

# Getting started

DB Change Manager provides features to help you manage databases from development to deployment and maintenance cycles. These features include:

- [Data comparison and synchronization](#)
- [Schema comparison and synchronization](#)
- [Database configuration management](#)
- [Reports, notifications, and change history](#)
- [Data masking](#)
- [Source control system integration](#)

## Data comparison and synchronization

DB Change Manager can analyze and compare the data in two different data sources. It generates a SQL script that can update one data source so that it matches another. You can modify the SQL and run it as you wish. You can even define a comparison and synchronization process, called a job, and set it to run automatically. These features help you better manage common scenarios, such as:

- Keeping primary and backup data in sync
- Keeping up with data changes during the development and testing cycles, and later on to deployment environments
- Managing data migrations during upgrade activities

## Schema comparison and synchronization

DB Change Manager can capture and compare the database schema in your enterprise. Over the lifecycle of a single schema, you can capture a baseline or archive schema and analyze changes through time. In addition, DB Change Manager generates a SQL script that can update a target schema so that it matches a standard. You can modify the SQL and execute it as needed.

## Database configuration management

In addition to schema, there are configuration settings that can vary across an enterprise. DB Change Manager can capture the configuration settings of a data source. You can capture a baseline or archive configuration and analyze changes through time. You can create a configuration standard that defines the norms for your enterprise and against which you can compare your databases.

## Reports, notifications, and change history

DB Change Manager supports your data management activities with notifications when processes execute, reports that capture key information, and change history tracking.

## Data masking

There are specific scenarios when you need to hide the true values of the data in a database. For example, during development you need to protect personal or proprietary data, yet you still need to work with data that accurately represents the source. By combining data synchronization with data masking features, such as randomization or shuffling, you can duplicate source data and use the masked copies in non-production or non-secure environments.

## Source control system integration

You can integrate DB Change Manager with source control systems such as Concurrent Versions System (CVS), Visual Source Safe (VSS), or Eclipse plugins. The Navigator in DB Change Manager is the interface for accessing files under source control. Job files and archive versions are stored as XML files and synchronization scripts are stored as text files. You perform all source code control actions using the source control system.