Case Study

> Website performance is critical for eCommerce. NCache has helped improve the performance and reliability of our website, which is necessary to build and sustain a growing eCommerce store."
>
> - Eric Warezak
> Manager,
> eCommerce Marketing

TruckSpring

NCache allows this organization to maintain high availability and accelerate the performance of their ASP.NET Shopping Cart.

Country or Region:
United States

Industry:
eCommerce

Company Profile:

Established in 1976, TruckSpring is one of Michigan's most well-known truck and trailer parts dealers. They offer spare parts for light, medium, and heavy-duty vehicles. They specialize in truck suspension, towing, and aftermarket parts. Their dedication to innovation and excellence in the automobile industry has solidified their position as a trusted industry provider, serving customers across Michigan and beyond.

> *"Website performance is critical for eCommerce. NCache has helped improve the performance and reliability of our website, which is necessary to build and sustain a growing eCommerce store."*

- Eric Warezak
Manager,
eCommerce Marketing

Introduction

TruckSpring (www.truckspring.com) offers a wide range of products designed to meet all personal and commercial needs of Truck owners. They use an ASP.NET eCommerce solution to fulfill their use-case. This solution provides a vast variety of features to support online businesses, offering customizable storefronts with advanced product management and secure payment processing capabilities.

Application Pain Points

TruckSpring caters to 100,000 users per month. They were experiencing the following pain points which were resolved by NCache.

- **Larger ASP.NET View State Payload:** Handling large ASP.NET View State pages resulted in increased data transfers between the client and server, leading to high network latency and prolonged response times. This degraded application performance, especially during traffic surges, as it caused frequent database trips.

- **Use of Sticky Sessions in ASP.NET:** The use of Sticky Sessions combined with the lack of load-balanced environment neagetively impacted application performance and scalability. This setup resulted in uneven traffic distribution across web servers - potentially overwhelming individual servers with excessive workload.

- **Database Performance Issue:** Heavy transactional loads on the SQL Server significantly increased application response time and degraded overall performance.

To deal with these crucial performance and availability issues, TruckSpring needed a reliable caching solution to better process user requests.

How NCache Fits into Their Soultion

TruckSpring has optimized their ASP.NET AJAX web form application through the robust caching features and load-balanced environment that NCache provides. TruckSpring is now able to manage user sessions effectively and provide a seamless browsing experience with its session management feature. NCache has also empowered TruckSpring to store and manage static application data, reducing page load times and enhancing overall performance through its ASP.NET View State and Object Caching features.

NCache has optimized TruckSpring's operations by distributing incoming traffic effectively across multiple servers. Additionally, using the two-node NCache Replicated Topology eliminated the risk of single point of failure and ensured high availability. Each server maintains an exact copy of the cache, guaranteeing application responses even during server failures or downtime, making the application highly reliable. This assures that the user gets a response even during server failure or application downtime making their application highly reliable. As an in-memory solution, NCache offers a significant performance boost by serving user's requests from within the memory escaping the resource-intensive network trips.

Alachisoft Product:

NCache

Customer Needs Met:

- Improved Application Performance
- Data Availability
- Data Reliability
- Reduced Database Trips

Application Architectural Overview and Deployment with NCache

The application architecture mainly comprises the following components:

- **Front End:** The front end is designed using AngularJS.
- **Web API Layer:** The application data is accessible using NCache or the data-store through ASP.NET Web Forms with AJAX (.NET Frame work 4.7).
- **Back End:** The back end employs SQL Server as the data store.

The current application is deployed within a single data center. It consists of two web servers with a load-balanced production environment. The deployment is shared between NCache and their application on both servers with a dedicated database server (SQL Server). This infrastructure caters to numerous monthly visitors and comprises 64GB of RAM, 6 cores, 12 virtual CPUs, and Intel Xeon processors on each server. They are currently employing NCache 5.3 SP2 with two distributed caches called TSCache and myBVCache. TSCache is configured to manage the ASP.NET Session State and View State, whereas myBVCache is configured for Object Caching. The following diagram depicts this application architecture visually:

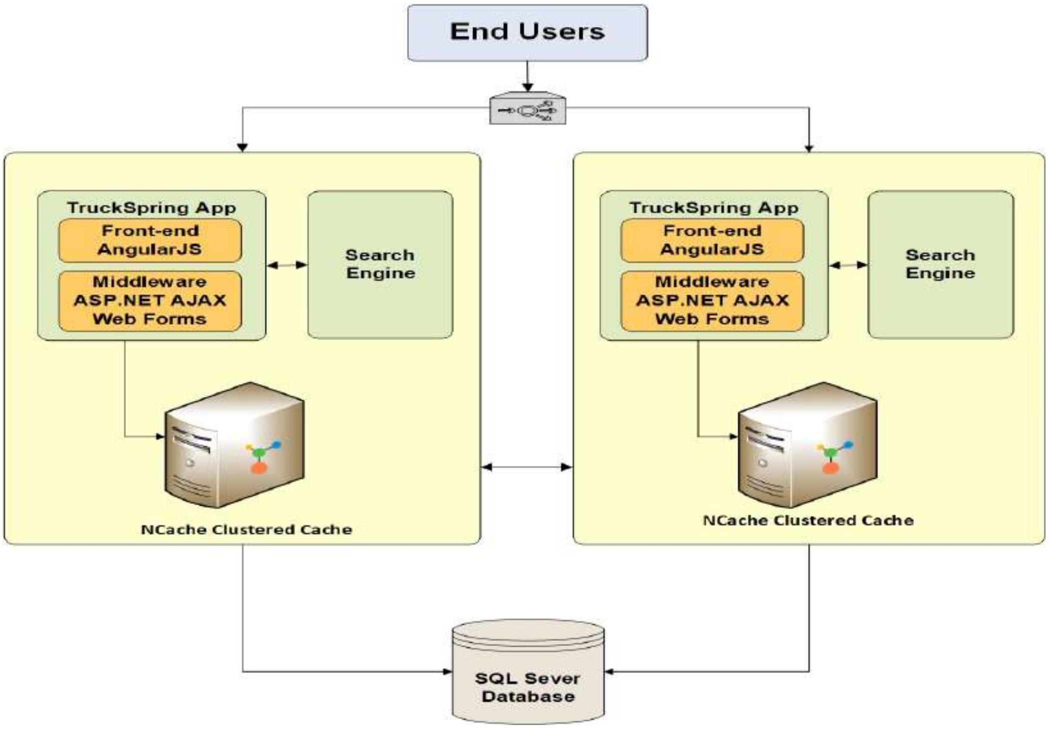

Figure - Architectural Diagram using NCache

NCache Feature Usage

Enhanced User Experience with ASP.NET View State Caching

As explained previously, one of the pain points of TruckSpring was the management of ASP.NET View State pages and their traffic. The larger ASP.NET View State pages lead to increased data transfer between the client and the server node, contributing to high network latency. Moreover, the transmission of such large pages caused delays and impacted the user experience. By caching ASP.NET View State data on the server-side, and reducing the size of the ASP.NET View State payloads NCache has minimized the application's response time.

Furthermore, NCache has enabled TruckSpring to compress the ASP.NET View State before network transmission resulting in faster transmission times. As their eCommerce Marketing Manager stated, "Reducing the page size has also reduced traffic on our CDN and reduced costs. Now, we are better prepared to scale our infr-astructure as traffic increases with NCache."

About Alachisoft:

Alachisoft provides a high-performance in-memory distributed cache called NCache. NCache is an Open Source middleware that runs in a production environment and boosts the 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, especially in the US, UK, and Western Europe.

Visit our website at
www.alachisoft.com

You can download a free 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

Improved Performance with ASP.NET Session State Caching

TruckSpring handles hundreds of user transactions daily, maintaining an equivalent number of user sessions which have significantly slowed down their application functionality. Along with application functionality delays, there was no failover support due to the use of Sticky Sessions.

Fortunately, NCache’s ASP.NET Session State Management feature configured along with the Replicated caching topology, is the optimal fix to address these performance issues. Using the NCache Session Cache (TSCache) saves the users' session data as a cache item in its in-memory store.

Thus, when the same user logs in next time, the application will have the relevant session data (username, password, favorite items, cart items, etc.,) from the cache instead of the SQL server - offloading the session state from the database. Moreover, the Replicated topology creates a replica of this session information and stores it on the other node. This ensures a backup in case of server failures without affecting the application’s performance. Hence, eliminating the use of sticky sessions.

Scalability with Object Caching

Along with ASP.NET Session Caching, TruckSpring also caches the website catalogs, menu bars, category structures, and other website elements using NCache Object Caching. Before incorporating NCache as part of their application, all user requests went to the SQL database and fetched results from there. These frequent database trips became a bottleneck which limited application scalability, as the addition of every menu item or catalog further slowed down the application. Now, the most visited catalogs and product categories are cached in myBVCache. So, when a user clicks on a catalog, the request is sent to this cache, and the desired items are fetched from there, instead of the database. This reduces the access time and improves application responsiveness.

Moving Forward with NCache

NCache is exactly what TruckSpring needed to fulfill their performance and scalability needs. They are looking forward to extending their integration with their UPS application module where they are currently retrieving UPS rates via database. The module is currently facing performance issues and they intend to integrate NCache in this module to resolve these issues. They are eager to explore how NCache SQL queries can further enhance their application deployment.