Load Balancing - Setup Guide

2017-02-10

C1 CMS

Contents

1   Introduction

The Load Balancing add-on allow C1 CMS to run in an environment where the public-facing website can be clustered and run independently of other nodes in the cluster, allowing for features such as load balancing and fail over being employed.

It propagates content and media file changes across nodes connected to the same database used by the Load Balancing add-on.
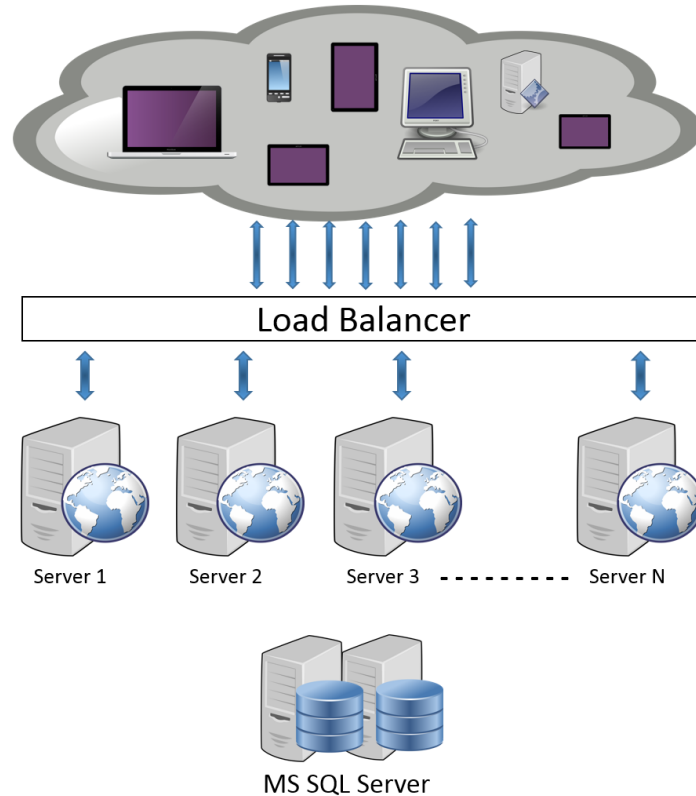

Figure 1: Network load balancing

Content editors can add, edit and delete content like pages and media files on the website - and these changes will be reflected across all nodes in a C1 CMS installation.

With the Load Balancing add-on you can make use of these additional features compared to a standard C1 CMS installation:

- C1 CMS front-end can now run in a clustered environment
- Content and media file changes are immediately updated on all nodes

1.1   Who should read this guide

The guide is intended for a technical person who knows the network load balancing technologies and capable of setting one up for a website.

We expect that this person has an SQL-based C1 CMS website running or is able to migrate an XML-based website to an SQL-based data store.

This person should have access to the System perspective in the CMS Console to install add-ons, migrate the website and make backups if needed, and check the logs. Please note that he or she should also have access to an SQL database to change settings on it.

2   Getting started

Along with the installation of the Load Balancing add-on, you will take a few pre- and post-installation setup steps.

1. Pre-installation setup:
   a. Use Composite C1 (now "C1 CMS") version 4.1 or later.
   b. Install the SQL Server Data Provider add-on.
   c. Migrate the website data store to an SQL database.
   d. Enable the Service Broker option on the database.
2. The add-on installation:
   a. Install the add-on
   b. Check the log
3. Post-Installation setup:
   a. Deploy the website on multiple servers.
   b. If necessary, set up one dedicated website instance for the CMS Console administration.

2.1   Prerequisites

- The Load Balancing add-on requires Composite C1 (now C1 CMS) 4.1 (or later) running with the SQL Data Provider.
- Both the IIS Server and Microsoft SQL Server must run on Windows Server 2008 or later.
- Microsoft SQL Server 2005 or later is required.
- Microsoft SQL Server must have the "Service Broker" feature enabled.
- Nodes (websites) are not required to be part of a NLB setup (some nodes may be, some may not).
- Any Network Load Balancing technology may be used.

2.2   Limitations

The Load Balancing add-on has the following limitations:

- Administrative sessions may not be load balanced, but need to run on a single server. You can work around this limitation in one of the following ways:
  o Run a dedicated C1 CMS website which is not part of the cluster and do content editing on this website
  o Have the load balancing mechanism use "sticky sessions" for administrative users. For Microsoft Network Load Balancing Services this can be achieved by setting affinity to "Single" or "Network".
- It will not migrate changes related to website development.
  Development and staging environments should be managed and migrated using other means – the Load Balancing add-on will not automate migration in this area.
  The example of changes not supported by the Load Balancing add-on are:
  o Data type changes
  o Changes made to files on the file system (for example, CSS, code files, templates)
  o C1 CMS add-ons installations

C1 CMS

3 Pre-installation setup

To properly work, the Load Balancing add-on requires that you take a few setup steps before installing its installation.

Please note that the add-on requires Composite C1 (now "C1 CMS") version 4.1 or later.

(Please also see "Prerequisites".)

The Load Balancing add-on uses the Service Broker component of Microsoft SQL Server database engine. That's why the C1 CMS website must use an SQL database as its data store and the Service Broker option must be enabled on this database.

If your website is an XML-based (uses XML files for its data store), you need to install the SQL Server Data Provider and migrate the website to an SQL-based data store.

3.1 Installing SQL Server Data Provider

For the installation steps, please see "SQL Server Data Provider: Installation Guide".

3.2 Migrating the website to an SQL-based data store

For the migration steps, please see "SQL Server Data Provider: User Manual".

3.3 Enabling the Service Broker on the database

To enable the Service Broker option on the database used as a data store for the C1 CMS website:

1. Stop the website on the web server.
2. Run Microsoft SQL Server Management Studio.
3. Connect to the server where the C1 CMS database is hosted.
4. Open the database's properties (right-click the database in the tree and click "Properties")
5. On the "Options" tab, scroll down to the "Service Broker" section.
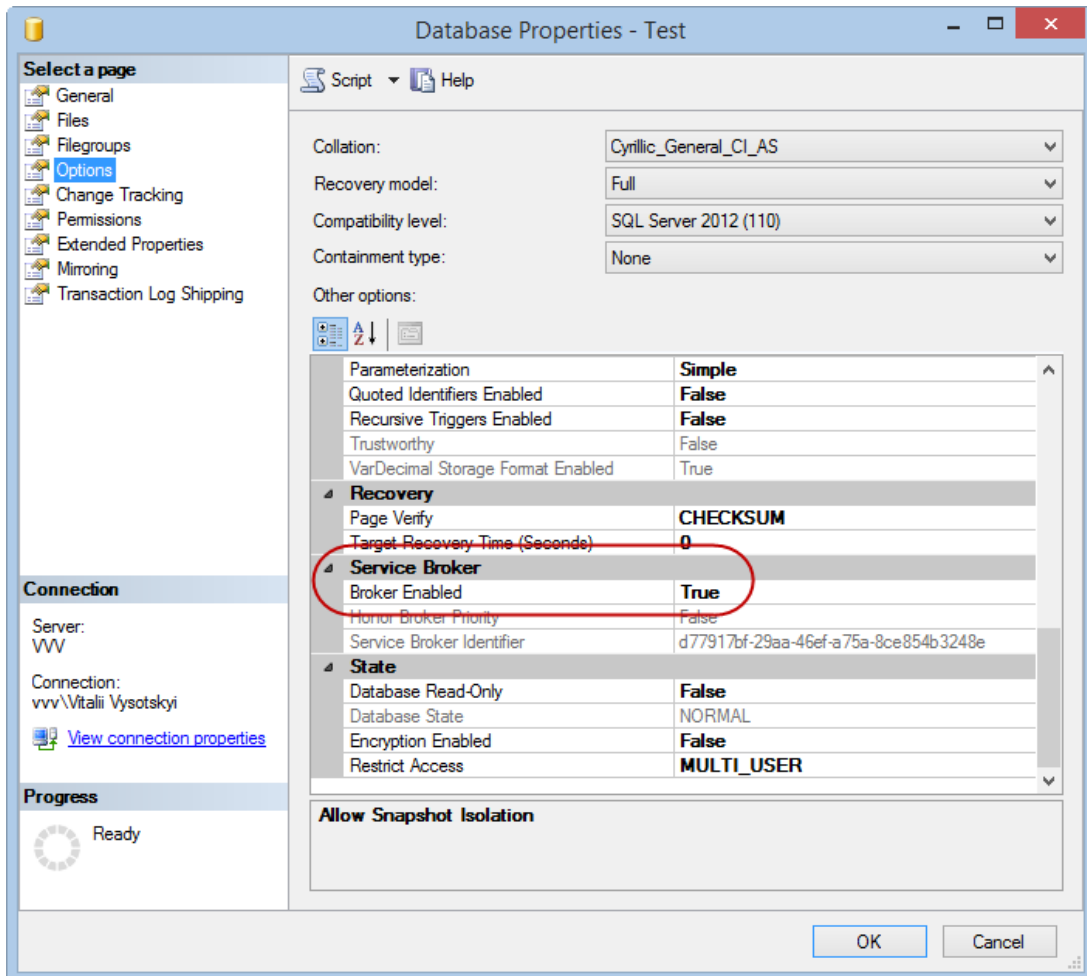6. Select "True" for the "Broker Enabled" option.

C1 CMS

Figure 2: Enabling Server Broker

7. Click "OK" in the "Database Properties" window.
8. Start the website on the web server.

You can use the Load Balancing Add-on with Microsoft Network Load Balancing Services, but other load balancing technologies should be usable as well.

When using Microsoft NLB you can use any affinity on the front-end, provided your web application does not introduce machine local session state. The CMS Console cannot run in a load balanced setup with affinity set to "None", but will work with affinity set to "Single" and "Network".
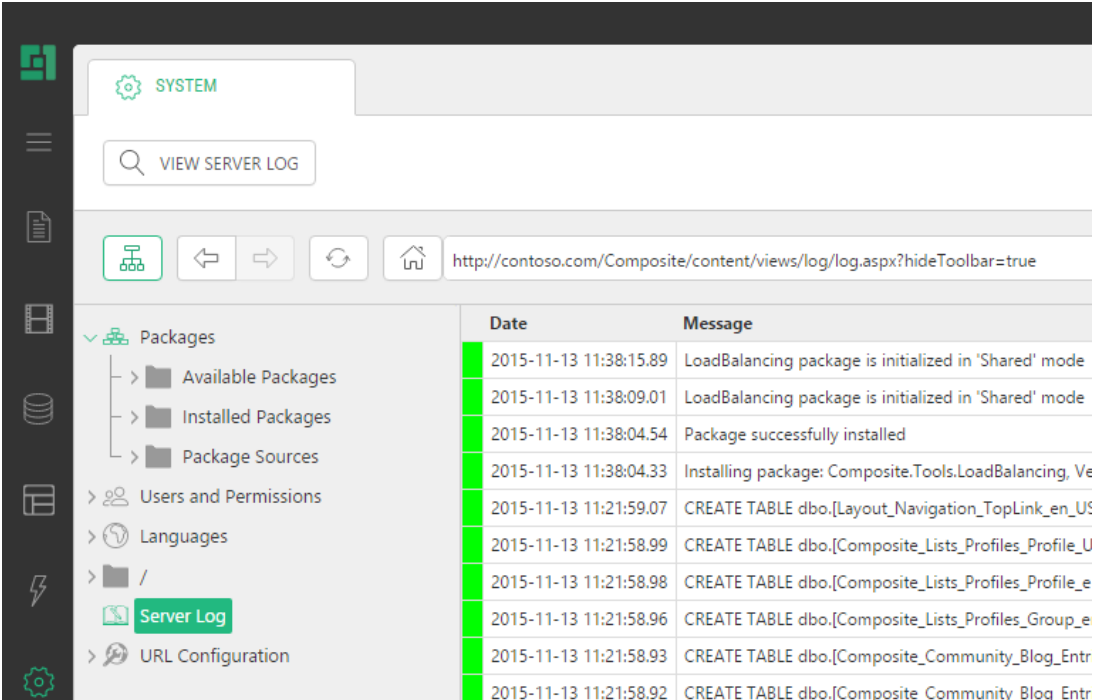
4   Installation

Now that you've completed the pre-installation setup, go on to install the add-on and make sure it is working properly.

To install the Load balancing add-on:

1. Log in to the CMS Console
2. In the **System** perspective, expand **Packages** / **Available Packages** / **Composite.Tools**.
3. Select **Composite.Tools.LoadBalancing** and click **Package Info** on the toolbar.
4. In the **Package Info** view that opens, click **Install**.
5. Complete the wizard.

After installing the add-on, check the server log (**System** / **Server Log** / **View Server Log**) to see if the add-on is running. You should see messages like "Package successfully installed" and "LoadBalancing package is initialized in 'Shared' mode".

Figure 3: Load Balancing started successfully

5 Post-installation setup and use

Once you've installed the Load Balancing add-on, go on to deploy copies of the website on other servers.

These copies (nodes) are not required to be part of a NLB setup. You can also may have a mixed setup where some nodes are part of the NLB and some are not.

You may want to consider having a dedicated instance of the website for the CMS Console administration.

5.1 Preparing another node in your cluster

1. Set up a new server and join it to your load balanced cluster.
2. Copy your C1 CMS website to the new machine.
3. Set up a new IIS site in accordance to the C1 CMS version you are running

Please see also "Configuration".

5.2 Deploying the Website on Multiple Servers

Once you've finished the setup, make a copy of the website to deploy it on other servers. You can use the XML-Based Site Backup add-on to make a zipped copy of the website. (Please note that the website files will be zipped only – the SQL database in use will not be backed up.)

Then go on and deploy it on as many servers as you need.

5.3 Making a dedicated instance for the CMS Console administration

The Load Balancing add-on does not synchronize all the website files (please see "Limitations".

We thus recommend that you have set up one dedicated instance for the CMS Console administration.

To disable usage of the CMS Console in production sites:

1. Completely remove the "/Composite" folder from the website.
2. **For Composite C1 (now "C1 CMS") version 4.2 (Update 1) or earlier only**, also remove references to AdministrativeAuthorizationHttpModule from web.config.

5.4 Using Load Balancing

Users maintaining content should not take any further steps to ensure content change synchronization after the add-on has been installed and the cluster set up.

The Load Balancing add-on will synchronize all changes to content and media files automatically.

Developers that make changes to website files such as CSS documents or Razor functions are responsible for synchronizing these changes across servers on their own.

C1 CMS

Load Balancing - Setup Guide

6   Configuration

By default the Load Balancing add-on will run in "shared" mode, which enables any node in the cluster to accept changes and alert other nodes about changed data. This works right out of the box and requires no maintenance or additional configuration.

You can change the mode to "server" or "client" and specify a connection string for the Load Balancing tables other than used by the C1 CMS website by default.

If you edit ~/web.config you will find the following settings in the <appSettings />:

```
<appSettings>
  <add key="Composite.LoadBalancing.Mode" value="shared" />
  <add key="Composite.LoadBalancing.ConnectionStringName" value="c1" />
</appSettings>
```

Listing 1: Load Balancing settings in web.config

Valid values for Composite.LoadBalancing.Mode are listed below.

- "server"
  The Load Balancing add-on will record and push changes made on the current node. It will not listen for changes made by other nodes.
- "client"
  The add-on will listen for changes pushed by nodes running in "server" mode and will invalidate affected caches when notified.
- "shared"
  This is the default setting. The combined mode where the node will both record and push changes and will listen for the changes pushed by other nodes.

C1 CMS