SQL Data Sync – Installation Guide

# Overview

## System

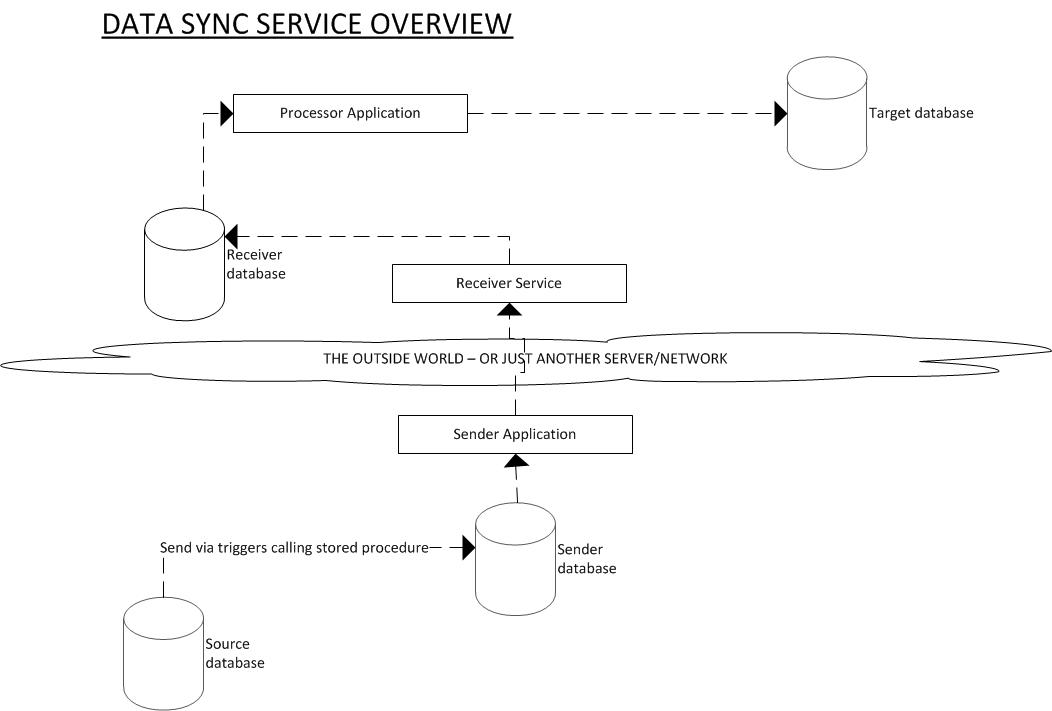
The SQL data sync solution is designed to keep a source and one or more target databases in sync when technologies such as SQL Service Broker are not an option. Data is sent over HTTP (port 80 or 443) and SQL is used to manage the messaging process, so no other components aside from SQL Server and HTTP are required.

Three components make up the solution

1. Sender (console app) – takes messages from the sender queue and pushes via HTTP to the receiver service
2. Receiver (web service) – receives messages from the sender and stores from processing
3. Processor (console app) – processes messages from the receiver queue and sends to target database

In addition to this, a database exists on the sender and receiver to manage queues. Triggers are implemented on source databases to track changes and put them in the sender queue.

The system architecture as shown below:



## System Requirements

Microsoft SQL Server 2008 or higher

Windows Server 2008R2 running IIS7

Microsoft .NET framework version 4.5

Microsoft MVC 4

## Assumptions

A good working knowledge of .NET web applications, IIS and the ability to perform basic database administration is assumed.

# Installation

## Package

The release package will contain folders for the three components needed. Transformations for the web.config file has not been carried out as part of the build and will need to be done on deployment using instructions in this document.

## Database

### Installation

Two databases are needed

- a send database that needs to reside on the same network and be able to “see” all source databases. This database is created using the script DataSync\SQL Scripts\SenderDB.sql

- a receive database that needs to reside on the same network and be able to “see” all target databases. This database is created using the script DataSync\SQL Scripts\ReceiverDB.sql

Triggers are needed to be run against any source database. These triggers are generated by the script DataSync\SQL Scripts\SyncDBTrigger.sql. Before running, do a find/replace for the following terms:

REPLACEME\_SENDERDATABASE with the name of the fully qualified name of the sender database

REPLACEME\_SYSTEMNAME with the name to be used for the system being synchronised (e.g. Calendar)

### Configuration

To set the system up, data needs to be put into the following tables:

#### SENDER DB

Insert into the SyncSystems table a row with the SystemName being that of the system being synched. Note the SystemId value of the created row.

Insert into the SyncSystemEndpoints table a row for each endpoint you want to send changes to the synched system to. Multiple endpoints can exist on the same receiver. The endpoint URL is the base url of the receiver service that will be installed below. No trailing / is necessary.

#### RECEIVER DB

Insert into the SyncSystemEndpoint table a row for each virtual endpoint – i.e. one for every database being written to for a sync message. The SystemName field must exactly match that used in the Sender Db and the EndpointName must match the Description field of the endpoint in the sender db. The DbConStr field should store the connection string for the database being written to in the form below:

data source=zzzz;initial catalog=yyyy;user id=xxxx;password=wwww

Replace zzzz with the server address, yyyy with the database name, xxxx with the username and wwww with the password to use.

## Receiver Service

The DataSync\Receiver folder should be deployed to a suitable location on the server.

### Web.config updates

The web.config file, found in the root of the deployed folder will need changing depending on the environment being deployed to.

#### root/services web.config

|  |  |
| --- | --- |
| Configuration/appSettings key | Value to be set to |
| dbconstr | The database string of the receiver database. Replace zzzz with the server address, yyyy with the database name, xxxx with the username and wwww with the password to use. |
| statstoken | The token that must be sent to the service to receive stats from it. |
| accepttoken | The token used to authenticate the sender with the receiver. The sender system must have the same token or messages will not send. |

Finally, configure the log4net\file node appropriately to save error logs.

### IIS Configuration

Once the installation steps have been completed, the final task is to set up IIS to host the site.

The first step is to create an app pool, using .net 4 and integrated pipeline mode.

Next, a new site should be created in IIS. This can either be as an application within the default web site, or as a new site in the IIS root. The physical path for the site should be that of the root folder of the site and the app pool set to the one created above. Bindings also be set as necessary.

Anonymous access must be enabled under the authentication tab.

It is recommended the receiver service be configured to only accept requests from the Sender servers IP address/range.

## Processor Application

The DataSync\Processor folder should be deployed to a suitable location on the server.

### Web.config updates

The web.config file, found in the root of the deployed folder will need changing depending on the environment being deployed to.

#### root/services web.config

|  |  |
| --- | --- |
| Configuration/appSettings key | Value to be set to |
| dbconstr | The database string of the receiver database. Replace zzzz with the server address, yyyy with the database name, xxxx with the username and wwww with the password to use. |
| batchsize | The number of messages to send in a process run, higher numbers send more data, but require a longer processing cycle |

Finally, configure the log4net\file node appropriately to save error logs.

### Configuration

Set up a scheduled task – to run when not logged in with a system user account and link it to the executable file in the directory the sync app has been installed to. A recommended run interval for this app is 5 minutes.

## Sender Application

The DataSync\Sender folder should be deployed to a suitable location on the server.

### Web.config updates

The web.config file, found in the root of the deployed folder will need changing depending on the environment being deployed to.

#### root/services web.config

|  |  |
| --- | --- |
| Configuration/appSettings key | Value to be set to |
| dbconstr | The database string of the sender database. Replace zzzz with the server address, yyyy with the database name, xxxx with the username and wwww with the password to use. |
| batchsize | The number of messages to send in a process run, higher numbers send more data, but require a longer processing cycle |
| sendtoken | The token used to authenticate the sender with the receiver. The receiver system must have the same token or messages will not send. |
| proxy.username | If a proxy server is needed, the username of the account to use to authenticate |
| proxy.password | If a proxy server is needed, the password of the account to use to authenticate |

If a proxy server is being used, configure settings in the system.net/defaultProxy node of the config file as appropriate. If there is no proxy server needed, delete the defaultProxy node and remove the two setting keys

Finally, configure the log4net\file node appropriately to save error logs.

### Configuration

Set up a scheduled task – to run when not logged in with a system user account and link it to the executable file in the directory the sync app has been installed to. A recommended run interval for this app is 5 minutes.