

**Java AJAX Web 2.0  
SKU Consolidation  
& Web Application for a  
Large Auto Parts Retailer**

Q3 developed a solution for SKU Consolidation & Web Application for a Large Auto Parts Retailer.

# Case Study - Java AJAX Web 2.0 - SKU Consolidation & Web Application for a Large Auto Parts Retailer

## Company Profile

The client is a large auto parts retailer.

A large auto parts retailer wanted to consolidate its inventory details spread across various formats so that it could provide a single interface for searching and sale of items from its inventory.

## Business Situation

## Challenges

- » Retailer had over 50,000 SKUs in their catalog, and the list was growing.
- » Auto parts details were spread across various data formats and had to be consolidated into a single, consistent format.
- » The retailer needed an automated mechanism for entering new SKUs into the inventory database.
- » A Web based UI application was required for searching and sale of specific auto parts.
- » The architecture of the web based application had to be very secure and reliable.
- » Response times had to be very fast for searching of the inventory.
- » The application had to be built using Open source standard based technologies.
- » On exiting the soft-phone the application should unsubscribe for the presence information for all the contacts listed in the buddy list

- » An n-tier Java/J2EE architecture was developed that was modular, highly scalable, and easy to maintain.
- » A common data model and database schema was defined for parts details using advanced data modeling tools.
- » Business Services and Data Layers were created for querying and updating data in the inventory database. All access to the database had to go thru the business and data layers.
- » Web Services/WSDL endpoints were created both for external suppliers to send data to the inventory database as well as for internal tools to import data from the existing formats to the common database.
- » A Web2.0 style AJAX based Web based application was created that provided the user with a rich user experience.

## Solution

- » Software utilities were created for reading the existing parts information and migrating it to the database using the Web Services described above.
- » Shopping and billing modules were created including interfacing with external systems such as payment gateways etc.