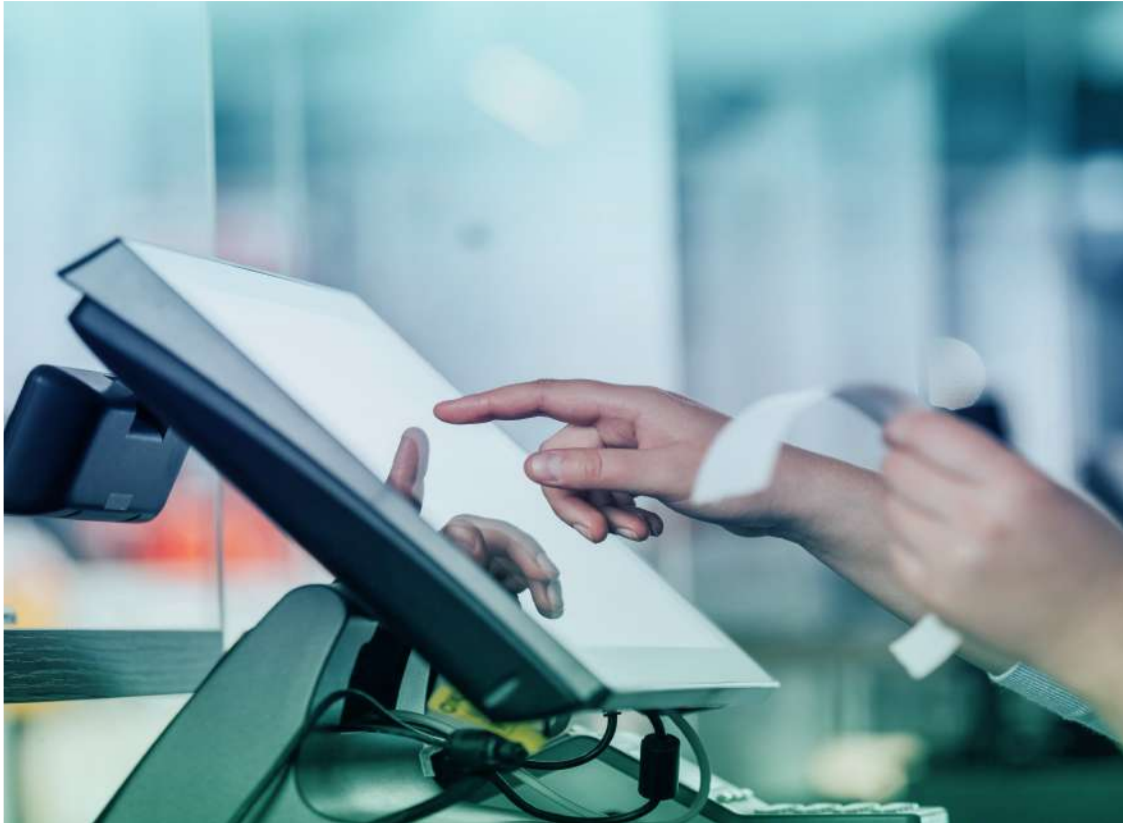


Case Study



“ NCache’s native .NET support made our migration smooth and hassle-free, and its powerful caching capabilities have significantly boosted our application’s performance across the board. It is the only solution that truly fits our needs and is working extremely well for us.”

- Chief Technology Officer

Retail Management Solution Provider

NCache allows this provider to ensure easy and straightforward migration, while enhancing performance and scalability for their retail management solution.

“NCache’s native .NET support made our migration smooth and hassle-free, and its powerful caching capabilities have significantly boosted our application’s performance across the board. It is the only solution that truly fits our needs and is working extremely well for us.”

- Chief Technology Officer

Country or Region:

United Kingdom

Industry:

Retail

Company Profile:

A global retail management solution provider, offering a platform that simplifies operations across multiple channels. It allows businesses to efficiently manage sales, inventory, customer interactions, and supplier relationships in a single, centralized system.

Introduction

This retail management solution provider developed an ERP (Enterprise Resource Planning) platform for their enterprise customers to help manage their business operations (e.g., tracking inventory, processing orders etc.). One of their application's key features is to monitor price changes for products through a series of auctions. As the ERP solution's customer base rapidly grew, the company faced scalability challenges, leading them to adopt load balancing to distribute queries across multiple servers. However, this approach created complications, as the simultaneous updates from multiple users resulted in data synchronization issues, undermining the system's reliability and negatively impacting the user experience. This challenge prompted them to search for data caching solutions, specifically a .NET-native caching solution, to improve performance and enhance scalability in their .NET-based application.

Application Pain Points

Using a load balancer to manage such increased traffic came with its own set of significant synchronization challenges. For instance, customer requests for inventory updates were sent to different web servers when the clients were load-balanced between these servers. This resulted in inconsistent data across servers where some displayed outdated inventory data while others showed recent changes. This inconsistency not only frustrated customers but also hindered their ability to make large-scale inventory updates smoothly, impacting business operations at large.

Additionally, with no caching solution in place, all requests went directly to the database, which not only worsened response times but also limited the system's scalability as customer demand increased. This highlighted the need for a reliable caching solution that could handle real-time updates efficiently. There was no effective solution in place before implementing NCache.

Given that their application is built upon .NET framework, they sought a caching solution that would communicate natively with their application.

How NCache Fits into Their Solution?

NCache was the optimal solution for addressing the application's issues in their distributed server environment. With NCache's CRUD operations, delays in updating data across web servers were no longer a concern. Now, when a customer is adding, inserting, or removing inventory data, the updated data is immediately stored in NCache. This enables other servers to quickly retrieve the latest data using NCache's Get operation, ensuring that all servers have consistent, up-to-date information without needing to query the database directly. Incorporating such a caching tier has reduced database load and eliminated any delays in updating inventory data across all their web servers, providing customers with accurate, real-time information.

Alachisoft Product:

NCache

Customer Needs Met:

- Easy & Straightforward Migration/Deployment
- Data Synchronization
- Performance & Scalability
- Reduced Database Trips
- High Availability

With NCache, the company can now scale their application effortlessly to meet growing customer demands. NCache's Partition-Replica topology distributes data across multiple cache servers, ensuring that data remains available and consistent even if one server fails. This provides high availability and reliability, allowing the application to handle increased traffic smoothly while maintaining consistent performance across all servers.

Moreover, NCache being a .NET native distributed caching solution, integrates smoothly with the provider's existing .NET application. This compatibility eliminates potential integration challenges from using solutions developed in different programming languages, making NCache as the perfect fit for their operational needs.

Application Architectural Overview and Deployment with NCache

The application architecture mainly comprises the following components:

- **Front-end:** The front-end is a WPF (Windows Presentation Foundation) application - a graphical user interface framework used to create desk top client applications. Customers install this application on their PCs to interact with the system.

- **Logic Layer:** The logic layer is built using C# Windows services and utilizes Open RIA Services for communication between the WPF client and the back-end. This layer is responsible for managing core business logic and serves as an intermediary, facilitating communication with both the caching layer and the database.
- **Back-end:** The back-end relies on Microsoft SQL Server as the primary data store.

The architecture includes five web and application servers, with two NCache servers hosted on two of the application servers. Each NCache server is equipped with 4 cores, 8GB RAM, and sufficient network bandwidth to meet the application's demands. For the database tier, there are two servers configured as Availability Groups: one active and one passive where each work together to keep the data available and up-to-date. So, if active server fails, the passive server can take over, without any loss of information. The following diagram illustrates the architecture:

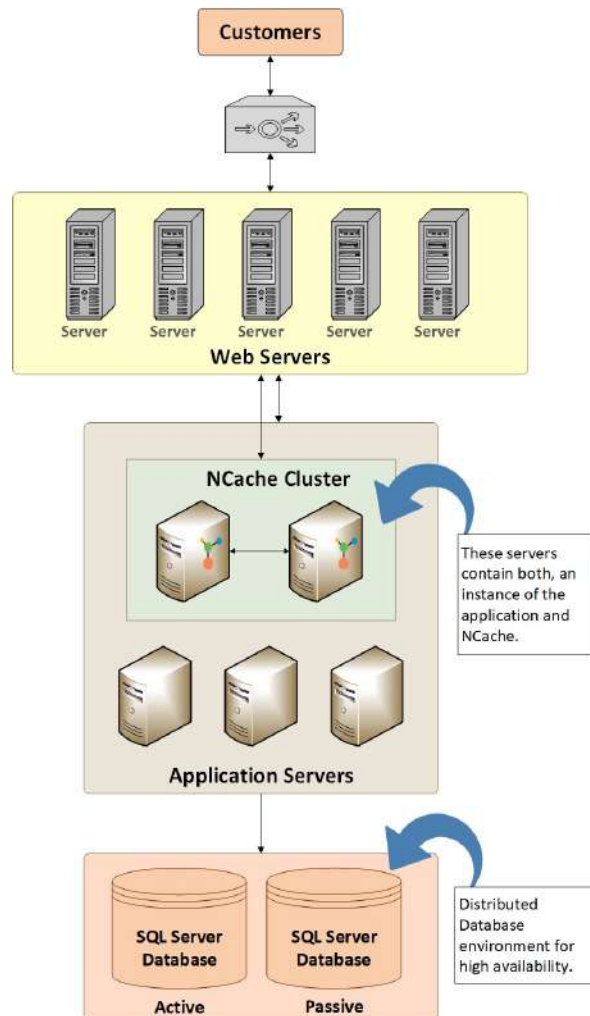


Figure - Architectural Diagram using NCache

The application handles approximately 2,000 concurrent customers under normal conditions, with NCache handling up to 144,000 customer requests per hour during lower peak times. Every time a request is made, the web server either interacts with NCache to retrieve or update cached data or connects directly to the application server as needed.

Given this high request volume, NCache is essential for maintaining consistent data updates and enhancing the overall application performance. Being hosted as a shared deployment on their application servers, NCache minimizes database load, resolves data inconsistencies, and improves application speed. Moreover, by employing the NCache Partition-Replica topology in their application, NCache enhances cluster performance by distributing cached data effectively, ensuring data availability and consistency for customers.

NCache Feature Usage

CRUD Operations for Data Caching

Their application utilizes NCache's CRUD operations. Such operations allow the application to efficiently add, fetch, update, and delete data in the cache, ensuring that all users have access to the most current information. By maintaining data consistency across multiple servers, the application minimizes discrepancies and enhances the user experience, especially during high-traffic periods.

Partition-Replica Topology for High Availability

This topology has effectively ensured high availability and fault tolerance for their application. By distributing data across multiple cache servers with replicas, the application remains resilient to server failures. This architecture guarantees uninterrupted access to cached data, enhancing reliability and high availability while effectively accommodating an increasing customer base.

Improved Performance and Scalability

NCache significantly boosts the application's performance by caching frequently accessed data, which reduces direct database calls. This not only minimizes the load on their database but also improves response times. As user traffic increases and the system scales, NCache's robust caching mechanism enables the application to handle higher volumes of requests without compromising performance.

About Alachisoft:

Alachisoft provides a popular high performance in-memory distributed cache called NCache. NCache is an Open Source middleware that runs in production environment and boosts performance and scalability of .NET web apps, SOA service apps, and general high traffic server apps. Since, 2005, NCache has proven essential to hundreds of customers worldwide, specially in US, UK, and Western Europe.

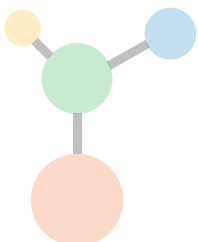
Visit our website at
www.alachisoft.com

You can download a free 30-Day fully working trial of NCache from here:
www.alachisoft.com/ncache

Alachisoft
Corporate Headquarters
12005 Ford Road, Suite 520
Dallas, TX 75234

US: +1 (214) 764-6933
UK: +44 207 993-8327
Fax: +1 (925) 886 8361

Sales Email:
sales@alachisoft.com
Technical Support Email:
support@alachisoft.com



Moving Forward with NCache

This retail management solution provider is enthusiastic to further integrate NCache in their application after experiencing significant improvements. They are keen to explore NCache's Write-through and Write-behind capabilities to optimize their database update processes. Furthermore, they are considering NCache's cloud deployment options to boost scalability and improve access for their customers. They also aim to implement NCache's locking features to maintain data consistency in high-concurrency scenarios.