

****

DB Failover

Version: v1.2

Last Updated: 27-May-2021

**Document Control**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Document Name** | DB Failover | | | | |
| **Author** | Haranath | | | | |
| **Distribution** | Tecknodreams Software consulting Pvt Ltd. | | |  | |
| **Version History** | **Version** | **Date** | **Description** | | **Update By** |
| 1.0 |  | Initial Draft | | Haranath |
|  | 1.2 | 13/10/2021 | MySQL 8 support added | | Haranath |

**Copyright**

Copyright © 2021 [Tecknodreams Software Consulting Pvt. Ltd.](http://www.tecknodreams.com) All Rights Reserved.

**Restricted Rights Legend**

This documentation is subject to and made available only pursuant to the terms of the Tecknodreams License Agreement and may be used or copied only in accordance with the terms of that agreement. It is against the law to copy the documentation except as specifically allowed in the agreement. This document may not, in whole or in part, be copied photocopied, reproduced, translated, or reduced to any electronic medium or machine readable form without prior consent, in writing, from Tecknodreams.

Information in this document is subject to change without notice and does not represent a commitment on the part of Tecknodreams.

# Introduction

Document will explain how to enable DB failover in sapphire application. DB failover can be enabled only Web component (Wildfly server).

To enable DB failover below steps involved

1. [MySQL replication enabling](#_MYSQL_Replication_Enabling)
2. [Standalone changes to enable DB failover](#_Standalone_XML_File)

# MYSQL Replication Enabling

Follow the attached document steps to enable MySQL replication



# Standalone XML File Changes

1. Stop sapphire service
2. Extract zip file attached and copy mysql-connector-java-5.1.46.jar and module.xml file to <SapphireInstalled path>\ WebManagement\modules\system\layers\base\com\mysql\main



Note : if MySQL 8 then above step not required

1. Open standalone.xml file in text editor and change the Datasource connection url as below

MySQL 5.7

jdbc:mysql://[primary host][:port],[secondary host 1][:port][,[secondary host 2][:port]]...[/[database]]?zeroDateTimeBehavior=convertToNull&amp;useOldAliasMetadataBehavior=true&amp;autoReconnect=true&amp;failOverReadOnly=false&amp;secondsBeforeRetryMaster=0&amp;queriesBeforeRetryMaster=0&amp;useSSL=false&amp;loadBalanceSQLStateFailover=00,12345

ex: jdbc:mysql://10.3.0.68:3306,10.3.0.69:3306/support?zeroDateTimeBehavior=convertToNull&amp;useOldAliasMetadataBehavior=true&amp;autoReconnect=true&amp;failOverReadOnly=false&amp;secondsBeforeRetryMaster=0&amp;queriesBeforeRetryMaster=0&amp;useSSL=false&amp;loadBalanceSQLStateFailover=00,12345

MySQL 8

jdbc:mysql://[primary host][:port],[secondary host 1][:port][,[secondary host 2][:port]]...[/[database]]?zeroDateTimeBehavior= CONVERT\_TO\_NULL&amp;useOldAliasMetadataBehavior=true&amp;autoReconnect=true&amp;failOverReadOnly=false&amp;secondsBeforeRetryMaster=0&amp;queriesBeforeRetryMaster=0&amp;useSSL=false&amp;loadBalanceSQLStateFailover=00,12345

ex: jdbc:mysql://10.3.0.68:3306,10.3.0.69:3306/support?zeroDateTimeBehavior= CONVERT\_TO\_NULL&amp;useOldAliasMetadataBehavior=true&amp;autoReconnect=true&amp;failOverReadOnly=false&amp;secondsBeforeRetryMaster=0&amp;queriesBeforeRetryMaster=0&amp;useSSL=false&amp;loadBalanceSQLStateFailover=00,12345

1. Start the sapphire service

MySQL JDBC Parameters

1. failOverReadOnly

failOverReadOnly property need to set false by default this value is true, if this value is true then if the driver fails to establish connection with primary host, it automatically switches to another host with read only mode means the connection will not do any write operations and it will do only read operations

1. secondsBeforeRetryMaster

determines how much time need to wait before trying to fall back to primary host

1. queriesBeforeRetryMaster

determines the number of queries that are executed before the driver tries to fall back to the primary host.

In general, an attempt to fall back to the primary host is made when at least one of the conditions specified by the two properties is met, and the attempt always takes place at transaction boundaries. However, if auto-commit is turned off, the check happens only when the method Connection.commit() or Connection.rollback() is called. The automatic fallback to the primary host can be turned off by setting simultaneously secondsBeforeRetryMaster and queriesBeforeRetryMaster to “0”. Setting only one of the properties to “0” only disables one part of the check.

1. loadBalanceSQLStateFailover

whenever SQL exception occurred driver failover triggered, this will cause the problems within transaction, in that case MYSQL lock will happen because without transaction commit and connection is connected to another host and new transaction started. It will cause to lock transaction (hang state) to avoid this we can specify the SQLState code prefixes to trigger failover