



# Cheezburger BDSM

*Better Database Schema Manager*

By Stefan Rusek  
Director of Technology  
Cheezburger Inc.

# You're a kitten blog!

## Why do you need a Schema Manager?

- Over 99% of images and videos are user submitted
  - About 20 million submissions
  - Over 2.5 TB of images
  - Rating widgets and favorite buttons on every one
  - Highly rated assets are published to the front page
- 500 GB Microsoft SQL Database
- Over 50 sites
  - Including ICanHasCheezburger, FailBlog, Memebase, and many more
- Back-end management system

# What does it do?

- Safely upgrade database schemata
- Schema under source control
- Dynamically created database automated testing
- Flexible callback system
- Tables, columns, indexes, stored procedures, foreign keys, views, and table valued functions
- Generate schema from existing database

# What does it not do?

- Actions needing data migration
  - Delete schema
  - Modify Columns
- Expensive hard to get right operations
  - Modify Indexes

Solution: Use callbacks or manual changes

# Tables

```
<table name="User">
  <columns>
    <column name="Name" type="varchar" length="50" />
    <column name="PasswordSalt" type="varchar" length="128" />
    <column name="PasswordHash" type="varchar" length="128" />
    <column name="CreateDate" type="datetime" />
  </columns>
  <indexes>
    <index name="PK_User" type="PrimaryKey" columns="Name" />
    <index name="PK_Registration" type="Index"
      columns="CreateDate DESC, Name" />
  </indexes>
</table>
```

# Stored Procedures

```
<procedure name="InsertImage">
  <parameters>
    <parameter name="@Name" type="varchar" length="50" />
    <parameter name="@Data" type="image" />
  </parameters>
  <body> INSERT INTO [Image] VALUES (@Name, @Data) </body>
</procedure>
<procedure name="GetImage">
  <parameters>
    <parameter name="@Name" type="varchar" length="50" />
  </parameters>
  <body>
    SELECT [Data] FROM [Image] WHERE [Name] = @Name;
  </body>
</procedure>
```

# Includes and Schema Versioning

```
<?xml version="1.0" encoding="utf-8" ?>  
<schema xmlns="http://schemas.icanhascheezburger.com/db"  
version="2">  
  <includes>  
    <include>User.xml</include>  
    <include>Image.xml</include>  
    <include>AllTheSchema.xml</include>  
  </includes>  
</schema>
```

- Each schema file's version number is tracked separately
- Updater only tries to update schema files that are out of date
  - A ForceCheck (full check) can be done

# Callbacks

## Version 1

```
<table name="Image">
  <columns>
    <column name="fDeleted" type="bit" default="(0)" />
  </columns>
</table>
```

## Version 2

```
<table name="Image">
  <columns>
    <column name="IsDeleted" type="bit" default="(0)">
      <callback method="RenamedFDeletedTolsDeleted"
        type="MyApp.Namespace.ClassName"/>
    </column>
  </columns>
</table>
```

```
public void RenameFDeletedTolsDeleted(DbConnection conn, DbTransaction trans,
  CallbackAction action, int version)
{
  if (version > -1 && version < 2 && action == CallbackAction.After)
    Execute(conn, trans, "UPDATE [Image] SET [IsDeleted] = [fDeleted]");
}
```

# The Nuget package

Includes 3 additional items:

- Schema/DatabaseHelper.cs
  - Handy helper class that makes creating, upgrading, and deleting databases super easy.
  - Can be plugged into or adapted for use in whatever activation scenario you want.
- Schema/Schema.xml
  - Example schema file, DatabaseHelper uses it by default, so just start by editing this file.
- Schema/Schema.xsd
  - Provides information to Visual Studio needed for schema file Intellisense.

# Let's see some code

- Simple Example
  - Add BDSM to an Empty Project
  - Write some schema
  - Show update log
  - Use database to do something useful.

Questions and Further examples based on what audience wants to see demonstrated.

# Ideas for further development

- Currently limited to MSSQL
- Entity Framework Integration
  - EF Code-First attributes -> schema.
  - Schema -> EF Code-First POCO classes
- A fluent schema definer
- Callbacks
  - Attribute callbacks with version numbers
  - SQL or Stored Procedure based callbaks

# Cheezburger BDSM

Install via Nuget:

```
PM> Install-Package CheezBDSM
```

Source code:

<https://github.com/cheezburger/Cheezburger-BDSM>

Want to work for Cheezburger?

<http://jobs.cheezburger.com/>