Accountancy Business and the Public Interest
ISSN: 1745-7718

Regular Issue
January 2025

Modern Resource Management and System Resource Allocation Using Constructive Data Processing with Knowledge Engineering Techniques

[1]D. Saravanan, [2]Dr. Anusha Sreeram, [3]Dr. Dennis Joseph, [4]V. Sai sudheer

1,2,3 Faculty of Operations & IT ICFAI Business School (IBS), Hyderabad, The ICFAI Foundation for Higher Education (IFHE), (Deemed to be university u/s 3 of the UGC Act 1956), Hyderabad-India.

4. Chief Operations Officer, SuperMaths, Hyderabad.

Abstract: One of the major challenges faced by users today is the effective utilization of available resources to improve processing speed and manage data efficiently. A key difficulty arises when users operate across multiple resources within a multidimensional environment. These users are often connected to systems with varying configurations, processing speeds, and network connectivity. This heterogeneous pool of information makes it extremely challenging for users to optimize application performance. To address the issue of efficient resource allocation, this paper proposes a technology called the Constructive Data Processor (CDP), which leverages a knowledge engineering mechanism. The proposed technique involves periodically collecting available information resources at regular intervals. When an operation is initiated, the system uses the collected data to identify the most feasible processing paths. These paths are then connected to the Constructive Data Processor, thereby enhancing the overall processing speed of user operations. The system design focuses on regular observing and estimation of available resources. Implementation results demonstrate that the proposed technique significantly improves system resource allocation when compared to existing methods.

Key terms: System resources, Constructive data processor, Knowledge engineering, System Enhancement, Operational performance, Periodical data collections.

1.Introduction:

Data allocation and handling have become significant challenges for many users in modern computing environments. The increasing use of diverse resources and multiprocessor operations complicates these tasks. As digital technologies evolve, users are generating and consuming more digital data than ever before. This explosion in data usage demands not only effective storage solutions but also intelligent and efficient resource utilization. One of the primary issues lies in the heterogeneous nature of user systems. Users operate on devices with varying processing speeds, configurations, and capacities. When clients request information from servers, their systems may lack the necessary capability to process or store the data due to hardware limitations. This mismatch between resource availability and demand creates bottlenecks in performance. Another major challenge is the variety of digital content types. With data now available in multiple formats and structures, users often struggle to access and process these complex datasets efficiently. This issue is especially pronounced in environments where shared resources are expected to function seamlessly across diverse platforms and configurations.

To address these challenges, this research proposes a technique known as the Constructive Data Processor (CDP), which leverages Knowledge Engineering techniques to improve resource allocation and system performance. The CDP collects resource information from connected clients and maintains this data in a central repository. Whenever a task requires resources, the system consults this repository to allocate the most suitable resources immediately, thereby enhancing processing efficiency. However, collecting reliable and up-to-date data from a pool of heterogeneous clients is not a simple task. Client resources and configurations change frequently, and maintaining an accurate, dynamic inventory of available resources is a complex challenge. The proposed solution addresses this by periodically monitoring client systems through a knowledge-based tracking system. This approach ensures that users do not need to perform additional steps to track resources—resource data is collected and updated automatically at regular intervals. system design significantly improves overall processing speed and performance, especially for complex or high-order data operations. By automating resource monitoring and allocation, the system reduces user intervention and allows for more effective data handling. Investigations and experimental results confirm that the proposed method enhances processing speed across different system configurations.

Traditional shared-resource systems suffer from compatibility issues when multiple clients operate on different platforms. Before data can be shared, the system must verify whether the client can manage the shared resources—this often delays the process and leads to inefficiencies. Furthermore, the presence of diverse system tools, development platforms, and network types adds complexity to the resource allocation process. The Constructive Data Processor addresses these issues by transforming heterogeneous client information into a unified, manageable format. This transformation is essential for allocating and connecting systems efficiently to improve processing speed and system responsiveness. The Constructive Data Processor (CDP) addresses the key limitations of traditional shared data processing systems. In conventional systems, client resources are treated independently, and their processing capabilities are not integrated. As a result, these systems lack parallelism and cannot function as a unified processing environment. This leads to reduced efficiency and slower data processing speeds.

In contrast, the proposed CDP system integrates all connected client systems into a single, collective processing unit, regardless of their hardware specifications or configurations. Unlike shared systems that rely on homogeneous resources, our approach supports heterogeneous environments, adapting to each client's capabilities and requirements. The system gathers relevant client information—such as processing power, storage availability, and performance metrics—and utilizes this data to dynamically form a high-performing, unified processing framework. A significant limitation of traditional shared data processors is the latency in resource allocation during high-performance operations. These systems require real-time data collection and validation of client availability before resource distribution can occur. This not only consumes time but also introduces uncertainty, as there is no guarantee that the system will be able to allocate resources efficiently or handle all requested operations.

Our proposed CDP system overcomes this issue through periodic, automated monitoring of all connected clients. The system continuously collects and updates client information at regular intervals. As a result, when a processing request is received, the system already has up-to-date resource data available, enabling immediate and efficient resource allocation. This significantly reduces response time and ensures that the system is always ready to process high-demand tasks without delay. By maintaining a real-time repository of client performance metrics and availability, the Constructive Data Processor functions proactively rather than reactively. This design not only improves overall system performance but also enhances scalability and resource optimization—making it a highly effective solution for modern, distributed computing environments. One of the major strengths of the proposed Constructive Data Processor (CDP) system is its ability to continuously monitor and maintain real-time availability of system resources. By periodically gathering and updating client performance data, the system ensures that accurate and up-to-date information is always available. This real-time readiness significantly improves overall system performance, particularly in dynamic and resource-intensive environments.

The CDP system is built on two critical functions:

1. Regular System Observation – Continuously monitors the status and capabilities of all connected clients.
2. System Forecasting – Predicts which homogeneous systems are available and best suited for current or upcoming processing tasks.

These functions allow the system to make informed decisions about resource allocation, based on the most recent performance and availability data. Because the system tracks resource metrics such as CPU load, idle time, and current task load, it can instantly identify underutilized systems and allocate resources accordingly. The dynamic nature of this data collection enables the system to adapt to changing conditions. For example, if a client system becomes overloaded or underutilized, that information is reflected in the CDP's resource pool. This ensures that the system maintains load balance across the network and allocates processing tasks in a way that maximizes efficiency. Furthermore, the CDP not only identifies which systems are available but also evaluates which systems are heavily loaded, which have fewer tasks, and how resources can be shared effectively. This comprehensive view of the network allows for intelligent time-sharing and load distribution, ensuring smooth and uninterrupted operation even during peak processing demands.

Overall, the timely acquisition of system data and predictive forecasting are key features that distinguish the proposed approach from traditional systems. They allow for proactive, rather than reactive, resource management—resulting in a highly responsive and optimized computing environment. The proposed Constructive Data Processor (CDP) introduces a significant advancement over traditional resource allocation systems. By leveraging real-time data collection and periodic system monitoring, the CDP performs resource allocation automatically and intelligently, eliminating the need for additional manual intervention during both pre- and post-processing stages. Unlike conventional systems that reactively gather data and respond to requests, the CDP functions proactively by maintaining an up-to-date repository of system performance metrics. This approach not only reduces latency but also ensures optimal use of available resources, even in heterogeneous and dynamic computing environments. Through its continuous monitoring, forecasting, and adaptive allocation strategies, the proposed methodology demonstrates clear improvements in system performance, efficiency, and scalability. The evidence from this study supports the conclusion that the CDP significantly outperforms existing techniques, offering a more robust, responsive, and intelligent solution for modern resource allocation challenges.

2. Present system and its challenges.

In existing resource allocation systems, client and resource information is not well organized or segregated, particularly in heterogeneous environments. This lack of structured information poses significant challenges for efficient resource distribution. When a resource request is received, the system often lacks critical data about the client's configuration, hardware specifications, or operating environment. As a result, incompatible resources may be allocated, leading to ineffective execution or even failure to perform required operations. Furthermore, traditional systems typically lack a comprehensive monitoring mechanism. Without real-time observation and tracking of client systems and their available resources, these systems are unable to implement a Constructive Data Processor approach. This not only reduces overall system efficiency but also results in uneven or suboptimal resource utilization. Another major limitation is the absence of detailed information regarding the system components and software configurations of client machines. Without this critical insight, existing systems cannot make intelligent decisions about resource compatibility, which in turn affects system reliability and performance. These shortcomings highlight the urgent need for a more advanced, intelligent, and adaptive resource allocation method—such as the one proposed in this study.

2.1 Main drawback on the present system:

1. No proper surveillance system.
   There is no mechanism in place to continuously monitor clients and system performance.
2. Basic system information is not available.
   Essential data such as hardware specifications, current status, and system capacity is missing.
3. All system-related information must be captured manually by the user.
   This leads to inefficiencies, errors, and delays in resource allocation.

4. No proper mechanism for resource monitoring and control functions.
   The system cannot track or manage resources dynamically in real time.
5. No constructive data processing technique.
   The system operates based on traditional shared resource allocation methods without intelligent coordination.
6. No intelligent decision-making for resource compatibility.
   The system does not evaluate whether allocated resources are suitable for the client's configuration.
7. Client information is not properly organized.
   Disorganized data leads to difficulty in identifying and fulfilling client requirements efficiently.
8. No mechanism for client and resource allocation mapping.
   There is no structured system to match clients with appropriate resources based on demand and compatibility.

2.2 Present system:

The Constructive Data Processor (CDP) distinguishes itself from traditional systems by taking a proactive approach to resource management. Instead of reacting to requests and collecting data on demand, the CDP continuously updates a centralized repository of system performance metrics. This proactive strategy minimizes latency and ensures efficient utilization of resources, even in complex, heterogeneous environments. Through real-time monitoring, predictive forecasting, and dynamic resource allocation, the proposed system achieves enhanced performance, greater efficiency, and improved scalability.

2.2.1 Advantage of the proposed technique:

1. System performance is regularly monitored and updated at periodic intervals.
   Continuous supervision ensures the system remains optimized and responsive.
2. Resource allocation is managed using knowledge engineering techniques.
   Intelligent decision-making enhances allocation accuracy and efficiency.

3. Historical system information is updated periodically,
   allowing the system to access necessary resources without delay.

4. The Constructive Data Processor (CDP) is fully implemented.
   This enables advanced processing and efficient resource management.
5. System data collection overhead is minimized,
   resulting in improved overall system performance.
6. Resource allocation is properly supervised,
   with detailed records maintained on which clients are assigned specific resources.
7. Predictive forecasting is integrated into the system.
   This enables proactive decision-making based on anticipated demand.
8. The CDP continuously updates a centralized performance repository,
   ensuring real-time access to system status and resource availability.

3.Desing outcome of Constructive data processor:

Conventional shared-resource systems often struggle with compatibility issues due to the diverse platforms on which clients operate. Before resources can be shared, the system must verify whether a client can handle the requested data—causing delays and reducing overall efficiency. The challenge is intensified by the presence of different system tools, development environments, and network configurations, which complicate the resource allocation process. The Constructive Data Processor (CDP) overcomes these limitations by converting heterogeneous client data into a standardized, unified format. This transformation is crucial for enabling seamless connectivity and efficient resource distribution, ultimately enhancing processing speed and system responsiveness. Furthermore, traditional systems treat each client's processing capabilities in isolation, lacking any integrated or parallel processing structure. In contrast, the CDP creates a cohesive processing environment, allowing for real-time collaboration across systems and significantly improving operational efficiency and throughput.

The proposed Constructive Data Processor (CDP) introduces a significant advancement over traditional resource allocation systems. By leveraging real-time data collection and periodic system monitoring, the CDP performs resource allocation automatically and intelligently, eliminating the need for additional manual intervention during both pre- and post-processing stages. Unlike conventional systems that reactively gather data and respond to requests, the CDP functions proactively by maintaining an up-to-date repository of system performance metrics. This approach not only reduces latency but also ensures optimal use of available resources, even in heterogeneous and dynamic computing environments. This process is shown in the below figure1.

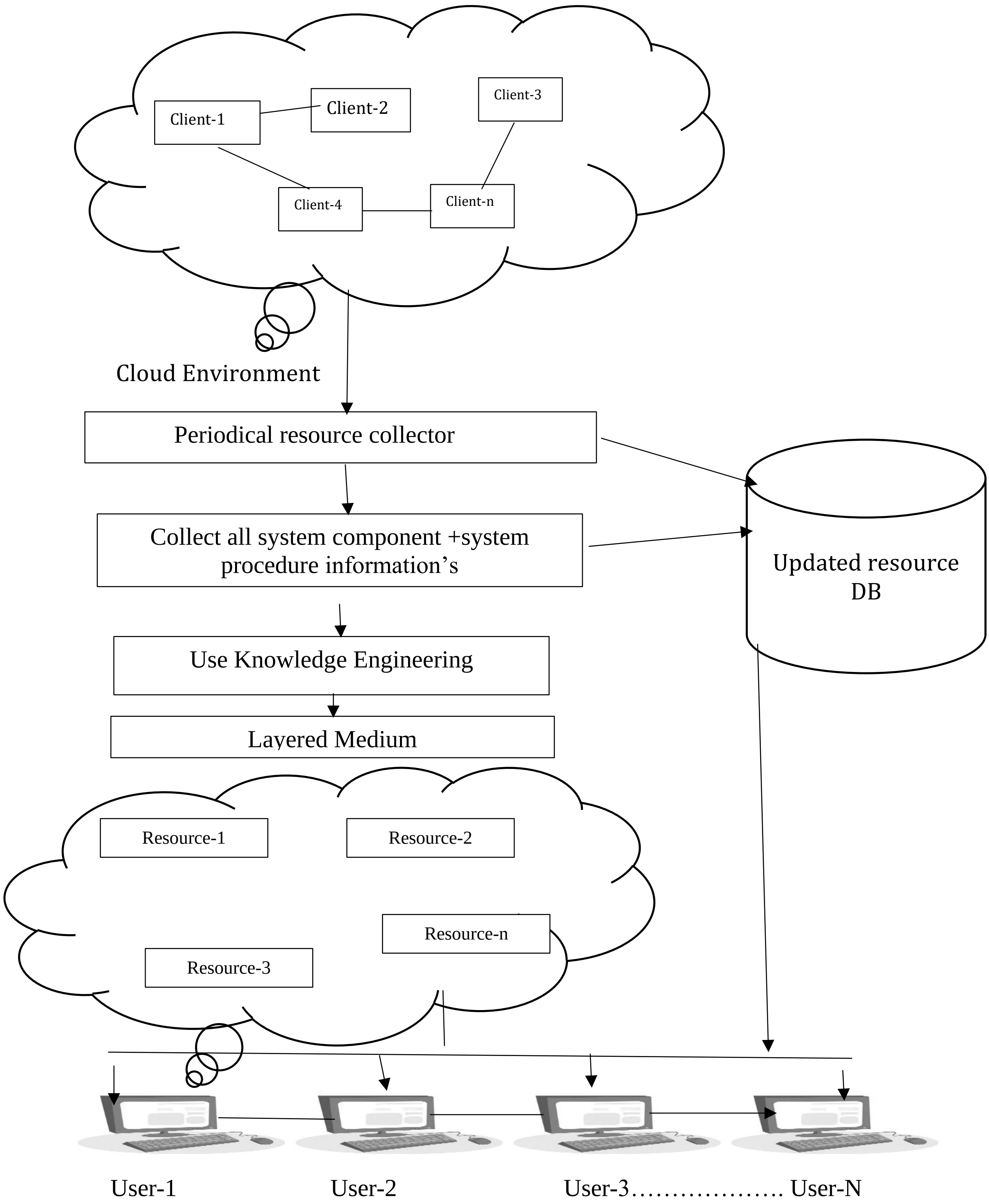

Fig 1 Proposed Architecture

4.Experiemental constructions:

4.1 Layered Medium:

In a networked environment, when information or resources are required, the service request system immediately generates a request to the service provider. The service provider, in turn, receives requests from multiple client machines—each with varying configurations. To ensure compatibility and performance, the system prioritizes delivering services using the highest available configuration, allowing client machines to accept and operate with resources from more advanced systems. Given the diversity of requests from various clients, the system must identify several key parameters: which client is requesting which resource, the nature and timing of the request, and the method of transmission. To manage this complexity, an intermediary layer is essential—one that can capture and analyse all relevant data, including client identification, requested resources, timestamps, and server response details. This intermediary, or layered medium, functions as an intelligent agent between the service requester and the service provider. It facilitates efficient resource allocation by continuously monitoring client requests and server responses, ensuring that resources are delivered accurately and promptly according to system requirements.

4.2 Periodical resource allocator:

The existing shared data processing systems face a major drawback due to the lack of an effective mechanism between service providers and service requesters. When the system is heavily loaded with datasets, there is no efficient method for dynamically redistributing the workload among clients. For proper reallocation, the server must be aware of which clients currently have lower workloads and are capable of accepting additional processing tasks. However, traditional systems do not have this visibility or control. Another critical issue is the frequent generation of high-dimensional data by the system, without any assurance that the receiving client systems can handle or process such complex data structures. This leads to compatibility challenges and reduced efficiency. The proposed solution addresses these limitations by implementing a constructive data processing (CDP) technique, which periodically collects and maintains resource availability information from all connected client systems. When an operation is triggered, the system utilizes this up-to-date resource database to identify optimal processing paths. These paths are then seamlessly connected to the

Constructive Data Processor, significantly improving data processing speed and overall system performance. This approach leverages a knowledge-based tracking mechanism that continuously monitors client systems without requiring manual intervention. As a result, resource data is updated automatically at regular intervals, enabling real-time decision-making for resource allocation. By automating monitoring and allocation, the proposed system not only reduces user involvement but also enhances the handling of complex or high-volume data operations. Experimental evaluations confirm that the proposed method consistently improves processing speed and system efficiency across diverse configurations.

4.3 Collect various system components:

In a Constructive Data Processing (CDP) system, two essential components play a central role: the service requester and the service provider. Effective communication between these two ensures that system resources are accurately identified, shared, and allocated based on the nature of incoming requests. Typically, service providers operate on high-performance configurations, while service requesters may vary significantly in system architecture, hardware capabilities, and software environments. This diversity makes it challenging to construct a unified processing environment. To address these challenges, the proposed system incorporates a mechanism that transforms heterogeneous client systems into a homogeneous processing framework. Before resources can be shared, the system must verify whether the requesting client is capable of handling the data or operations being transferred. This compatibility verification often introduces delays and inefficiencies. Additionally, the existence of various development platforms, tools, and network protocols further complicates the resource allocation process. The CDP resolves these issues by consolidating diverse client data into a standardized and manageable format. This transformation is crucial for efficient system integration and resource distribution. As a result, the system reduces latency and maximizes the use of available resources, even within highly dynamic and heterogeneous environments. By continuously monitoring client systems, forecasting potential resource demands, and adaptively allocating resources, the proposed CDP methodology enhances overall system performance, resource efficiency, and scalability.

4.4 Constructive of knowledge engineering:

Constructive Knowledge Engineering plays a critical role in the proposed technique. In existing systems, most processes are handled manually, leaving the service provider without any reliable information about client systems or their configurations. As a result, when a client submits a request, the system lacks the historical data necessary to determine what type of system the client is using, its capacity, or configuration. Without this information, it is difficult for the server to allocate resources appropriately. To address this issue, the proposed technique incorporates a Knowledge Engineering framework that periodically collects and updates client and system data into a centralized database. This database functions as a dynamic resource pool, containing vital information such as system capacity, configurations, and access history. When a client request is received, the system can immediately access this data and respond efficiently based on real-time information. This automated process eliminates the need for manual data gathering and significantly reduces computational overhead. By providing timely and accurate client information, the proposed system enhances the speed and effectiveness of resource allocation. Overall, Constructive Knowledge Engineering improves processing efficiency and ensures that the system operates in a proactive, rather than reactive, manner.

A lack of structured and timely information significantly hinders efficient resource distribution in traditional systems. When a client submits a resource request, the system often lacks critical data such as the client’s configuration, hardware capabilities, and operating environment. This gap can result in the allocation of incompatible resources, causing inefficiencies or complete task failure. Moreover, conventional systems typically operate without a comprehensive monitoring mechanism. Without real-time tracking of client performance and resource availability, they are unable to implement a Constructive Data Processor (CDP) approach. This leads to imbalanced resource usage and decreased system efficiency. The proposed CDP system addresses these challenges through dynamic data collection and continuous system observation. It actively monitors client status and updates resource information at regular intervals, enabling it to respond quickly to changes in network conditions. If a client becomes overloaded or underutilized, the CDP immediately reflects this in its resource management, ensuring balanced task distribution. Beyond identifying available systems, the CDP assesses each system's load, activity level, and readiness to share resources. This intelligent analysis enables effective time-sharing, improved load balancing, and smooth system performance, even during peak operational demands.

4.5 updated resource Database:

A key feature of the proposed system is the construction and maintenance of a dynamic resource database. Before a service provider can efficiently respond to client requests and distribute workloads, it must have essential information—such as which clients are available to accept the load and their current operational status. This information is critical for achieving balanced and efficient processing within the Constructive Data Processor (CDP) environment. To address this, the proposed system incorporates a Knowledge Engineering technique that stores and updates the historical and operational data of all connected clients. This resource database is continuously updated, collecting availability and configuration data from each client system at regular intervals. When a service request is initiated, the system consults this up-to-date resource pool to identify the most suitable processing paths. These paths are then connected to the CDP, which significantly enhances processing speed and overall system efficiency. The knowledge-based tracking mechanism operates automatically, eliminating the need for manual input and enabling real-time, intelligent decision-making for resource allocation.

5. Experimental outcomes:

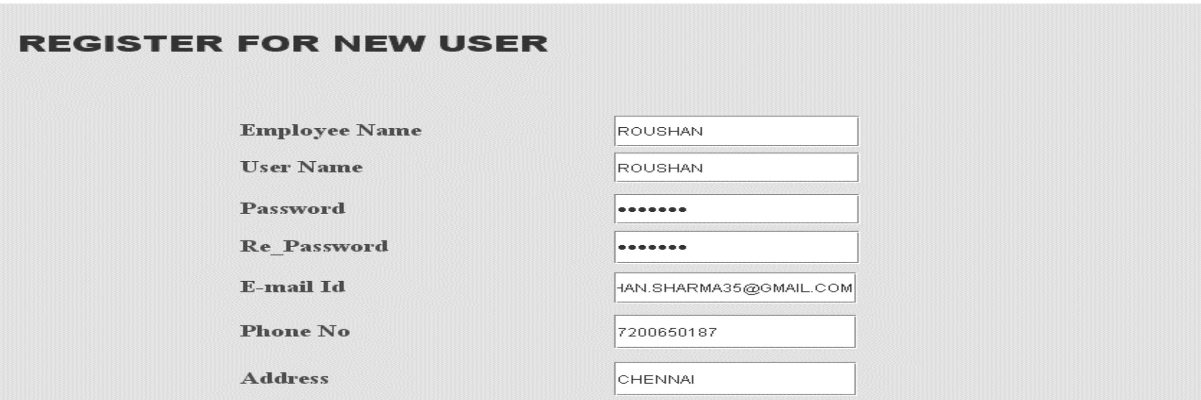

Figure 2. Register the new service request process.

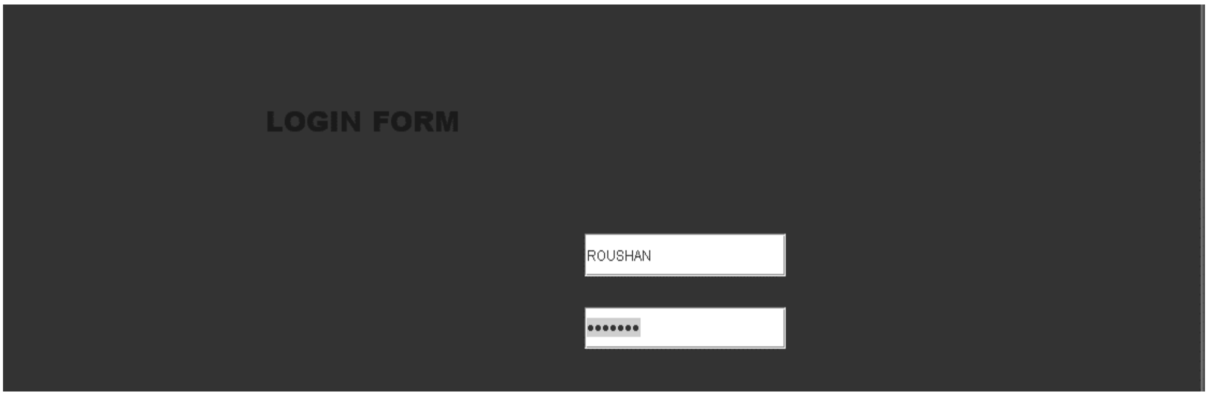

Figure 3. Create a successful Service request credentials.

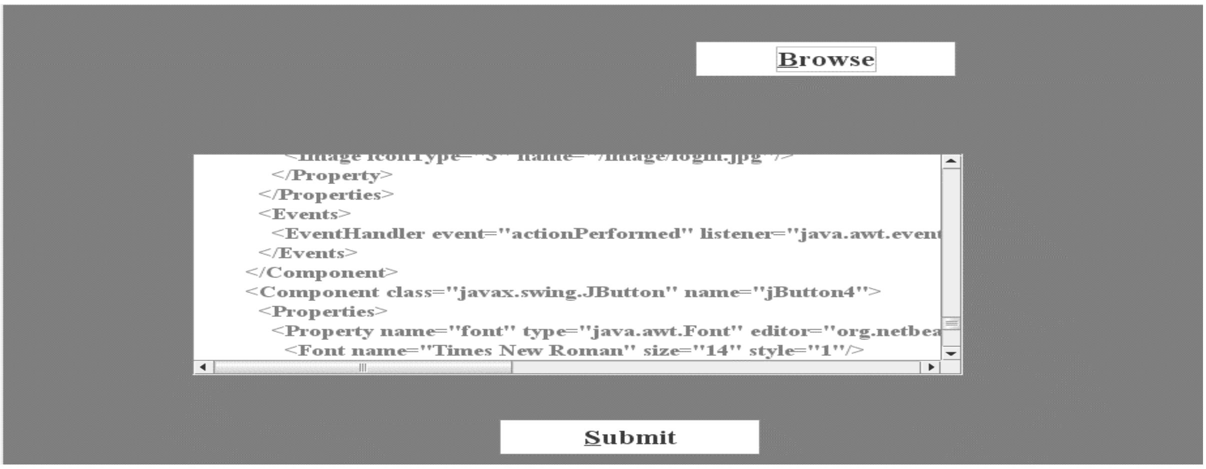

Figure 4. Collect various system components process.

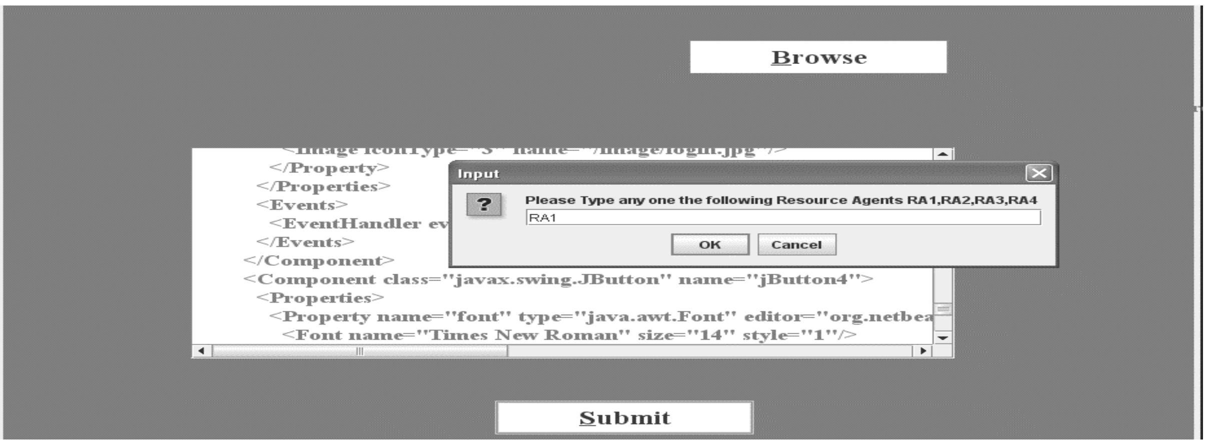

Figure 5. Getting of registered service request details.

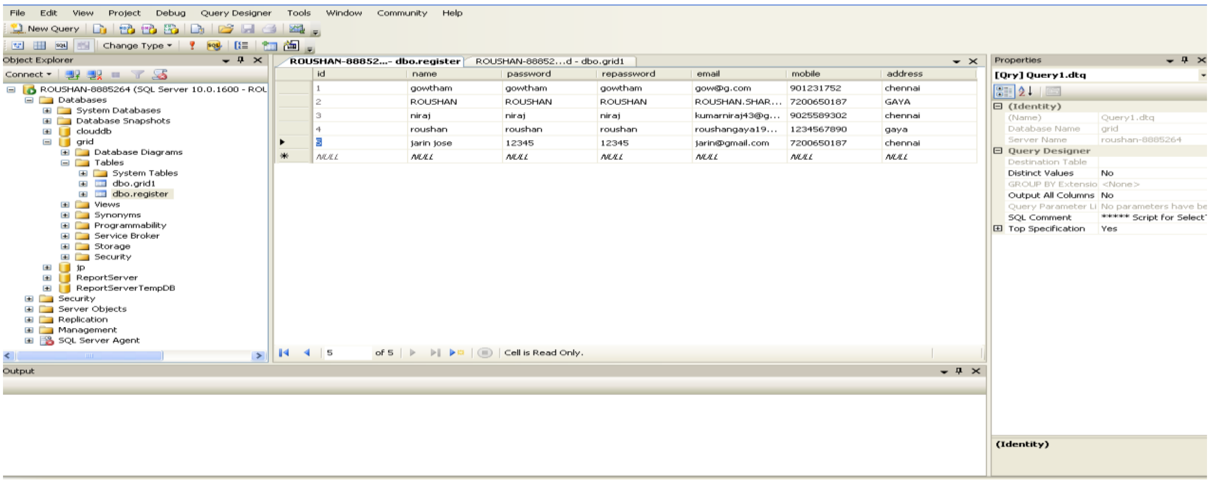

Figure 6.Getting of the complete path details of resisted service requester.

6. Conclusion and future enhancement:

The growing reliance on diverse computing resources and multiprocessor systems has significantly increased the complexity of resource management. As digital technologies continue to advance, users are generating and consuming unprecedented volumes of data. This surge not only demands robust storage capabilities but also calls for intelligent, adaptive resource utilization strategies. Traditional systems often struggle under these demands due to variations in system tools, development platforms, and network configurations, which hinder efficient resource allocation. Furthermore, conventional architectures typically treat each client in isolation, lacking a unified or parallel processing framework, which results in fragmented performance and underutilized resources. The proposed Constructive Data Processor (CDP) addresses these limitations by transforming heterogeneous client data into a standardized, unified format. This transformation enables seamless interoperability between systems, improving connectivity and enabling more efficient resource sharing. By proactively implementing real-time monitoring, predictive forecasting, and dynamic allocation, the CDP ensures minimal latency and maximized resource usage. Ultimately, this approach enhances overall processing speed, system responsiveness, and scalability—making it particularly well-suited for complex, heterogeneous computing environments.

Reference's:

1. L.F. Bittencourt and E.R.M. Madeira, "A Performance-Oriented Adaptive Scheduler for Dependent Tasks on Grids," Concurrency and Computation: Practice and Experience, vol. 20, no. 9, pp. 1029-1049, Jan 2012.
2. Wolski, Rich, et al. "Grid resource allocation and control using computational economies." Grid computing: making the global infrastructure a reality (2003): 747-771.
3. Li, Jie, et al. "Resource orchestration of cloud-edge–based smart grid fault detection." ACM Transactions on Sensor Networks (TOSN) 18.3 (2022): 1-26.
4. D.Saravanan, Dr. Vaithyasubramaniyan, Dr. Dennis joseph "Effective Utilization of image information using Data mining technique" , Intelligent systems reference library 172, Volume 172, Pages 207-215, ISBN 978-3-030-32643-2, Nov 2019
5. KodeDurga Prasad, S.Senthikumar, D.Saravanan," Distributed Scheduling algorithm to bringing fairness to Peer-to Peer Systems", International Journal of Modern communication technologies & Research(IJMCTR), Volume 02-,Issue-02, February 2014, Pages 01-04, ISSN:2321-0850
6. M.L. Massie, B.N. Chun, and D.E. Culler, "The Ganglia Distributed Monitoring System: Design, Implementation, and Experience," Parallel Computing, vol. 30, no. 7, pp. 817-840, July 2011.
7. F. Wolf and B. Mohr, "Hardware-Counter Based Automatic Performance Analysis of Parallel Programs," Proc. Conf. Parallel Computing (ParCo '03), pp. 753-760, Sept. 2003.
8. Jamil, Bushra, et al. "Resource allocation and task scheduling in fog computing and internet of everything environments: A taxonomy, review, and future directions." ACM Computing Surveys (CSUR) 54.11s (2022): 1-38.
9. Dennis Joseph ,Dr.S.Vijaylakshmi ,D.Saravanan, "An efficient local synchronization based algorithm for distributed peer-to-peer Network", In the Proc. Of 9th National conference on Emerging issues in Computer applications,(EICA 2017), Page 16, March 2017, ISSN:0974-1925.
10. D.Saravanan, Dr. Dennis joseph." File allocation among the trustworthiness areas attached to the distributed environment, using server storage testing methods", Journal of informatics education and research, ISSN 1526-4726, Vol 3 ,Issues 2(2023), July 2023 ,Pages 320-326.
11. Belgacem, Ali. "Dynamic resource allocation in cloud computing: analysis and taxonomies." Computing 104.3 (2022): 681-710.
12. D.M. Swany and R. Wolski, "Multivariate Resource Performance Forecasting in the Network Weather Service," Proc. ACM/IEEE Conf. Supercomputing, pp. 1-10, Nov. 2002.
13. A. Waheed et al., "An Infrastructure for Monitoring and Management in Computational Grids," Proc. Fifth Int'l Workshop Languages, Compilers and Run-Time Systems for Scalable Computers, vol. 1915, pp. 235-245, Mar. 2000.
14. Luo, Quyuan, et al. "Resource scheduling in edge computing: A survey." IEEE communications surveys & tutorials 23.4 (2021): 2131-2165.
15. A. Waheed et al., "An Infrastructure for Monitoring and Management in Computational Grids," Proc. Fifth Int'l Workshop Languages, Compilers and Run-Time Systems for Scalable Computers, vol. 1915, pp. 235-245, Mar. 2000.
16. Murad, S. A., Muzahid, A. J. M., Azmi, Z. R. M., Hoque, M. I., & Kowsher, M. (2022). A review on job scheduling technique in cloud computing and priority rule based

intelligent framework. Journal of King Saud University-Computer and Information Sciences, 34(6), 2309-2331.