

# Procesamiento Masivamente Paralelo en Servidores: Características, Opciones, Aplicaciones y Ejemplos

## Massively Parallel Processing in Servers: Characteristics, Options, Applications and Examples

Diana Alejandra  
Herrera Blanco  
Código: 2162017

Karen Daniela  
Rodríguez Martínez  
Código: 2161906

Víctor Alfonso  
Mantilla Villamizar  
Código: 2151846

Juan Pablo  
Viviescas Romero  
Código: 2152499

Escuela de Ingeniería de Sistemas e Informática  
Universidad Industrial de Santander

10 de agosto de 2019

### Abstract

Massively Parallel Processing Servers (MPPS) are services dedicated to provide data to multiple clients, in the scale of thousands of clients, practically at the same time preserving the practices of Quality of Service. MPPS are very important in the current state of web services such as video streaming services (Youtube, Netflix), cloud storing (Dropbox, Onedrive), social networks (Facebook, Instagram), among many other services and applications. In this paper it is explained the basic specifications of these kind of systems. There are several architectures in the market that fulfill these requirements, which are analyzed in this document and it is proposed what, in the researchers best knowledge, the best fitting candidate for parallel processing. It is also described the relationship between the Machine Learning methods of data processing, like Deep Learning, with MPPS. Finally, it is shown a real life example of this kind of servers.

### Resumen

Servidores de Procesamiento Masivamente Paralelo (MPPS) son servicios dedicados a proveer datos a múltiples clientes, de la escala de los miles de clientes, prácticamente al mismo tiempo conservando las prácticas de Calidad en el Servicio. Los MPPS son muy importantes en el estado actual de los servicios web tales como servicios de streaming de video (Youtube, Netflix), almacenaje en la nube (Dropbox, Onedrive), redes sociales (Facebook, Instagram), entre muchos otros servicios y aplicaciones. En este documento se explican las especificaciones básicas de esta clase de sistemas. Hay muchas arquitecturas en el mercado que cumplen estos requerimientos, los cuales son analizados en este documento y se propone cual, en el conocimiento de los investigadores, el mejor candidato que se ajusta al procesamiento paralelo. Se describe también la relación entre los métodos de procesamiento de datos del Machine Learning, como el Deep Learning, con el MPPS. Finalmente, se muestran algunos ejemplos de la vida real de esta clase de servidores.

## 1. Introducción

En el mundo altamente entrelazado debido a las comunicaciones modernas provistas por la computación en red y sus arquitecturas asociadas, que van desde los clientes hasta los servidores a gran escala que poseen las grandes compañías de la industria. Desde hace algún tiempo es de conocimiento común que los modelos tradicionales de procesamiento de datos "de uno a la vez" no tiene la escalabilidad y velocidades requeridas para procesar los grandes volúmenes de datos que se producen minuto a minuto en la aldea global. Es por este motivo que es de suma importancia conocer y analizar de una forma completa que abarque todos los aspectos de los servidores que se puedan considerar masivamente paralelos. Este documento abarca esos aspectos, desde la descripción conceptual hasta ejemplos reales actualmente en uso de estas arquitecturas.

## 2. Estado del arte

Actualmente, los centros de datos modernos requieren de nivel alto de paralelismo en su desarrollo debido a los miles de solicitudes que reciben, las cuales a menudo se convierten en tareas rutinarias y repetitivas lo que genera la ejecución de secuencias de instrucciones casi idénticas. Los procesadores convencionales ejecutan solicitudes independientes por separado y no aprovechan la similitud de sus flujos de instrucciones [11], lo que genera un gran gasto de energía ya que la mayoría es empleada en buscar, decodificar y programar instrucciones y accesos a la memoria en lugar de dedicarse en las unidades funcionales, que en el caso en que sean similares pueden reducir la energía del procesador [8], [9].

Una técnica arquitectural para aprovechar la similitud de solicitud cruzada es "Una instrucción, múltiples hilos" (Singles-Instruction Multiple-Threads SIMT), que funciona agrupando las solicitudes similares y las ejecuta simultáneamente y en bloque. Permite que cada grupo de instrucciones pueda ser obtenida, decodificada, renombrada, programada y emitida una sola vez pero que se aplique a todo el grupo de solicitudes, es decir, que toda la lógica compleja sea compartida por todos los hilos. [12]

El potencial de los servidores SIMT ha llevado a los investigadores a indagar utilizando GPU. En particular, Rhythm re-implementó el software del servidor desde cero, incluido el servidor web y la pila de red, apuntando al modelo de programación GPU. Aunque sus resultados demostraron que las solicitudes del servidor pueden procesarse de manera SIMT, también expuso el problema fundamental de las GPU en este dominio: las GPU no pueden cumplir con los estrictos requisitos de calidad de servicio (QoS) de las aplicaciones del servidor. Los procesadores de servidor SIMT efectivos deben emplear técnicas de ocultación de latencia, tales como predicción de sucursales, ejecución fuera de orden y especulativa, y organizaciones avanzadas de caché, que están ausentes de las GPU. [12]

El objetivo de MPPS es ejecutar el software del servidor con modificaciones mínimas en la fuente al tiempo que se equilibra la eficiencia SIMT con los estrictos requisitos de QoS del servidor. MPPS combina varios núcleos convencionales con muchos núcleos SIMT optimizados para las cargas de trabajo del servidor, con todos los núcleos compartiendo la LLC y la interfaz de memoria. [12]

En MPPS, los núcleos convencionales son responsables de ejecutar el programador de subprocesos de solicitud y manejar las I / O, las llamadas al sistema y las interrupciones. El planificador es responsable de examinar las solicitudes entrantes, agruparlas según su similitud y enviar los grupos de subprocesos a los núcleos SIMT. [10]

El sistema de memoria MPPS tendrá altas demandas de ancho de banda debido a los accesos concurrentes desde una gran cantidad de subprocesos SIMT. Sin embargo, a diferencia de las GPU, que atienden una gran cantidad de accesos concurrentes al fusionar solicitudes adyacentes, MPPS debe admitir y rastrear una gran cantidad de solicitudes de memoria en vuelo distintas. [10]

## 3. Marco Teórico

### 3.1. Arquitectura computacional

Es la tarea de "determinar cuáles atributos son importantes para un computador nuevo, y luego diseñar un computador para maximizar el desempeño y la eficiencia energética mientras se mantienen las limitaciones de costo, poder y disponibilidad"[1].

En la arquitectura se diseñan, entre otras, el conjunto de instrucciones, organización funcional, la lógica y la implementación.

"La implementación puede abarcar el diseño del circuito integrado, el empaquetamiento, el poder y el enfriamiento"[1].

Para realizar un diseño se deben conocer muchas tecnologías y se deben actualizar constantemente estos conocimientos para estar al día con la innovaciones en estos diferentes campos.

### 3.2. Procesamiento En Paralelo

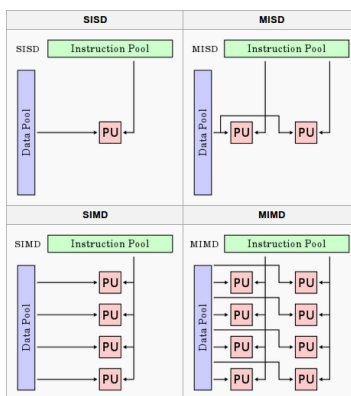


Figura 1: Clasificación de Flynn. Fuente de la imagen: [3]

Según la clasificación de Flynn [2] las arquitecturas SIMD y MIMD son las que se pueden considerar realmente como arquitecturas de procesamiento en paralelo. El procesamiento en paralelo hace que un programa se procese más rápido que en un procesador normal. Tomando los datos del Data Pool y las instrucciones del Instruction Pool, partes de un código se ejecutan en cada procesador que tenga el sistema.

#### 3.2.1. Arquitectura SIMD

- Distribuye el proceso en una cantidad grande de hardware.
- Opera de manera concurrente sobre varios elementos de datos diferentes.
- Realiza el mismo computo sobre todos los elementos de datos.

#### 3.2.2. Arquitectura MIMD

- Una alta tasa de transferencia efectiva se puede conseguir si el proceso se puede dividir en *streams* paralelos, y así conservando todos los procesadores activos.
- Un recurso fallido puede ser removido fácilmente y su trabajo puede ser bicado en otro recurso disponible, consiguiendo así algún grado de tolerancia a los fallos, ya que los procesadores y los bloques de memoria.
- Una reconfiguración dinámica de recursos es posible para acomodar cargas de proceso variable.

### 3.3. Computación Cliente/Servidor

El término provee soluciones para el manejo de datos. También hace referencia a un modelo computacional para el desarrollo de sistemas computarizados. En este modelo se distribuyen funciones entre dos tipos de procesos autónomos: **Servidor** y **cliente**. Un cliente es cualquier proceso que solicita servicios específicos al proceso del servidor. Un servidor es un proceso que provee los servicios solicitados por el cliente. El servidor y el cliente pueden existir en diferentes dispositivos o en el mismo [5].

Cuando un cliente y un servidor residen en equipos de cómputo diferentes, un servidor puede proveer servicios a más de un cliente. Un cliente puede solicitar a su vez servicios de varios servidores.

### 3.4. Inteligencia Artificial

Algunas definiciones de Inteligencia Artificial [6] son:

La Inteligencia Artificial es la rama de la ciencia e ingeniería que se ocupa del entendimiento computacional de lo que se llama comúnmente **comportamiento inteligente**, y con la creación de artefactos que exhiban dicho comportamiento.

*Rauch-Hindin*

La Inteligencia Artificial es el estudio de cómo fabricar computadoras que hagan cosas que, en el momento, las personas hacen mejor

*E. Rich. Traducción propia*

Esta última definición se ajusta a grandes rasgos a lo que es la inteligencia artificial. "Tareas como la ejecución de muchos cálculos en un período corto de tiempo son los puntos fuertes de los computadores digitales... En muchas otras áreas, sin embargo, los humanos son superiores a las máquinas. Por ejemplo, una persona que entra a un cuarto desconocido reconocerá los alrededores en una fracción de segundo y si es necesario, así de simple tomar decisiones y planear actos." [7]

Así que la inteligencia artificial intenta encontrar métodos que le permitan a una máquina emular las formas en la que un ser humano realiza ciertas tareas de manera autónoma.

## 4. Características

La tecnología de procesamiento masivamente paralelo, MPP (Massively Parallel Processing) es un proceso sencillo que consiste en tomar un número de procesadores, cada procesador tiene su propio sistema operativo y memoria, y agruparlos para crear un potente y económico sistema de ordenador a gran escala.

MPP acelera el rendimiento de bases de datos que manejan grandes cantidades de datos. Las bases de datos que usan esta tecnología utilizan procesadores de múltiples núcleos, múltiples procesadores y servidores, y dispositivos de almacenamiento equipados para procesamiento paralelo. Esa combinación permite leer muchos datos en muchas unidades de procesamiento al mismo tiempo para una mayor velocidad.

Debido a la distribución dispersa de los recursos RAM, los procesadores utilizan un esquema de paso de mensajes análogo a los paquetes de datos en redes. Con esto cada sección de memoria observa únicamente aquellos accesos con los que está asociada y se reduce el tráfico del bus. Además, un procesador puede disponer de la memoria RAM sobrante de los otros procesadores, solo en el caso en que su memoria RAM no sea suficiente.

En la medición de la escalabilidad del procesamiento masivamente paralelo (MPP) con grandes volúmenes de datos provistos por una robusta base de datos, se tienen ciertos factores que afectan el rendimiento: [18]

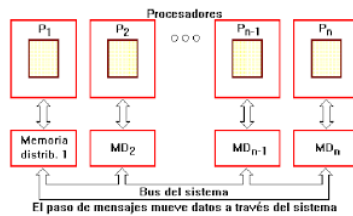


Figura 2: Distribución de memoria en procesamiento masivamente paralelo. Fuente de la imagen: [4]

- Recursos del sistema
 

Depende en gran medida del I/O y el uso de la memoria. Se debe conocer el rendimiento de referencia del hardware en el que se implementó DBMS, CPU, discos duros, controladores de disco, RAM e interfaces de red.
- Carga de trabajo
 

La carga de trabajo es la demanda total del DBMS y claramente varía con el tiempo. Se trata de consultas de usuarios, aplicaciones, transacciones y comandos del sistema solicitados a través del DBMS. Por ejemplo, en situaciones donde se quiere información o trabajos por lotes donde se trabaja con grandes cantidades de datos de un tiempo determinado. Si se traza un plan se puede volver mas eficiente el uso de los recursos del sistema.
- Rendimiento
 

El rendimiento de un DBMS se mide en número de consultas o transacciones por segundo o tiempos de respuesta promedio a estas solicitudes.
- Contención
 

Esta condición se da cuando dos o más componentes de procesos intentan utilizar el sistema de manera conflictiva que terminan compitiendo por lo recursos del sistema al mismo tiempo. A medida que aumenta la contención, disminuye el rendimiento.
- Optimización
 

El rendimiento general del sistema se puede ver afectado por optimizaciones del sistema gestor de base de datos. El optimizador de consultas debe tener planes de acceso eficientes de modo que la formulación de SQL, los parámetros de configuración de BD, la distribución de los datos, entre otros procesos, no afecten el rendimiento del sistema.

## 5. Opciones en en el mercado

Para Las opciones en el mercado analizamos a tres grandes fabricantes en los que se encuentran IBM, NRCPC y NUDT conclumos que el mejor fabricante entre ellos 3 seria IBM.

Para comenzar IBM acaba de crear la supercomputadora con mayor velocidad en el mundo actualmente, la summit, que tiene una velocidad de 200 pentaflaps, esto es posible gracias a su modelo de computación hererogenea, es decir que cuenta con distintos tipos de procesadores con características especializadas para una tarea en concreta.

Para poder procesar en paralelo y lograr esa extraordinaria velocidad de 200 pentaflaps no es una tarea sencilla ya que requiere lograr dominar una serie de retos que impone la computación heterogenea, en los cuales algunos de ellos son.

- Los diferentes procesadores pueden tener diderentes estructuras de memoria caché, la velocidad de leer datos, como por ejemplo que algunos procesadores acceden mediante un byte de tamaño, o de una masa, etc.

- Los procesadores y demás elementos de computo pueden estar interconectados de diferentes formas, como por ejemplo acceso a memoria directa, buzones, etc. Ademas algunas partes pueden tener coherencia en el cache mientras que en otras necesitan de algun tipo de software para mantener su estabilidad.
- Al poseer diferentes tipos de CPU con diferentes procesadores generan un gran gasto de energia que puede reducir su rendimiento.

Por haber logrado superar los inconvenientes mencionados anteriormente y lograr un computador de procesamiento en paralelo que procese con la mayor eficiencia de la computación heterogenea, IBM a nuestro criterio ofrece la mejor alternativa en servidores de procesamiento en paralelo.



Figura 3: Summit - IBM [4]

Al comparar con las otras supercomputadoras de los otros dos fabricantes, la Summit es mucho más potente que la Sunway TaihuLight de NRCPC y la Tianhe de NUDT, aunque la Sunway también están procesando en paralelo no utilizan distintos tipos de procesadores como la Summit de IBM, sino que utilizan una sola gama de procesadores, lo cual significa que está situado en la computación heterogenea y lo que hace que su velocidad sea de 95 petaflops, en cambio la Tianhe utiliza dos tipos de procesador pero no está utilizando su máximo potencial al estar usando computación heterogenea, por lo cual solo alcanza los 93 petaflops, por lo anterior mencionado el mejor fabricante sería IBM.

## 6. Aplicaciones a la inteligencia artificial

Es importante encontrar la mejor manera de procesar la información utilizando dispositivos electrónicos, que de alguna manera emulan el paralelismo masivo de los datos. La inteligencia artificial trata temas de investigación como las redes neuronales, las técnicas de clasificación o los diagramas de flujo; este estudio se maneja a través de big data, caracterizado por su volumen elevado, velocidad elevada y alta variedad, que demanda soluciones innovadoras y eficientes de procesamiento para la mejora del conocimiento y la toma de decisiones.

El procesar información de una alta cantidad de datos requiere plantear un sistema con varios nodos en lugar de uno solo, con algoritmos distribuidos/paralelo efectivos. La razón principal es la rapidez.[15]

El rendimiento de las redes neuronales se mide en conexiones por segundo(cps), es decir, por el número de fragmentos de datos transportados a través de todos los bordes de la red cada segundo. Los procesadores regulares alcanzan hasta 10000 cps, mientras que un masivamente paralelo es capaz de proporcionar hasta 10000000 cps de transferencia y lectura de datos; esto evita las demoras y hace eficiente el procedimiento de inteligencia artificial.[17]

Una aplicación puede realizarse en el proceso de aprendizaje profundo, o mejor conocido como Deep Learning, que es un modelo gerarquico de inteligencia artificial, en que consiste en que una maquina le llega una instrucción simple que luego es enviada a otros niveles que combinan esta nueva información con otra que genera una información más compleja y así sucesivamente, el problema del deep learning

es que al subdividir diferentes procesos el costo computacional se incrementa, haciendo que procese muy lentamente la información, pero si el servidor procesara en paralelo y tuviera una arquitectura similar a los niveles de jerarquía del deep learning, en otras palabras que tuviera un procesador para cada nivel, esto ayudaría a incrementar la velocidad de procesamiento, ya que un procesador estaría a cargo de una parte concreta, mejorando notablemente su eficiencia.

## 7. Ejemplo de la vida real

### **Massively Multiplayer Online Games (MMOG)**

En la actualidad, la ejecución de este tipo de aplicaciones se lleva a cabo mayoritariamente mediante arquitecturas basadas en Cliente-Servidor. La alta influencia de nuevos usuarios (jugadores) compromete cada día los problemas de escalabilidad existentes en estas infraestructuras, para esto, con el fin de mejorar la eficiencia de ejecución y la alta concentración de usuarios, se encuentra la ejecución paralela de los juegos MMOG, la cual distribuye la carga computacional que se genera.

Sin embargo, dividir estas aplicaciones en un conjunto de tareas con el fin de asignarlas a distintos nodos de procesamiento es complejo. En este sentido, se ha desarrollado una solución desde una perspectiva basada en dos fases. En la primera, se estudia el comportamiento y necesidades de recursos de cómputo de las aplicaciones MMOG, y se identifican las características de los diferentes tipos de juegos; en la segunda, se propone una arquitectura híbrida conformada por una parte central y una distribuida, que permite la ejecución paralela de juegos MMOG, aportando escalabilidad y eficiencia. [13]

### **Sistema Data Warehouse en empresas del sector de distribución alimentaria**

Las grandes compañías del sector alimentario disponen de entornos completos de transformación de datos y almacenamiento. Sin embargo, la lentitud en los procesos de integración y análisis de los datos provocaba una tardía respuesta en ámbitos como campañas de marketing, análisis de la cesta de la compra, entre otras. Hace 2 años se comenzó a invertir en tecnologías en paralelo para modernizar los sistemas de Data Warehouse, ya que la toma de decisiones en este negocio se iba a ver muy beneficiada al tener un análisis de datos así de potente.

Data Warehouse reúne información de varias fuentes. Esta recopilación de datos perdura y trasciende a través del tiempo y permite almacenar en un único lugar la información generada en distintos momentos del tiempo por distintas aplicaciones de software, que a su vez han utilizado distintas tecnologías de almacenamiento y variadas técnicas de gestión de bases de datos.

La arquitectura tradicional de data warehouse emplea una estructura de tres niveles compuesta por:

- Nivel inferior: contiene el servidor de base de datos utilizado para extraer datos de muchas fuentes diferentes.
- Nivel medio: el nivel intermedio alberga un servidor OLAP, que transforma los datos en una estructura más adecuada para análisis y consultas complejas.
- Nivel superior: el nivel superior es la capa del cliente. Contiene las herramientas utilizadas para el análisis de datos de alto nivel, informes de consultas y minería de datos.

Así, data warehouse utiliza los datos para respaldar la toma de decisiones administrativas. [14]

### **Redes Neuronales para predecir la aptitud del alumno y sugerir acciones**

una aplicación de redes neuronales como apoyo al docente, al ser capaz de predecir las fallas de los estudiantes. Sobre esta predicción, se podrán establecer las actividades y acciones que el docente puede proponer al alumno para mejorar su desempeño. Para validar las hipótesis se adoptará un diseño

cuasiexperimental.

Las hipótesis a validar son:

- a) Una red neuronal puede predecir fallas de los alumnos en su desempeño.
- b) En base a sus propias predicciones, una red neuronal es capaz de determinar la clase de acciones a tomar por el docente para mejorar el desempeño del alumno

Las redes neuronales son modelos que intentan reproducir el comportamiento del cerebro que constan, de dispositivos elementales de proceso denominados neuronas artificiales.

Una definición clara puede ser: “una red neuronal artificial es un procesador distribuido, masivamente paralelo, que tiene una predisposición natural a almacenar conocimiento experimental y hacerlo disponible para su utilización”

### **Orientación de anuncios y predicción de las preferencias de los clientes**

Los servidores eno paralelo como por ejemplo Facebook constantemente reciben información acerca de las preferencias de los usuarios, como su ropa preferida, gustos musicales, etc. ellos deben clasificar toda esta clase de información para poder ofrecer publicidad adecuada para cada usuario para garantizar que no pase desapercibida, su servidores deben dividir en secciones cada preferencia del usuario para determinar que es lo que más le interesa y enviarle publicidad acerca de ello, esto se realiza mediante deep learning, ya que deben jerarquizar los niveles en que pasará la información para ser procesada, aunque cabe añadir que esta practica debe realizarse con mucho cuidado, porque los usuarios pueden sentir que su privacidad está siendo vulnerada y co ello sus derechos estan siendo violados, como ocurrió en el caso de Cambridge analitica.

## **Referencias**

- [1] Hennessy, J. L., Patterson, D. A. (2011). *Computer Architecture: A Quantitative Approach* (5th ed.). San Diego, CA, USA: Elsevier Science.
- [2] Roosta, S. H. (2013). *Parallel processing and parallel algorithms: Theory and computation*. Rensselaer: Springer.
- [3] <https://i1.wp.com/www.thephysicsmill.com/blog/wp-content/uploads/flynnstaxonomy.png>
- [4] <https://docentes.uaa.mx/guido/wp-content/uploads/sites/2/2014/11/multiprocesamiento.pdf>
- [5] Subhash, C. Y., Sanjay, S. K. (2009). *Introduction to client/server computing*. New Delhi: New Age International Pvt.
- [6] Bourbakis, N. G. (2014). *Artificial Intelligence Methods And Applications*. Springer International Publishing.
- [7] Ertel, W. (2017). *Introduction to Artificial Intelligence*. Cham, Switzerland: Springer International Publishing.
- [8] M. Ferdman, T. Wenisch, A. Ailamaki, B. Falsafi, and A. Moshovos, “Temporal instruction fetch streaming” in Proc. 41st Int. Symp. Microarchitecture, 2008, pp. 1–10
- [9] M. Ferdman, C. Kaynak, and B. Falsafi, “Proactive instruction fetch” in Proc. 44th Int. Symp. Microarchitecture, 2011, pp. 152–162
- [10] N. Hardavellas, M. Ferdman, B. Falsafi, and A. Ailamaki, “Reactive NUCA: Near-optimal block placement and replication in distributed caches,” in Proc. 36th Int. Symp. Comput. Archit., 2009, pp. 184–195



- [11] R. Hameed, W. Qadeer, M. Wachs, O. Azizi, A. Solomatnikov, B. Lee, S. Richardson, C. Kozyrakis, and M. Horowitz, “*Understanding sources of inefficiency in general-purpose chips*” in Proc. 37th Int. Symp. Comput. Architecture, 2010, pp. 37–47.
- [12] Agrawal, V., Dinani, M., Shui, Y., Ferdman, M., Honarmand, N. (2019). *Massively Parallel Server Processors*. IEEE COMPUTER ARCHITECTURE LETTERS, 4.
- [13] Barri Vilardell, I. O. (2012). *Modelado y planificación de aplicaciones de Juegos masivos multijugador en red en entornos distribuidos*. Dialnet.
- [14] Rivas, U. F. (2015). *Modernización del sistema DataWarehouse en empresas del sector de distribución alimentaria*. Getafe.
- [15] Jiménez, C. M. (2014). *Big data. Un nuevo paradigma de análisis de datos*.Comillas, 7.
- [16] Rodríguez, U. G. (2014). *Diseño de una célula básica de procesamiento neuronal en tecnología CMOS*.Veracruz.
- [17] Díaz García, Ma. del Prado (2003) *Un sistema ubicuo para el control de propiedades de refino de crudo*.  
Universidad de Castilla La Mancha [7] Galan-Marin. G. and Muñoz-Perez,
- [18] Daeng, F., Suhajito, Diana, Suganda, A. (2018). *Implementation of Database Massively Parallel Processing System to Build Scalability on Process Data Warehouse*. ScienceDirect, 12. Obtenido de [https://www.researchgate.net/publication/327317047\\_implementation\\_of\\_Database\\_Massively\\_parallel\\_processing\\_system](https://www.researchgate.net/publication/327317047_implementation_of_Database_Massively_parallel_processing_system)