# State v Migrations based Database Development

There are two philosophically different ways of dealing with database development. These are state based database development and migrations based database development. Each of these work well and can be used for most of your database development. Each also has flaws or limitations that you may need to work around.

In the next two sections, we’ll look at how each of these two development models works, along with the advantages and disadvantages that you may face when using either of them.

Note that we will explore the issues of development related to relational databases that typically allow various types of code inside of the platform. Various other types of databases, such as document databases, graph databases, etc., are not discussed in this course.

# What is State Based Migration?

A set of database code exists in a state at any point in time, just like any application code. However, unlike application code that can be completely removed and rebuilt, parts of a database must be transitioned from one state to the next. This is true of tables, but potentially other objects in relational database systems.

As a result, we cannot just compile the new version of code in the database. Instead, we need to provide transitional code that will change the state of the code.

In a state based development system, we use some sort of comparison method to examine the current state of a database (or code in a version control system) and compare that to another version of the database (or version control), and generate the SQL code required to migrate objects from one version to the next.

The typical flow for this is to make changes in development and then perform a comparison with the version of the database in a downstream environment, such as production, and then generate a script to transform the downstream environment to look like development. In essence, we generate a script to “deploy” our code. Let’s look at an example.

Demo:

Let’s show a table, let’s make changes, let’s compare.

In the demonstration, I used the Schema Compare from Visual Studio, but there are many ways to perform this comparison. A number of third party companies, including my employer, Redgate Software, make comparison tools that allow you to compare two versions of a database and produce a script to deploy the changes.

Advantages and Disadvantages of State

There are good things with using state. One is that developers can make changes, undo them, make new changes, and never worry about what work they do. When they get