Case Studies and Examples for using Data Compare

Remote Backup and Configuration Maintenance

Problem:

Our company has a Web site hosted at a remote location. The hosting company maintains and supports the Microsoft SQL Server that is the back end for the Web application. However, the hosting company does not provide any capabilities to backup the database on demand and to restore if needed. Also, we need some of the data collected from the Web site locally to analyze user traffic and preferences. Another problem was that the Web site is driven by some of the configuration tables in the database that we need to maintain.

Solution:

With SQL Studio Data Compare we were able to connect remotely to the hosting company SQL Server. Using the functions to synchronize with a local database we could pull all data from the Web site into our local database. Then our team was able to utilize the local database to run usage reports. To solve the other problem with maintaining the configuration settings of the Web site we created a synchronized copy of the database on our development server. Then we would make changes to the development database, test, and if successful use Data Compare to synchronize the changes to the Web site database.

Synchronization between Databases

Problem:

Our development team maintains three different databases. The Development database is used by developers to work on new features and to make updates. Our Test database is managed by the QA team testing all changes before being deployed to production. The production database gets updates only with changes approved by the QA team. To create dynamic content and to simplify maintenance our application utilized a lot of nomenclature tables. All those tables drive the functionality, formulas, and content of the application. It becomes a very difficult task to merge the new changes to test and production and to keep track of what the differences are.

Solution:

Utilizing Data Compare we were able to save projects defining compare and synchronization between the Development and Test databases, and between the Test and Production databases. The development team would use the saved project between Development and Test to compare the content of the Development and the Test databases and to merge any new changes. The QA team would use the saved project between Test and Production to compare changes and to roll out new updates to the Production database.

Data Merge Solution

Problem:

We have multiple branch offices that collect data from our customers. All data is identified by the branch office code, but has the same structure. We could not utilize a central storage for our data since the cost of the high speed connectivity was too high and unnecessary. In order for our central office to analyze the customer data and to keep track of inventory, we needed to pull all data from the branch offices centrally once a day. Also, we have some centrally updated information like rates that has to be uploaded to each branch office on regular bases.

Solution:

Using Data Compare we created projects defining synchronization between each branch office database and the central office database. Every day we compare the differences and pull the new data from the branch office. At the same time we check the differences in the rate tables and upload to the branch office any changes or updates. For our most critical data that is on-line during the day we configured a scheduled task in Windows to execute SQL Studio Data Compare command line batch file to generate change scripts and to execute on all remote servers at specified time in non business hours.