soft y Baan.

DB2 incluye todo lo necesario para implementar una solución de replicación de datos en cualquier tipo fr ambiente distribuido o heterogéneo, pues permite enviar los datos a cualquier sitio para cubrir todos los requerimientos de una empresa, desde oficinas centrales a sucursales, usuarios móviles, proveedores, clientes y socios de negocios.

Gracias a su alcance global y de bajo costo, Internet puede ser una solución de negocios muy poderosa para realizar operaciones comerciales garantizando un nivel de seguridad y confiabilidad con sus servicios de autorización y autenticación integrados a redes y sistema operativos, soportando el network-computing utilizando Java y JDBC, incluyendo capacidad nativa de almacenar varios tipos de datos: alfanuméricos, video, imagen, audio y los definidos por el usuario.

DB2 y SQL Server tenían grandes compañías detrás con otros negocios que les permitió aguantar la política agresiva de Oracle. Recientemente IBM adquirió Informix con lo que el mercado de las bases de datos comerciales en UNIX (Linux) quedó entre IBM y Oracle.

ERP

Los Sistemas de Planeación de Recursos Empresariales (ERPs) son Sistemas de Información Gerencial que integran y manejan muchas de las practicas de los negocios asociados con las operaciones de producción y de los aspectos de distribución de una compañía comprometida en la manufactura de productos o servicios.

La Planeación de Recursos Empresariales es un término derivado de la Planeación de Recursos de Manufactura (MRPII) que seguido de la Planeación de Requerimientos de Material (MRP). Los sistemas ERP típicamente manejan la manufactura, logística, distribución, inventario, envíos, facturas y una contabilidad para la compañía de la Planeación de Recursos Empresariales o el software ERP puede intervenir en el control de muchas actividades de negocios como ventas, entregas, pagos, producción, administración de inventarios, calidad de administración y la administración de recursos humanos.

Los sistemas ERP's ocasionalmente son llamados "oficinas de respaldo" ya que indican que el cliente y el público general no están directamente involucrados. Este sistema es, en contraste con el sistema de apertura de datos (front office), que crea una relación administrativa del consumidor o servicio al consumidor (CRM), un sistema que trata directamente con los clientes, o con los sistemas de negocios electrónicos tales como comercio electrónico, gobierno electrónico, telecomunicaciones electrónicas y finanzas electrónicas; así mismo, es un sistema que trata directamente con los proveedores, no estableciendo únicamente una relación administrativa con ellos (SRM).

ERP's están funcionando ampliamente a través de las empresas. Todos los departamentos funcionales que están involucrados en la operación o producción están integrados en un solo sistema. En adición a la manufactura, almacenamiento, logística e información tecnológica, a las que se le podrían incluir la contabilidad, recursos humanos, mercadotecnia y administración estratégica.

Framework

En el desarrollo de software, un framework es una estructura de soporte definida en la cual otro proyecto de software puede ser organizado y desarrollado. Típicamente, un framework puede incluir soporte de programas, librerías y un lenguaje de scripting entre otros softwares para ayudar a desarrollar y unir los diferentes componentes de un proyecto.

Un framework representa una arquitectura de software que modela las relaciones generales de las entidades del dominio. Provee una estructura y "manera de trabajo" la cual extienden o utilizan las aplicaciones del dominio.

Globus Toolkit

La herramienta Globus ha emergido como el estándar de facto para la capa intermedia (middleware) de la informática distribuida. Globus tiene recursos para administrar:

- La gestión de recursos(Protocolo de Gestión de Recursos en Rejilla o Grid Resource Management Protocol)
- Servicios de Información (Servicio de Descubrimiento y Monitorización o Monitoring and Discovery Service)
- Gestión y Movimiento de Datos (Acceso Global al Almacenamiento Secundario, Global Access to secondary Storage y FTP en grilla, GridFTP)

La mayoría de grillas que se expanden sobre las comunidades académicas y de investigación de Norteamérica y Europa están basadas en las herramientas Globus Toolkit como núcleo de la capa intermedia.

Grid

La computación en grid o en malla es un nuevo paradigma de computación distribuida en el cual todos los recursos de un número indeterminado de computadoras son englobados para ser tratados como un único superordenador de manera transparente.

Estas computadoras englobadas no están conectadas o enlazadas firmemente, es decir no tienen porque estar en el mismo lugar geográfico. Se puede tomar como ejemplo el proyecto SETI@Home, en el cual trabajan computadoras alrededor de todo el planeta para buscar vida extraterrestre.

Grid computing

Grid computing es una tecnología innovadora que permite utilizar de forma coordinada todo tipo de recursos (entre ellos computo, almacenamiento y aplicaciones específicas) que no están sujetos a un control centralizado. En este sentido es una nueva forma de computación distribuida, en la cual los recursos pueden ser heterogéneos (diferentes arquitecturas, supercomputadores, clusters, ...) y se encuentran conectados mediante redes de área extensa (por ejemplo Internet). Desarrollado en ámbitos científicos a principios de los 90, su entrada al mercado comercial siguiendo la idea de la llamada Utility computing supone una revolución que dará mucho que hablar.

El término grid se refiere a una infraestructura que permite la integración y el uso colectivo de ordenadores de alto rendimiento, redes y bases de datos que son propiedad y están administrados por diferentes instituciones. Puesto que la colaboración entre instituciones envuelve un intercambio de datos, o de tiempo de computación, el propósito del grid es

facilitar la integración de recursos computacionales. Universidades, laboratorios de investigación, empresas, etc. se asocian para formar grid para lo cual utilizan algún tipo de software que implemente este concepto.

Las características de esta arquitectura serían:

- Capacidad de balanceo de sistemas: No habría necesidad de calcular la capacidad de los sistemas en función de los picos de trabajo, ya que la capacidad se puede reasignar desde la granja de recursos a donde se necesite;
- Alta disponibilidad. Con la nueva funcionalidad, si un servidor falla, se reasignan los servicios en los servidores restantes;
- Reducción de costes: Con esta arquitectura los servicios son gestionados por "granjas de recursos". Ya no es necesario disponer de "grandes servidores" y podremos hacer uso de componentes de bajo coste. Cada sistema puede ser configurado siguiendo el mismo patrón;

Se relaciona el concepto de grid con la nueva generación de Internet. El nuevo protocolo de Internet IPv6 permitirá trabajar con una Internet más rápida y accesible. Una de las ideas clave en la superación de las limitaciones actuales de Internet IPv4 es la aparición de nuevos niveles de servicio que harán uso de la nueva capacidad de la red para intercomunicar los ordenadores.

Este avance en la comunicación permitirá el avance de las ideas de grid computing al utilizar como soporte la altísima conectividad de Internet. Es por ello que uno de los campos de mayor innovación en el uso del grid computing, fuera de los conceptos de supercomputación, es el desarrollo de un estándar para definir los Grid Services frente a los actuales Web Services.

Hardware

Se denomina hardware o soporte físico al conjunto de elementos materiales que componen un ordenador. Hardware también son los componentes físicos de una computadora tales como el disco duro, CD-Rom, floppy, etc.. En dicho conjunto se incluyen los dispositivos electrónicos y electromecánicos, circuitos, cables, tarjetas, armarios o cajas, periféricos de todo tipo y otros elementos físicos.

El hardware se refiere a todos los componentes físicos (que se pueden tocar) de la computadora: discos, unidades de disco, monitor, teclado, ratón (mouse), impresora, placas, chips y demás periféricos. En cambio, el software es intangible, existe como ideas, conceptos, símbolos, pero no tiene sustancia. Una buena metáfora sería un libro: las páginas y la tinta son el hardware, mientras que las palabras, oraciones, párrafos y el significado del texto son el software. Una computadora sin software sería tan inútil como un libro con páginas en blanco.

Informática distribuida

Refiérase a Computación distribuida.

Infraestructura**IPv6**

IPv6 es la versión 6 del Protocolo de Internet (Internet Protocol), un estándar del nivel de red encargado de dirigir y encaminar los paquetes a través de una red.

Diseñado por Steve Deering de Xerox PARC y Craig Mudge, IPv6 está destinado a sustituir al estándar IPv4, cuyo límite en el número de direcciones de red admisibles está empezando a restringir el crecimiento de Internet y su uso, especialmente en China, India, y otros países asiáticos densamente poblados. Pero el nuevo estándar mejorará el servicio

globalmente; por ejemplo, proporcionando a futuras celdas telefónicas y dispositivos móviles con sus direcciones propias y permanentes. Al día de hoy se calcula que las dos terceras partes de las direcciones que ofrece IPv4 ya están asignadas.

IPv4 soporta 4.294.967.296 ($4,294 \times 10^9$) direcciones de red diferentes, un número inadecuado para dar una dirección a cada persona del planeta, y mucho menos para cada coche, teléfono, PDA o tostadora; mientras que IPv6 soporta 340.282.366.920.938.463.463.374.607.431.768.211.456 ($3,4 \times 10^{38}$ ó 340 sextillones) direcciones -- cerca de $4,3 \times 10^{20}$ (430 trillones) direcciones por cada pulgada cuadrada ($6,7 \times 10^{17}$ ó 670 mil billones direcciones/mm²) de la superficie de La Tierra.

Adoptado por el Internet Engineering Task Force en 1994 (cuando era llamado "IP Next Generation" o IPng), IPv6 cuenta con un pequeño porcentaje de las direcciones públicas de Internet, que todavía están dominadas por IPv4. La adopción de IPv6 ha sido frenada por la traducción de direcciones de red (NAT), que alivia parcialmente el problema de la falta de direcciones IP. Pero NAT hace difícil o imposible el uso de algunas aplicaciones P2P, como son la voz sobre IP (VoIP) y juegos multiusuario. Además, NAT rompe con la idea originaria de Internet donde todos pueden conectarse con todos. Actualmente, el gran catalizador de IPv6 es la capacidad de ofrecer nuevos servicios, como la movilidad, Calidad de Servicio (QoS), privacidad, etc. El gobierno de los Estados Unidos ha ordenado el despliegue de IPv6 por todas sus agencias federales para el año 2008.

Se espera que IPv4 se siga soportando hasta por lo menos el 2025, dado que hay muchos dispositivos heredados que no se migrarán a IPv6 nunca y que seguirán siendo utilizados por mucho tiempo.

IPv6 es la segunda versión del Protocolo de Internet que se ha adoptado para uso general. También hubo un IPv5, pero no fue un sucesor de IPv4; mejor dicho, fue un protocolo experimental orientado al flujo de streaming que intentaba soportar voz, video y audio.

Linux

Linux es la denominación de un sistema operativo y el nombre de un núcleo. Es uno de los paradigmas del desarrollo de software libre (y de código abierto), donde el código fuente está disponible públicamente y cualquier persona, con los conocimientos informáticos adecuados, puede libremente usarlo, modificarlo y redistribuirlo.

El término Linux estrictamente se refiere al núcleo Linux, pero es más comúnmente utilizado para describir al sistema operativo tipo Unix (que implementa el estándar POSIX), que utiliza primordialmente filosofía y metodologías libres (también conocido como GNU/Linux) y que está formado mediante la combinación del núcleo Linux con las bibliotecas y herramientas del proyecto GNU y de muchos otros proyectos/grupos de software (libre o no libre). El núcleo no es parte oficial del proyecto GNU (el cual posee su propio núcleo en desarrollo, llamado Hurd), pero es distribuido bajo los términos de la licencia GPL (GNU General Public License).

La expresión Linux también es utilizada para referirse a las distribuciones Linux, colecciones de software que suelen contener grandes cantidades de paquetes además del núcleo. El software que suelen incluir consta de una enorme variedad de aplicaciones, como: entornos gráficos, suites ofimáticas, servidores web, servidores de correo, servidores FTP, etcétera. Coloquialmente se aplica el término Linux a éstas, aunque en estricto rigor sea incorrecto, dado que la distribución es la forma más simple y popular para obtener un sistema Linux.

La marca Linux (Número de serie: 1916230) pertenece a Linus Torvalds y se define como "un sistema operativo para computadoras que facilita su uso y operación".

Desde su lanzamiento, Linux ha incrementado su popularidad en el mercado de servidores. Su gran flexibilidad ha permitido que sea utilizado en un rango muy amplio de sistemas de

cómputo y arquitecturas: computadoras personales, supercomputadoras, dispositivos portátiles, etc.

Los sistemas Linux funcionan sobre más de 20 plataformas diferentes de hardware; entre ellas las más comunes son las de los sistemas compatibles con PCs x86 y x86-64, computadoras Macintosh, PowerPC, Sparc y MIPS.

Asimismo, existen Grupos de Usuarios de Linux en casi todas las áreas del planeta.

Mainframe

Un ordenador central o mainframe es un ordenador grande, potente y costoso usado principalmente por una gran compañía para el procesamiento de una gran cantidad de datos; por ejemplo, para el procesamiento de transacciones bancarias.



Figura N° 44 Imagen de un mainframe Honeywell-Bull DPS 7 Mainframe BWW

A menudo, los ordenadores centrales soportan miles de usuarios de manera simultánea que se conectan mediante falsos terminales. Algunos ordenadores centrales pueden ejecutar o dar cobijo a muchos sistemas operativos y por lo tanto, no funcionan como un ordenador sólo, sino como varios ordenadores virtuales. En este papel, un ordenador central por sí solo puede reemplazar docenas o cientos de pequeños ordenadores personales, reduciendo los costes administrativos y de gestión al tiempo que ofrece una escalabilidad y fiabilidad mucho mejor.

Máquina Virtual Paralela

La Máquina Virtual Paralela (conocida como PVM por sus siglas en inglés de Parallel Virtual Machine) es una librería para la el computo paralelo en un sistema distribuido de computadoras. Esta diseñado para permitir permitir que una red de computadoras heterogénea comparta sus recursos de computo (como el procesador y la memoria RAM) con el fin de aprovechar esto para disminuir el tiempo de ejecución de un programa al distribuir la carga de trabajo en varias computadoras.

La librería PVM fue desarrollada por la Universidad de Tennessee, el Laboratorio Nacional Oak Ridge y la Universidad Emory. La primera versión fue escrita en ORNL en 1989, y después de ser modificado por la Universidad de Tennessee fue lanzada la versión 2 en marzo de 1991. La versión 3 fue lanzada en marzo de 1993 con mejoras en la tolerancia a fallas y portabilidad.

Marco de trabajo

Refiérase a framework

Message Passing Interface

La Interfaz de Paso de Mensajes (conocido ampliamente como MPI, siglas en inglés de Message Passing Interface) es un protocolo de comunicación entre computadoras. Es el estándar para la comunicación entre los nodos que ejecutan un programa en un sistema de

memoria distribuida. Las implementaciones en MPI consisten en un conjunto de librerías de rutinas que pueden ser utilizadas en programas escritos en los lenguajes de programación C, C++, Fortran y Ada. La ventaja de MPI sobre otras librerías de paso de mensajes, es que los programas que utilizan la librería son portables (dado que MPI ha sido implementado para casi toda arquitectura de memoria distribuida), y rápidos, (porque cada implementación de la librería ha sido optimizada para el hardware en la cual se ejecuta).

Microsoft SQL Server

Microsoft SQL Server es un sistema de gestión de bases de datos relacionales (SGBD) basada en el lenguaje SQL, capaz de poner a disposición de muchos usuarios grandes cantidades de datos de manera simultánea.

Entre sus características figuran:

Soporte de transacciones.

Gran estabilidad.

Gran seguridad.

Escalabilidad.

Soporta procedimientos almacenados.

Incluye también un potente entorno gráfico de administración, que permite el uso de comandos DDL y DML gráficamente.

Permite trabajar en modo cliente-servidor donde la información y datos se alojan en el servidor y las terminales o clientes de la red sólo accesan a la información.

Además permite administrar información de otros servidores de datos

Este sistema incluye una versión reducida, llamada MSDE con el mismo motor de base de datos pero orientado a proyectos más pequeños.

Microsoft SQL Server constituye la alternativa de Microsoft a otros potentes sistemas gestores de bases de datos como son Oracle o Sybase.

Es común desarrollar completos proyectos complementando Microsoft SQL Server y Microsoft Access a través de los llamados ADP (Access Data Project). De esta forma se completa una potente base de datos (Microsoft SQL Server) con un entorno de desarrollo cómodo y de alto rendimiento (VBA Access) a través de la implementación de aplicaciones de dos capas mediante el uso de formularios Windows.

Para el desarrollo de aplicaciones más complejas (tres o más capas), Microsoft SQL Server incluye interfaces de acceso para la mayoría de las plataformas de desarrollo, incluyendo .NET.

Microsoft SQL Server, al contrario de su más cercana competencia, no es multiplataforma, ya que sólo está disponible en Sistemas Operativos de Microsoft.

Microprocesador

Un microprocesador es un conjunto de circuitos electrónicos altamente integrado para cálculo y control computacional. El microprocesador es utilizado como Unidad Central de Proceso en un sistema microordenador y en otros dispositivos electrónicos complejos como cámaras fotográficas e impresoras, y como añadido en pequeños aparatos extraíbles de otros aparatos más complejos como por ejemplo equipos musicales de automóviles.

El considerado primer microprocesador, el Intel 4004, fue desarrollado en 1971. Los diseñadores jefe fueron Ted Hoff y Federico Faggin de Intel, y Masatoshi Shima de Busicom (más tarde de ZiLOG).

Middleware

El middleware es un software que generalmente actúa entre el Sistema Operativo y las aplicaciones con la función de proveer los siguientes requerimientos de un cluster:

- una interfaz única de acceso al sistema, denominada SSI (Single System Image), la cual genera al usuario la sensación de que utiliza un único ordenador muy potente;
- herramientas para la optimización y mantenimiento del sistema: migración de procesos, checkpoint-restart (congelar uno o varios procesos, mudarlos de servidor y continuar su funcionamiento en el nuevo host), balanceo de carga, tolerancia a fallos, etc.;
- escalabilidad: debe poder detectar automáticamente nuevos servidores conectados al cluster y proceder a su utilización.

Existen diversos tipos, como por ejemplo: MOSIX, OpenMOSIX, Condor, OpenSSI, etc.

El middleware recibe los trabajos entrantes al cluster y los redistribuye de manera que el proceso se ejecute más rápido y el sistema no sufra sobrecargas en un servidor. Esto lo realiza mediante políticas definidas en el sistema (automáticamente o por un administrador) que le indican donde y cómo debe distribuir los procesos y por un sistema de monitorización del cluster, el cual controla la carga de cada CPU y la cantidad de procesos en él.

El middleware también debe poder migrar procesos entre servidores con distintas finalidades:

balancear la carga: si un servidor está muy cargado de procesos y otro está ocioso, pueden pasarse procesos a este último para liberar al primero y optimizar el funcionamiento

mantenimiento de servidores: si hay procesos corriendo en un servidor que necesita mantenimiento o una actualización, es posible migrar los procesos a otro servidor y proceder a desconectar del cluster al primero.

priorización de trabajos: en caso de tener varios procesos corriendo en el cluster, pero uno de ellos de mayor importancia que los demás, puede migrarse este proceso a los servidores que posean más o mejores recursos para acelerar su procesamiento.

Multiproceso

Multiproceso es tradicionalmente conocido como el uso de múltiples procesos concurrentes en un sistema en lugar de un único proceso en un instante determinado. Como la multitarea que permite a múltiples procesos compartir una única CPU, múltiples CPUs pueden ser utilizados para ejecutar múltiples hilos dentro de un único proceso.

El multiproceso para tareas generales es, a menudo, bastante difícil de conseguir debido a que puede haber varios programas manejando datos internos (conocido como estado o contexto) a la vez. Los programas típicamente se escriben asumiendo que sus datos son incorruptibles. Sin embargo, si otra copia del programa se ejecuta en otro procesador, las dos copias pueden interferir entre sí intentando ambas leer o escribir su estado al mismo tiempo. Para evitar este problema se usa una variedad de técnicas de programación incluyendo semáforos y otras comprobaciones y bloqueos que permiten a una sola copia del programa cambiar de forma exclusiva ciertos valores.

Multitarea

Es una característica de un sistema operativo moderno. Permite que varios procesos sean ejecutados al mismo tiempo compartiendo uno o más procesadores.

Tipos de multitarea:

- Nula: El sistema operativo carece de multitarea. Aún así puede lograrse a veces algo parecido a una multitarea implementándola en espacio de usuario, o usando trucos como los TSR de MS-DOS. Un ejemplo típico de sistema sin multitarea es MS-DOS y sus clones.
- Cooperativa: Los procesos de usuario son quienes ceden la CPU al sistema operativo a intervalos regulares. Muy problemática, puesto que si el proceso de usuario se cuelga y no cede la CPU al sistema operativo, todo el sistema estará entonces colgado. Da lugar también a latencias muy irregulares, y la imposibilidad de tener en cuenta este esquema en sistemas operativos de tiempo real. Un ejemplo sería Windows hasta la versión 95.
- Preferente: El sistema operativo es el encargado de administrar el/los procesador/es, repartiendo el tiempo de uso de este entre los procesos que estén esperando para utilizarlo. Cada proceso utiliza el procesador durante cortos períodos de tiempo, pero el resultado final es prácticamente igual que si estuviesen ejecutándose al mismo tiempo. Ejemplos de sistemas de este tipo serían Unix y sus clones (FreeBSD, Linux...), VMS y derivados, AmigaOS, Windows NT...
- Real: Sólo se da en sistemas multiprocesador. Es aquella en la que varios procesos se ejecutan realmente al mismo tiempo, en distintos microprocesadores. Suele ser también preferente. Ejemplos de sistemas operativos con esa capacidad: variantes Unix, Linux, Windows NT, etc.

Niveles de calendarización

En los sistemas operativos de propósito general, existen dos tipos de calendarizador. El calendarizador a corto plazo es el que se ha descrito aquí, siendo también el más importante. En inglés, se denomina scheduler. Pero también existe un calendarizador a largo plazo (en inglés, planner) relacionados con aquellos procesos que no se encuentran inmediatamente

en memoria principal. Su misión es mover procesos entre memoria principal y disco (lo que se conoce como swapping).

Estos dos calendarizadores son habituales en sistemas operativos multi-hilo, donde cada proceso puede constar de varios hilos. Se supone que cada hilo debe ejecutarse concurrentemente con los demás hilos de su propio proceso y de otros procesos. Así, el calendarizador a corto plazo se preocupa de los hilos, mientras que el calendarizador a largo plazo se preocupa de los procesos.

Nodos

Pueden ser simples ordenadores, sistemas multi procesador o estaciones de trabajo (workstations).

Espacio real o subabstracto en el que confluyen parte de las conexiones de otros espacios reales o abstractos que comparten sus mismas características y que a su vez también son nodos. Todos estos nodos se interrelacionan entre sí de una manera no jerárquica y conforman lo que en términos sociológicos o matemáticos le llamamos red

Peer-To-Peer

A grandes rasgos, una red informática entre iguales (en inglés peer-to-peer -que se traduciría de par a par- o de punto a punto, y más conocida como pedospe) se refiere a una red que no tiene clientes y servidores fijos, sino una serie de nodos que se comportan simultáneamente como clientes y como servidores de los demás nodos de la red. Este modelo de red contrasta con el modelo cliente-servidor la cual se rige de una arquitectura monolítica donde no hay distribución de tareas entre sí, solo una simple comunicación entre un usuario y una terminal en donde el cliente y el servidor no pueden cambiar de roles.

Cualquier nodo puede iniciar, detener o completar una transacción compatible. La eficacia de los nodos en el enlace y trasmisión de datos puede variar según su configuración local

(cortafuegos, NAT, ruteadores, etc.), velocidad de proceso, disponibilidad de ancho de banda de su conexión a la red y capacidad de almacenamiento en disco.

El P2P se basa principalmente en la filosofía de que todos los usuarios deben compartir, conocida como filosofía P2P la cual es aplicada en algunas redes P2P en forma de un sistema enteramente meritocrático en donde "el que mas comparta, mas privilegios tiene y mas acceso dispone de manera más rápida a mas contenido". Con este sistema se pretende asegurar la disponibilidad del contenido compartido, ya que de lo contrario no sería posible la subsistencia de una red P2P.

Aquellos usuarios que no comparten contenido en una red P2P, se les denomina "leechers", los cuales muchas veces representan una amenaza para la disponibilidad de recursos en una red P2P debido a que solo consumen recursos sin reponer lo que consumen, por ende podrían agotar los recursos compartidos de una red P2P y atentar contra la estabilidad de la misma.

Políticas de calendarización

A continuación se enumeran diversas políticas de calendarización. Lo habitual es utilizar políticas mixtas. Generalmente, el calendarizador a corto plazo utiliza round-robin, mientras que el calendarizador a largo plazo utiliza varias colas FIFO. Cada una de estas colas corresponde a una prioridad diferente.

- Planificación Round-robin
- Round-robin con pesos.
- Prioridades monótonas en frecuencia (RMS (Rate-monotonic scheduling))
- Menor tiempo de respuesta primero (EDF (Earliest deadline first scheduling))
- FIFO.
- LIFO.

Proceso (informática)

Un proceso es un concepto manejado por el sistema operativo que consiste en el conjunto formado por:

Las instrucciones de un programa destinadas a ser ejecutadas por el microprocesador.

Su estado de ejecución en un momento dado, esto es, los valores de los registros de la CPU para dicho programa.

Su memoria de trabajo, es decir, la memoria que ha reservado y sus contenidos.

Otra información que permite al sistema operativo su planificación.

Esta definición varía ligeramente en el caso de sistemas operativos multihilo, donde un proceso consta de uno o más hilos, la memoria de trabajo (compartida por todos los hilos) y la información de planificación. Cada hilo consta de instrucciones y estado de ejecución.

Los procesos son creados y destruidos por el sistema operativo, pero lo hace a petición de otros procesos. El mecanismo por el cual un proceso crea otro proceso se denomina bifurcación (fork). Los nuevos procesos son independientes y no comparten memoria (es decir, información) con el proceso que los ha creado.

En los sistemas operativos multihilo es posible crear tanto hilos como procesos. La diferencia estriba en que un proceso solamente puede crear hilos para sí mismo y en que dichos hilos comparten toda la memoria reservada para el proceso.

Red de computadoras

Una red de computadoras (también llamada red de ordenadores, red informática o red a secas) es un conjunto de tres o más computadoras o dispositivos conectados entre sí y que comparten información (archivos), recursos (CD-ROM, impresoras, etc.) y servicios (e-mail, chat, juegos), etc.

Red Hat Linux

Red Hat Linux es una distribución Linux creada por Red Hat, que fue una de las más populares en los entornos de usuarios domésticos.

Es una de las distribuciones Linux de "mediana edad". La versión 1.0 fue presentada el 3 de noviembre de 1994. No es tan antigua como la distribución Slackware, pero ciertamente es más antigua que muchas otras. Fue la primera distribución que usó RPM como su formato de paquete, y en un cierto plazo ha servido como el punto de partida para varias otras distribuciones, tales como la orientada hacia PCs de escritorio Mandrake Linux (originalmente Red Hat Linux con KDE), Yellow Dog Linux, la cual se inició desde Red Hat Linux con soporte para PowerPC, y ASPLinux (Red Hat Linux con mejor soporte para caracteres no-Latinos).

Desde el 2003, Red Hat ha desplazado su enfoque hacia el mercado de los negocios con la distribución Red Hat Enterprise Linux. Red Hat Linux 9, la versión final, llegó oficialmente al final de su vida útil el pasado 30 de abril de 2004, aunque el proyecto Fedora Legacy continua publicando actualizaciones.

Sandbox

Término en inglés utilizado para identificar un espacio de trabajo aislado, cualquier tarea o programa que se ejecute dentro del sandbox únicamente puede utilizar recursos localizados dentro del mismo.

Sistema Operativo

Debe ser de fácil uso y acceso y permitir además múltiples procesos y usuarios. Ejemplos:

- GNU/Linux
- Solaris
- Windows NT
- Sistema Operativo Windows 2000 Server
- Sistema Operativo Windows 2003 Server
- Cluster OS
- Etc

Solaris

Solaris es un sistema operativo de la empresa Sun Microsystems basado inicialmente en el sistema UNIX BSD de la Universidad de Berkeley, del cual uno de los fundadores de la compañía fue programador en sus tiempos universitarios. Más adelante incorporó funcionalidades del System V, convirtiéndose prácticamente en un sistema operativo totalmente basado en System V. Quizá sea uno de los UNIX comerciales más usados, principalmente en el entorno Internet.

Es una evolución del anterior sistema SunOS de la compañía, aunque la transición no fue de una versión para otra. En 1992 salió la segunda versión de Solaris, que incorporó las funcionalidades del System V, por lo que SunOS solo tendría sentido a partir de ese momento como núcleo de este nuevo entorno operativo Solaris. De esta forma Solaris 2 contenía SunOS 5.0. Desde ese momento se distingue entre el núcleo del sistema operativo (SunOS), y el entorno operativo en general (Solaris), añadiéndole otros paquetes como Apache o DTrace. Como ejemplo de esta función, Solaris 8 contiene SunOS 5.8.

Solaris funciona principalmente sobre la arquitectura SPARC en 32 y 64 bits (esta última conocida como UltraSparc) de la misma compañía y sobre la arquitectura Intel, aunque en esta última se suele usar con fines didácticos, y escasamente en entornos de producción.

Proporcionó desde sus primeros momentos un excelente soporte para aplicaciones de red en protocolos IP, y fue el primer entorno donde se desarrolló el sistema Java, donde sigue teniendo un excelente rendimiento.

Proporciona prácticamente todas las funcionalidades típicas de los sistemas UNIX en entorno servidor, como Sockets, Multitarea, Threads, entorno de escritorio basado en X-Window en el que se incluye diferentes escritorios como Open Look CDE o más recientemente Sun Java Desktop System, basado en GNOME.

En los últimos tiempos la compañía ha puesto en marcha una clara estrategia de acercamiento entre Solaris y Linux desarrollando incluso productos que permiten ejecutar programas de Linux en Solaris.

Software

Software -también conocido como programática o equipamiento lógico es el conjunto de programas que puede ejecutar el hardware para la realización de las tareas de computación a las que se destina. Se trata del conjunto de instrucciones que permite la utilización del ordenador. El software es la parte intangible de la computadora, es decir programas, aplicaciones etc.

El término software fue usado por primera vez en 1957 por John W. Turkey, ingeniero en ciencias de la computación. Surge por analogía con la palabra hardware, el sustrato físico donde se ejecuta el software.

Hoy día, el software está presente en casi todo lo que nos rodea, aunque no siempre de manera perceptible.

Probablemente la definición más formal de software es la atribuida a la IEEE en su estándar 729: la suma total de los programas de cómputo, procedimientos, reglas [,] documentación y datos asociados que forman parte de las operaciones de un sistema de cómputo (Ver: IEEE Std 729-1993, IEEE Software Engineering Standard 729-1993: Glossary of Software Engineering Terminology. IEEE Computer Society Press, 1993).

Bajo esta definición el concepto de software va más allá de los programas de cómputo en sus distintas formas: código fuente, binario o código ejecutable, además de su documentación. Es decir, el software es todo lo intangible.

Superordenador

Supercomputadoras son computadoras con capacidades computacionales muy superiores a las comunmente disponibles en la época en que fue construída.

Hoy en día el diseño de Supercomputadoras se sustenta en 3 importantes tecnologías:

- la de Registros Vectoriales, creada por Seymour Cray, considerado el padre de la Supercomputación, quien inventó y patentó diversas tecnologías que condujeron a la creación de máquinas de computación ultra-rápidas. Esta tecnología permite la ejecución de inúmeras operaciones aritméticas en paralelo.
- el sistema conocido como M.P.P. por las siglas de Massively Parallel Processors o Procesadores Masivamente Paralelos, que consiste en la utilización de cientos y a veces miles de microprocesadores estrechamente coordinados.
- clusters de ordenadores de uso general y relativo bajo costo, interconectados por redes locales de baja latencia y el gran ancho de banda.

Recientemente, con la popularización de la Internet, han surgido proyectos de computación distribuída, en donde softwares especiales aprovechan el tiempo ocioso de miles de ordenadores personales para realizar grandes tareas por un bajo costo.

Por su alto costo, los superordenadores tienen su uso limitado a organismos gubernamentales, militares y grandes centros de investigación, en donde tienen aplicaciones científicas, como en la simulación de procesos naturales (previsión del tiempo, análisis de cambios climáticos, entre otros procesos), modelaje molecular, simulaciones físicas como túneles de viento, criptoanálisis, etc. Pero probablemente el mayor uso de este tipo de tecnología es en el desarrollo de armas de destrucción masiva, por el gobierno estadounidense y en la criptoanálisis, por la Agencia de Seguridad Nacional (NSA) del mismo país.

Este tipo de máquinas generalmente tiene su arquitectura proyectada y optimizada enteramente con la aplicación final en mente.

TCO (Total cost of ownership)

El Costo Total de Propiedad, TCO por sus siglas en inglés, es un tipo de cálculo diseñado para ayudar a los consumidores y a los administradores a evaluar los costos (directos e indirectos) y los beneficios relacionados con la adquisición de cualquier componente relacionado con la tecnología de la información. Su intención es llegar a un monto final que refleje el costo efectivo de la compra y que comprenda todos los aspectos.

El análisis de TCO es original de Gartner Group, el cual ha desarrollado gran cantidad de metodologías y herramientas de software para aplicarlas. Un análisis de TCO para la compra de un servidor incluye precio, servicio y soporte, consideraciones de la red para instalar el servidor, seguridad, capacitación de los usuarios y licenciamiento de software, entre otros.

Toolkit

Refiérase a Framework.

Unix

Unix® (o UNIX) es un sistema operativo portable, multitarea y multiusuario; desarrollado en principio por un grupo de empleados de los laboratorios Bell de AT&T, entre los que figuran Ken Thompson, Dennis Ritchie y Douglas McIlroy.

Hoy día, la palabra UNIX se utiliza para denotar diferentes conceptos dependiendo del contexto en que es usada. Esto suele dar lugar a confusiones:

UNIX - familia: desde el punto de vista técnico, UNIX se refiere a una familia de sistemas operativos que comparten unos criterios de diseño e interoperabilidad en común. Esta familia incluye más de 100 sistemas operativos desarrollados a lo largo de 20 años. No obstante, es importante señalar que esta definición no implica necesariamente que dichos sistemas operativos compartan código o cualquier propiedad intelectual.

UNIX - el sistema operativo original: desde el punto de vista histórico, UNIX se refiere a la subfamilia de sistemas operativos que descienden de la primera implementación original de AT&T. El termino descendencia ha de interpretarse como trabajos derivativos que comparten propiedad intelectual con la implementación original.

UNIX - la marca: desde el punto de vista legal, Unix es una marca de mercado. Dicha marca es propiedad de “The Open Group”, una organización de estandarización que permite el uso de dicha marca a cualquier sistema operativo que cumpla con sus estándares publicados. Todo ello independientemente de que el sistema operativo en cuestión sea descendiente o clónico del Unix original. Resumiendo, la marca Unix no es propiedad de ninguna compañía.

Windows

Microsoft Windows es el nombre de una familia de sistemas operativos no libres desarrollados por la empresa de software Microsoft Corporation. Todos ellos tienen en común el estar basados en una interfaz gráfica de usuario basada en el paradigma de ventanas (de ahí su nombre en inglés).

Las versiones de Windows que existen hasta el momento se basan en dos líneas separadas de desarrollo que finalmente convergen en una sola con la llegada de Windows XP.

XIV. Bibliografía

- Accenture (2007). *Grid Computing Harnessing the power of many to deliver high performance*. Accenture Corporation 2007.
- Alibhai Zafeer (2006). *Contract Net Protocol*. IRMS Laboratory, SFU. Disponible en <http://www.ensc.sfu.ca/research/irms/courses/files/Contract%20Net%20Protocol.pdf> . Recuperado el 17 de abril del 2006.
- Bednarz A. y Dubie D. (2003). *How to get to utility computing*. Network World, Diciembre 2003.
- Berman Fran, Fox Geoffrey y Hey Tony (2003). *The Grid. Past, present, future*. Editorial John Wiley & sons, Ltd, 2003.
- Buckler, Grant (2003). *Volunteer computing?*. Computer Dealer News, 18 de abril del 2003, volumen 19 número 6, p15, 3/5p
- Buyya Rajkumar (2002), *Economic-based Distributed Resource Management and Scheduling for Grid Computing*. School of computer science and software engineering Monash University, Melbourne Australia, 2002.
- Buyya Rajkumar, Abramson David y Giddy Jonathan (2006). *Nimrod/G: An architecture for a Resource Management and Scheduling System in a Global Computational Grid*. Monash University, University of Queensland, 2006
- Buyya Rajkumar, S. Chapin, and D. DiNucci. (2000). *Architectural Models for Resource Management in the Grid*, First IEEE/ACM International Workshop on Grid Computing (GRID 2000), Springer Verlag LNCS Series, Germany, Dec. 17, 2000, Bangalore, India.

- Buyya Rajkumar, Stockinger Heinz, Giddy Jonathan, Abramson David. (2002). *Economic models for resource management and scheduling in Grid computing*. Monash University, Melbourne, Australia y CERN, European Organization for Nuclear Research. 2002.
- Buyya Rajkumar, Stockinger Heinz, Giddy Jonathan, Abramson David. (2004). *Economic Models for Management of Resources in Peer-to-Peer and Grid Computing*. Monash University, Melbourne, Australia y CERN, European Organization for Nuclear Research. 2004.
- Buyya Rajkumar. (2005). Grid Computing Info Centre. Recuperado el 11 de junio del 2005, de <http://www.GridComputing.com>.
- Casanova H., Plank M. Y Dongarra J. *A network Server for Solving Computational Science Problems*. International Journal of Supercomputing Applications and HPC, Volumen 11, Noviembre 1997.
- Cerbero (2006). El método. Disponible en <http://www.monografias.com/trabajos6/elme/elme2.shtml>. Recuperado el 05 de enero del 2007.
- Chapin Steve J., Cirne Walfredo, Feitelson Dror G., Patton Jones James, Leutenegger Scott T., Schwiegelshohn Uwe, Smith Warren, and Talby David, "Benchmarks and Standards for the Evaluation of Parallel Job Schedulers". In *Job Scheduling Strategies for Parallel Processing*, D. G. Feitelson and L. Rudolph (Eds.), Springer-Verlag, 1999, Lect. Notes Comput. Sci. vol. 1659, pp. 66-89.
- Chetty M. y Buyya Rajkumar (2002). *Weaving Computacional Grids: How Analogous are they with electrical grids?* IEEE Computing Society, Julio/Agosto del 2002.
- Coffee, Peter (2004). *Grids in the enterprise*. eWeek, 09 de febrero del 2004, volumen 21 número 6, p49-51, 3p.

- Cohen, Robert B. (2003). Grid Computing - Projected Impact on North Carolina's Economy and Broadband Use through 2010. Rural Internet Access Authority Research Report, Setiembre 2003.
- Computer World, 2005. *Gird 101*. Computerworld; 16 de mayo del 2005, volumen 39 número 20, sección especial, p2-3, 2p.
- Dror Feitelson. 2006. Parallel Workloads Archive. Disponible en <http://www.cs.huji.ac.il/labs/parallel/workload>. Recuperado el 28 de agosto del 2006.
- D. Ferguson, C. Nikolaou, J. Sairamesh, and Y. Yemini. (1996). *Economic Models for Allocating Resources in Computer Systems, In Market-based Control: A Paradigm for Distributed Resource Allocation*, World Scientific Press, Singapore, 1996.
- D. Reed, I. Pratt, P. Menage, S. Early, and N. Stratford. (1999). *Xenoservers; Accounted execution of untrusted code*, Proceedings of the 7th Workshop on Hot Topics in Operating Systems (HotOSVII), Rio Rico, Arizona, USA, March 28-30, 1999, IEEE CS Press, USA, 1999.
- De Lancie, Philip (2003). Get a grip on grids. EContent, enero del 2003, volumen 26 número 1, p36, 7p
- De Roure, David (2002). *The evolution of the grid*. University of Southampton, Reino Unido, 2002.
- Dongara Jack, 2004. *High Performance Computing, Clusters and Grids, Oh My!*. University of Tennessee and Oak Ridge National Laboratory, 2004.
- Fernández A., Heymann E., Salt J. y Senar M. (2004). *Servicios de asignación y planificación de recursos en grid*. Boletín Red Iris, número 66-67, año 2004.

- Fernández, Pita A. Metodología de la investigación (1996). Recuperado el 08 de enero del 2007. Disponible en <http://www.fisterra.com/mbe/investiga/9muestras/9muestras.htm>
- Ferreira Luis, Berstis Viktors, et al. (2003). *Introduction to Grid Computing with Globus*. IBM Corporation 2003.
- Foster, Ian (2003). *THE GRID: Computing without Bounds*. Scientific American; Abril del 2003, volumen 288 número 4, p78, 8p
- Fuentes A., Huedo E., Montero S. Y Llorente M (2004). *Impacto de la interconexión en la planificación de trabajos en Grids*. Boletín de RedIris, número 68-69, setiembre 2004.
- G. Tesauro and J. Kephart. (1998). *Foresight-based pricing algorithms in an economy of software agents*, First International Conference on Information and Computation Economies, Charleston, South Carolina, October, 1998. http://www.ibm.com/iac/papers/ice98_fs/
- Gentzsch, Wolfgang (2004). *Riding the grid wave*. Computer World., 16 de junio del 2004, volumen 38, número 49, p30-30, 1p
- Globus Alliance (2006). Carl Kesselman, *Press Releases*, recuperado el 06 de mayo del 2006, disponible en www.globus.org
- Grajales G., Tevni (2003). Tipos de investigación. Recuperado el 20 de diciembre del 2006, disponible en <http://tgrajales.net/investipos.pdf>
- Gould, Lawrence S. (2003). Power from the masses with grid computing. *Automotive Design & Production*, abril 2003, volumen 115 número 4, p50, 1p
- Gujarati, Damodar (2004). *Basic Econometrics*. West Point Military Academy. Editorial McGraw-Hill, Cuarta Edición, 2004.

- Hernández Sampieri Roberto (1991). *Metodología de la investigación*. Editorial McGraw Hill, México, 1991.
- Higgins, Amy y Koucky (2002). *Sherri Grids share CPUs for big analysis projects*. *Machine Design*, 07 de noviembre del 2002, volumen 74 número 21, p37, 1p, 1c
- Ian Foster and C. Kesselman. (1999). *The Grid: Blueprint for a Future Computing Infrastructure*, Morgan Kaufmann Publishers, USA, 1999.
- IBM Corporation (2006). *IBM Grid Computing for Economic Development*. Recuperado el 06 de mayo del 2006. Disponible en http://www-1.ibm.com/grid/pdf/grid_computing_brochure.pdf
- InfoWorld, 08 de setiembre del 2003, volumen 25 número 35, p16-17, 2p
- Insight Research Corporation (2005). *Grid Computing: Vertical Market Perspective (2005)*. New Jersey, United States, 2005.
- K. Czajkowski, I. Foster, N. Karonis, C. Kesselman, S. Martin, W. Smith, and S. Tuecke. (1998). *Resource Management Architecture for Metacomputing Systems*, Proceedings of the 4th International Workshop on Job Scheduling Strategies for Parallel Processing, March 30, 1998, Orlando, Florida, USA, Springer Verlag Press, Germany, 1998.
- K. Krauter, R. Buyya, and M. Maheswaran. (2002). *A Taxonomy and Survey of Grid Resource Management Systems for Distributed Computing*, International Journal of Software: Practice and Experience (SPE), Wiley Press, New York, USA, May 2002.
- Kenichi Omae (1995). *The evolving global economy: making sense of the new world order*. Boston, MA : Harvard Business School, 1995.

- Kite, Shane (2002). *Grid Is Here; Where's Web?* Securities Industry News, 09 de setiembre del 2002, volumen 14 número 35, p1, 2p, 2c
- Krass, Meter (2003). *Grid Computing*. CFO 2003, volumen 19 número 15, p38-39, 2p
- Lisa, Gary (2004). Grid computing captures attention. *Electric Light & Power*. Enero /Febrero del 2004, volumen 82 Número 1, p6-6, 3/4p
- Lundquist, Eric (2002). *The power in grid computing, numbers*. eWeek, 22 de julio del 2002, volumen 19 número 29, p5, 1/2p
- M. Stonebraker, R. Devine, M. Kornacker, W. Litwin, A. Pfeffer, A. Sah, and C. Staelin. (1994). *An Economic Paradigm for Query Processing and Data Migration in Mariposa*, Proceedings of 3rd International Conference on Parallel and Distributed Information Systems, Austin, TX, USA, 28-30 Sept. 1994. Los Alamitos, CA, USA: IEEE Computer Society Press, 1994.
- Margulius, David L. (2002). *Next-gen distributed computing*. InfoWorld, 01 de abril del 2002, volumen 24 número 13, p40, 2p
- Marsh, Paul P (2005). *Get on the GRID*. Computing & Control Engineering. Febrero/Marzo del 2005, volumen 16, número 1, p41-45, 5p
- Mearian, Lucas (2004). *Schwab Deploys Linux-based Grid*. : Computerworld, 01 de mayo del 2004, volumen 38 número 1, p6-6, 1/3p
- Metz, Cade (2003) *Grid Computing*. PC Magazine, julio del 2003, volumen 22 número 12, p106, 2p, 1c

Metz, Cade (2003). *Grid Computing*. PC Magazine, enero del 2003, volumen 22 número 1, p80, 1p, 2c

Meza Cascante, Luis Gerardo, (2003). *El paradigma positivista y la concepción dialéctica del conocimiento*. Escuela de Matemática, Instituto Tecnológico de Costa Rica, 2003. Recuperado el 10 de Julio de 2006, disponible en <http://www.cidse.itcr.ac.cr/revistamate/ContribucionesV4n22003/meza/pag1.html>

Minoli, Daniel. *A networking approach to grid computing*. John Wiley & Sons, Inc, New Jersey, Estados Unidos, 2005.

Murillo Hernández, William. La investigación científica. Recuperado el 24 de diciembre del 2006. Disponible en <http://www.monografias.com/trabajos15/invest-cientifica/invest-cientifica.shtml>.

N. Nisan, S. London, O. Regev, and N. Camiel. (1998). *Globally Distributed computation over the Internet: The POPCORN project*, International Conference on Distributed Computing Systems (ICDCS'98), May 26 - 29, 1998, Amsterdam, The Netherlands, IEEE CS Press, USA, 1998.

Neel, Dan (2002). *Grid readied for the enterprise*. InfoWorld, 25 de febrero del 2002, volumen 24 número 8, p13, 1/4p

Olsen, Florence (2003). *'Grids' May Be the Next Wave in High-Performance Computing*. Chronicle of Higher Education, 03 de enero del 2003, volumen 49 número 17, pA36, 1/4p

Palazón, Javier (2003). *Grid Computing, potencia compartida*. Recuperado el 01 de abril de 2006, disponible en http://www.microsoft.com/spain/enterprise/perspectivas/numero_9/research.msp

Platform Computing, 2004. Market Study Report. 2004 Investment Survey. 2004.

Research Triangle Park, Carolina News Wire (2006). New global grid computing technology demonstrated by researchers in US and Japan. Recuperado el 15 de setiembre del 2006. Disponible en <http://carolinanewswire.com/news/News.cgi?database=1news.db&command=viewone&id=162&op=t>.

S. Lalis and A. Karipidis. (2000). *An Open Market-Based Framework for Distributed Computing over the Internet*, Proceedings of the First IEEE/ACM International Workshop on Grid Computing (GRID 2000), Dec. 17, 2000, Bangalore, India, Springer Verlag Press, Germany, 2000.

Sairamesh Jakka y Kephart J.. (1998). *Price Dynamics of Vertically Differentiated Information Markets*, Proceedings of the First International Conference on Information and Computation Economies, Charleston, South Carolina, October, 1998.

Sandholm, Thomas (1999). *Distributed Rational Decision Making*. Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence. 1999.

Santibañez, Leopoldo. (2004). Tesis de doctorado, Modernización portuaria formulación técnico económica para un modelo de concesiones para Costa Rica. Universidad Latinoamericana de Ciencia y Tecnología, Agosto 2004, San José, Costa Rica.

- Scannell, Ed y Krill, Paul (2003). *Oracle Bangs on Grid Drum*.
- Scannell, Ed, Krill, Paul (2003). *Oracle Bangs on Grid Drum*. InfoWorld, 08 de setiembre del 2003, volumen 25, número 35, p16-17, 2p.
- Schick, Shane (2002). *Biotech community explores grid computing to extend life of IT assets*. Computing Canada, 05 de julio del 2002, volumen 28 número 14, p28, 1/2p.
- Schroeder, Mary (2003). *Morgan Stanley Jumps on Grid Computing Bandwagon*. Securities Industry News, 06 de octubre del 2003, volumen 15 número 39, p15-15, 1/3p
- Schwartz, Ephraim (2003). *The Power of the Grid*. InfoWorld 08 de setiembre del 2003, Volumen 25 Número 35, p14-14, 1p
- Smith, Heather, Konsynski, Benn (2004). *Grid Computing*. MIT Sloan Management Review 2004, volumen. 46 número 1, p7-9, 3p
- Sun Microsystems (2003). Network Computing Made More Secure, Less Complex With New Reference Architectures, Sun Infrastructure Solution. Recuperado el 06 de mayo del 2006. Disponible en <http://www.sun.com/smi/Press/sunflash/2003-09/sunflash.20030917.3.xml>.
- Taft, Darryl K, (2005). *Grid developers eye Microsoft*. eWeek, 28 de febrero del 2005, volumen 22, número 9, p10-10, 1/3p
- Tanner, John (2003). *Grid computing set for big growth*. America's Network, 15 de mayor del 2003, volumen 107 número 8, p32, 4p
- Tanner, John C. Get on the grid. Telecom Asia, marzo 2003, volumen 14 número 3, p20, 3p
- Taschek, John (2003). *Time to end gridlock*. eWeek, 02 de junio del 2003, volumen 20 número 22, p63, 1p

- Technology Review (2002). *Is Internet history about to repeat itself?* Technology Review, mayo 2002, volumen 105 número 4, p32, 5p
- Teresko, John (2001). The promise of grid computing. Industry Week/IW, setiembre del 2001, volumen 250 número 12, p25, 1p, 1c
- Thibodeau, Patrick (2003). *IBM Expands Grid Computing Offerings*. Computerworld, 05 de mayo del 2003, volumen 37, fascículo 18.
- Thilmany, Jean (2003). *Getting on the grid*. Mechanical Engineering, marzo 2003, volumen 125 número 3, p46, 3p, 2c
- Trombly, Maria (2002). *Grid computing could cut costs, improve performance*. Securities Industry News, 21 de enero del 2002, volumen 14 número 3, p3, 2p
- Trombly, Maria (2004). *Grids Promise to Move Beyond Analytics*. Securities Industry News, 29 de marzo del 2004, volumen 16 número 12, p12-14, 3p, 1c
- Trombly, Maria (2004). *Utility Computing Still to Come*. Securities Industry News, 29 de marzo del 2004, volumen 16 número 12, p12-17, 3p, 2c
- University of Tennessee (2004). High Performance Computing, Clusters, and Grids. Jack Dongarra.
- Vaas, Lisa (2005). *GlobusWorld conference reveals growing pains*. eWeek, 14 de febrero del 2005, volumen 22, número 7, p21-21, 1/2p
- Vijayan, Jaikumar (2003). *IBM Targets Corporate Users With Grid Computing*. Computerworld 03 de febrero del 2003, volumen 37 número 5, p6, 1/4p

Wade, Will (2003). *Grids Put Computers to Work During Operators Down Time*. American Banker, 26 de octubre del 2003, Volumen 168 Número 186, p9, 1/4p

Y. Amir, B. Awerbuch, and R. Sean Borgstrom. (1998). *A Cost-Benefit Framework for Online Management of a Metacomputing System*, Proceedings of 1st International Conference on Information and Computational Economy, October 25-28, 1998, Charleston, SC, USA.

Zhang J, Chung J y Zhou Q. “*Developing grid computing applications, Part I: Introduction of a grid architecture and toolkit for building grid solutions*”. IBM Developers works library. IBM Corporation. Octubre 2002.