

## Information Repository Utilizing SQL's DTS with JD Edwards: How We Did Data Warehousing Without Breaking the Bank (or our Implementation Schedule)

By Sharon Petrella, Maja Mitrovic, and Joel Foote of Dealer Tire, LLC.



*Editor's note: We had the good fortune to do some consulting work at Dealer Tire recently; and it is an exciting company. Rapid growth, an emphasis on web based solutions, and strong CIO leadership combine to make this a great environment. Data Warehousing and Business Intelligence are hot buzzwords these days. Instead of buying a packaged solution, Dealer Tire took a different path—using technologies that were already in place, they came up with an economic solution with high ROI.*

Please note that although this article features Dealer Tire's EnterpriseOne ERP solution, the same techniques can be applied to a PeopleSoft World environment.

### Introduction

Dealer Tire, LLC, a Wholesale Tire Distribution Company, implemented Peoplesoft's EnterpriseOne product in January of 2003 in just 9 short months. In brief, the implementation included: seven custom Web Interfaces, Electronic Data Interchange (Customer and Supplier), Data Collection in eighteen Warehouse Locations, Custom Forms, and Data Conversion. The one area that we failed to accommodate our end-users with was the ability for them to have a self-service tool for reporting and analytics.

As with any new system implementation, there were more functional demands and requirements than our Business Budgets would allow. One of the key requirements was to allow the business

users access to the information; not just for transacting business, but for utilizing the information in any form as well. We wrote over forty reports in the EnterpriseOne Toolset (RDA) prior to going "live". It was both difficult and tedious to get a large number of reports written in a short period of time. It was also very expensive since we did not have the in-house expertise to write the reports; all of our resources were focused on other assignments.

Once we went "live", the user community had another fifty reports that they wanted implemented that we could not give them in the timeframe and required. We took a step back and reviewed the requirements to see what we could come up with for the least cost and time possible.

Our answer to the problem was the following:

- Take a current legacy piece of hardware (Large Dell Server)
- Utilize an existing SQL Database
- Create an "Operational Data Store" (ODS) on the SQL Database
- Make use of the Data Transformation Services (DTS) tools within SQL as the Extraction, Transformation and Load Tool (ETL) from Enterprise One
- Create a Reconciliation Process between Enterprise One and the ODS

- Move the Table Structure required for Functional Reports
- Provide End-User tools – Excel, for easy design and user accessibility via Outlook Public Folders

In less than sixty days, we had a new, secured, reconciled, self-service information repository (most people would call it a Data Warehouse), for which we had to spend only an additional \$30,000. It is our platform for Reporting and Analytics, and it gets the job done.

### Functional Overview

The data warehouse answers the business users' demands for self-sufficiency by providing business information in a relatively timely fashion. With the ability to capture information not available in the system, the data warehouse reduces islands of information located on the users' local computers and shared network drives, providing a single, consistent place to store corporate data.

### Data Delivery

The tables within the data warehouse can be accessed by the business users through an ODBC connection on their local computer, giving them the ability to retrieve and manipulate data in either Microsoft Excel or Access (both are part of the standard build on our corporate workstations). While our business community has both tools available, Dealer Tire has chosen to use

# JDEtips on Data Warehousing Journal

Microsoft Excel combined with Microsoft Query as our reporting foundation. Excel spreadsheets, charts and pivot tables are then utilized as presentation instruments.

Distinct resistance was felt from some business users (mostly power users who frequently generate their own reports) opposing the choice of Microsoft Excel as the reporting application. They had grown accustomed to and familiar with the Access toolset and could more readily leverage the data in the data warehouse with MS Access. Although we considered the power and ease of use Access provides, Excel was chosen over Access as a development tool for three main reasons:

1. More efficient network utilization (compared to Access)
2. Inherent data-packaging and presentation capabilities

3. Mainstream business user familiarity with Excel

**Network Utilization** – As a general rule, Excel queries use less network traffic than those issued from Access. Whereas Access frequently retrieves the entire table into memory before executing statements and commands, Excel passes the query to the database and transfers only the results of the query across the network. In addition, Excel more efficiently utilizes workstation memory, allowing the user's PC to realistically multi-task while using Excel to query the data warehouse. This contrasts with Access where most workstation memory can end up dedicated to holding the tables that were read as part of the query.

Using Excel also allows physical storage and network traffic efficiencies. The disk space required to store an Excel bound query to a pivot table is less than the same query in Access because Excel has

the ability to store just the query logic without the associated data.

**Data Packaging and Presentation** – The pivot table component of Excel allows users to analyze data by adding sorting, grouping, filtering, outlining, and other data-manipulation capabilities. It also provides the following functionality:

- Cross-tabulation (the ability to rotate the rows and columns to see different summaries of the source data)
- Browse report data
- Dynamically filter (AutoFilter, Filter by Selection, Filter fields) and sort
- Group by row or by column
- Create totals
- Work with large or small amounts of data

Sales	Product Line	Vendor	Brand	Item	Qtr	Qtr1	Qtr2	Qtr3	Qtr4	Grand Total
	Accessories					\$28,931	\$26,472	\$34,735	\$42,319	\$132,457
	Assemblies					\$85,937	\$9,425	\$23,483	\$359,071	\$477,916
	Tires					\$10,276,049	\$11,882,313	\$13,449,571	\$14,913,040	\$50,520,973
	<b>Grand Total</b>					<b>\$10,390,918</b>	<b>\$11,918,210</b>	<b>\$13,507,789</b>	<b>\$15,314,429</b>	<b>\$51,131,346</b>

Figure 1: Sales Analysis Report

## This Article Continues...

**Subscribers**, log in from our main search page to access the full article:

[www.JDEtips.com/MyAccess.html](http://www.JDEtips.com/MyAccess.html)

**Not a Subscriber? Gain access to our full library of JDE topics:**

[www.JDEtips.com/JD-Edwards-Library](http://www.JDEtips.com/JD-Edwards-Library)

Visit [www.JDEtips.com](http://www.JDEtips.com) for information on the JDEtips University schedule, private training and consulting, and our Knowledge Express Document Library.

License Information: The use of JDE is granted to JDEtips, Inc. by permission from J.D. Edwards World Source Company. The information on this website and in our publications is the copyrighted work of JDEtips, Inc. and is owned by JDEtips, Inc.

NO WARRANTY: This documentation is delivered as is, and JDEtips, Inc. makes no warranty as to its accuracy or use. Any use of this documentation is at the risk of the user. Although we make every good faith effort to ensure accuracy, this document may include technical or other inaccuracies or typographical errors. JDEtips, Inc. reserves the right to make changes without prior notice.

Oracle and J.D. Edwards EnterpriseOne and World are trademarks or registered trademarks of Oracle Corporation. All other trademarks and product names are the property of their respective owners.

Copyright © by JDEtips, Inc.