

ActiveBatch®

Customer IT Automation Success Story

SUBWAY® Adopts an “Architectural” Approach to Automating Teradata



Company: SUBWAY®

Industry: Food & Beverage, Restaurant

Customer Site: Milford, Connecticut, United States

Brief Company Overview:

SUBWAY® is an American fast food restaurant franchise that primarily sells submarine sandwiches and salads. SUBWAY® is one of the fastest growing franchises in the world with 39,517 restaurants in 102 countries and territories. It is the largest single-brand restaurant chain globally and is the second largest restaurant operator globally.



SUCCESS STORY HIGHLIGHTS

- Centralize disparate technologies, databases, applications, and more
- Dynamically manage workflows and dependencies from a central console and break complex workflows into “manageable chunks”
- Run Teradata utilities directly from ActiveBatch and centralize processes
- Improve report deliverability and increase staff efficiency

Orchestration of Complex Processes and Dependencies

As one of the fastest growing restaurant franchises in the world, SUBWAY® relies heavily on the timely processing of operational and sales data from its more than 30,000 restaurants worldwide. The foundation for the timely arrival of this information is a data warehousing project that relies on information being passed between a collection of data sources, legacy platforms, and a Teradata Data Warehouse Appliance that was incorporated to address growing data volumes. To orchestrate these complex processes and manage dependencies between these systems, SUBWAY® relies on ActiveBatch® Workload Automation and Job Scheduling from Advanced Systems Concepts, Inc.

The original catalyst for deploying ActiveBatch came in 2010 prior to SUBWAY® implementing the Teradata Data Warehouse Appliance. At the time, the company was relying on a disparate collection of SQL servers to schedule SSIS and DTS packages to integrate a legacy OpenVMS system, a newer .NET/SQL platform, and a data warehouse solution. “Being able to dynamically manage the dependencies between SSIS and DTS jobs was critical, because we couldn’t run the data warehouse load until the OpenVMS and .NET platform was populated and synchronized,” says Ciana Barrueco, Database Administrator at SUBWAY®.

Automating SQL Server jobs across disparate machines was difficult because SQL Server Agent provides no ability to manage dependencies between different servers or start/stop a job within a workflow. As a result, the SUBWAY® database team used “lots of manual handoffs and flag waving” to manage dependencies, Barrueco says.

“To try to streamline the process, one step of each SQL job was supposed to pass data to a job on another server, but we were still literally jumping from one server to the next trying to manage everything.”

A better solution was needed, so in 2010 SUBWAY® went to market for an enterprise job scheduling solution to unify and automate its batch processes. Ease of use, cross-platform support, and competitive pricing were par-amount. Microsoft Orchestrator was also considered, but in the end ActiveBatch was selected, thanks in part to a flexible pricing/licensing model and ActiveBatch’s Integrated Jobs Library, which provided the team with an unparalleled graphical interface of workflows and dependencies between jobs.

“The [SUBWAY®] team would spend approximately 10 hours a week managing and updating the data loads, managing the production and development environments, and scheduling backups. With ActiveBatch that’s been reduced to less than 4 hours a week.”

- Ciana Barrueco, Database Administrator at SUBWAY®

Learn more at ActiveBatch.com ►

Deployed later that year, the number of ActiveBatch end-users has since grown to include 15 members of the SUBWAY® database team, all of who have realized the benefit of using ActiveBatch over the previous system of scheduling jobs across SQL servers. With ActiveBatch, the team can dynamically manage workflows and dependencies from a central console and has broken these complex workflows into “manageable chunks” that can be dynamically managed based on event triggers, such as a database being updated or an FTP being sent/received, something the team completely lacked when scheduling via SQL Server. “We can now re-execute pieces of the workflow if it’s necessary, as opposed to rerunning the entire workflow from the start,” Barrueco says. Overall, Barrueco estimates that using ActiveBatch requires 75% less time over using SQL Server Agent in terms of building new workflows, updating existing ones, time saved from manually managing dependencies, and more reliable execution of processes.

Enter Teradata to Meet Data Volumes

In 2011, SUBWAY® embarked on a project to implement a Teradata Data Warehouse Appliance to enable the IT organization to meet projected data volumes over the coming years. Initially, SUBWAY® investigated various ETL scheduling tools to automate the Teradata environment, but it became obvious that using ActiveBatch would be ideal since any Teradata loads would be dependent on the batch processes already automated within ActiveBatch. “In terms of streamlining the end-to-end automation of our data warehousing processes, it was clear that using a single automation solution was the best approach,” says Leslie Kosturko, Senior Data Manager at SUBWAY®. “From a reporting standpoint, these processes would be responsible for feeding data downstream to a MicroStrategy BI solution for sales and profitability reports, operational reports from the various franchises and HQ-level reports.”

Report Deliverability Reaches 99.5%

“In terms of streamlining the end-to-end automation of our data warehousing processes, it was clear that using a single automation solution was the best approach.”

- Leslie Kosturko, Senior Data Manager at SUBWAY®

Advanced Systems Concepts worked with the Teradata Solutions Architect team to build the Teradata Extension, which added a series of production-ready Job Steps for common Teradata utilities to the ActiveBatch Integrated Jobs Library, including *StartPTJob*, *LoadData*, *SQL Query*, and *ARC*. Two critical requirements for SUBWAY® included the ability for ActiveBatch to run Teradata Parallel Transporter and SQL scripts directly from within ActiveBatch. In addition, the ability to schedule and run ARC-MAIN backups utilities was critical. "The Teradata Database doesn't have a traditional backup/recovery tool," says Gerry Brinkman, Senior Solution Architect at Teradata. "The utility itself is a relatively low-level, script-driven utility that has no policies and procedures around it, so you really need a scheduler to automate and manage those processes dynamically. The ActiveBatch Extension for Teradata really added that automation framework to allow SUBWAY® to automate key Teradata processes."

Less Scripting, More Drag-and-Drop Automaton

The production-ready Job Steps within the Teradata Extension has saved the SUBWAY® database team the time and resources of scheduling these utilities via a command line script. For example, the *StartPTJob* provides the team with a field to enter the database script and provides dropdown menus for job variables and parameters that are auto-populated based on variables that are passed from preceding jobs. "The ability to display the entire workflow and all the dependencies and variables that are being passed in a graphical format dramatically simplifies the management and updating of these processes, saving us time," says Trevor McDonald, database administrator at SUBWAY®.

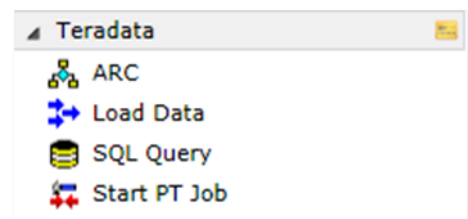
The database team also leverages ActiveBatch's event automation architecture and Flow Control Job Steps that are part of the Integrated Jobs Library to dynamically manage the execution of the Teradata workflows. The event automation architecture goes beyond the static nature of date/time scheduling and allows jobs and workflows to be triggered and managed based on IT and business events, such as an email or database trigger, file

"Teradata doesn't have a traditional backup/recovery tool. The utility itself is a relatively low-level, script driven utility that has no policies and procedures around it, so you really need a scheduler to automate and manage those processes dynamically."

- Gerry Brinkman, Senior Solution Architect at Teradata

constraints and others. For example, Barrueco leverages ActiveBatch job constraints and variables for the execution of SQL Server SSIS jobs to ensure all rows within a SQL table are populated before proceeding to the next job within the workflow, which helps to ensure downstream data quality.

ActiveBatch® Integrated Jobs Library Job Steps for Teradata



The Flow Control Job Steps such as *ForEachRow*, *If-Else*, and *If-Branch* are used to manage and direct jobs within workflows based on conditions and dependencies, allowing users to simply drag-and-drop a Flow Control Job Step into the workflow in the appropriate location. For example, SUBWAY® uses the *ForEachRow* Job Step prior to a SQL Server Job Step to execute a SQL command to retrieve each row of values from a SQL table to be passed downstream to Teradata. Other Flow Control Job Steps set a conditional branch to allow jobs to be looped.

Job reusability within ActiveBatch has also proven invaluable. After the initial implementation of Teradata, the database team built the workflows within ActiveBatch to do a parallel load of both the development and production environments, but quickly realized that to ensure the production environment was working with valid data, the data loads should be done separately. "When we switched the loading of the development and production environments into separate workflows within ActiveBatch, many of the jobs within those workflows shared common variables, steps, etc.," Barrueco says.

To save time in the development of these new workflows, Barrueco leveraged ActiveBatch Reference Jobs, which gives a user the ability to use a single job within ActiveBatch as a template to be "referenced" throughout multiple workflows. "That was great, because it meant we only had to make the changes once at the template level and those changes are passed down automatically to the referenced jobs. It's a big time saver."

The Benefits

Overall, the database team estimates that managing these data warehousing processes within ActiveBatch using the Integrated Jobs Library, event automation framework, and other capabilities is twice as fast as using command line scripts to execute these utilities, according to Barrueco. "The team would spend approximately 10 hours a week managing and updating the data loads, managing the production and development environments and scheduling backups," Barrueco says. "With ActiveBatch that's been reduced to less than 4 hours a week."

Kosturko says that using ActiveBatch to manage the end-to-end automation of the underlying ETL and data warehousing processes that feed the franchise profitability and operational reports has improved report deliverability to 99.5%.

Moving forward, Kosturko only expects these data warehousing processes to become increasingly complex as this franchise-based business expands globally and reporting becomes increasingly intricate. "Data volumes are only going to increase. We'll need to be able to meet those demands and ActiveBatch will be the automation solution to automate our datacenter."

"The ability to display the entire workflow and all the dependencies and variables that are being passed in a graphical format dramatically simplifies the management and updating of these processes, saving us time."