



**EXPRESSCLUSTER X for Linux**

**Quick Start Guide for Docker**

**(MySQL)**

**Version 2**  
**Feb 15, 2017**

---

---

## **Disclaimer**

The contents of this document are subject to change without notice. NEC Corporation assumes no responsibility for technical or editorial mistakes in or omissions from this document. To obtain the benefits of the product, it is the customer's responsibility to install and use the product in accordance with this document. The copyright for the contents of this document belongs to NEC Corporation. Copying, altering, or translating this document, in full or in part, without the permission of NEC Corporation, is prohibited.

## **Trademark Information**

Information in this document is subject to change without notice. No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of NEC Corporation.

Linux is a registered trademark or trademark of Linus Torvalds in the United States and other countries.

Other product names and slogans written in this manual are trademarks or registered trademarks of their respective companies.

---

---

## **Contents**

<b>1</b>	<b>About This Guide</b> .....	<b>4</b>
1.1	Using This Guide .....	4
1.2	Revision History .....	4
1.3	Evaluation Environment.....	4
1.4	For More Information.....	4
<b>2</b>	<b>Overview</b> .....	<b>6</b>
<b>3</b>	<b>System Requirements and Planning</b> .....	<b>7</b>
3.1	System Requirements .....	7
3.2	System Planning.....	7
<b>4</b>	<b>EXPRESSCLUSTER X Installation</b> .....	<b>9</b>
<b>5</b>	<b>Base Cluster Setup</b> .....	<b>9</b>
<b>6</b>	<b>Docker Installation</b> .....	<b>10</b>
6.1	Install Docker (Primary Server) .....	10
6.2	Install Docker (Secondary Server) .....	10
<b>7</b>	<b>MySQL Cluster Setup</b> .....	<b>12</b>
7.1	Create MySQL Container (Primary Server).....	12
7.2	Create MySQL Container (Secondary Server).....	14
7.3	Create the Script to Control MySQL Container .....	16
7.4	Setup MySQL Monitor Resource .....	16
<b>8</b>	<b>Verify Functionality</b> .....	<b>18</b>
8.1	Move the Failover Group.....	18
8.2	Failover on Server Shutdown .....	18
<b>9</b>	<b>Appendix</b> .....	<b>19</b>
9.1	System Planning Worksheet.....	19

---

# 1 About This Guide

## 1.1 Using This Guide

This guide provides a hands-on “Quick Start” set of instructions to create your application cluster with Docker and EXPRESSCLUSTER X for Linux. The guide assumes users have Linux system administration knowledge and skills with experience in installation and configuration of Linux operating systems, networks, and Docker.

## 1.2 Revision History

Version	Date	Description
1	Oct 21, 2016	Initial version
2	Feb 15, 2017	Fixed typo and revised tables.

## 1.3 Evaluation Environment

This clustering method has been evaluated with the following OS and software.

- CentOS Linux release 7.2.1511
- MySQL Community Server 5.7.15
- EXPRESSCLUSTER X 3.3 for Linux (expresscls-3.3.3-1.x86\_64)

## 1.4 For More Information

We have the following guides for instant support.

- **Getting Started Guide** – This guide explains general cluster concepts and overview of EXPRESSCLUSTER functionality.
- **Installation and Configuration Guide** – This guide explains EXPRESSCLUSTER X installation and configuration procedures in detail.
- **Reference Guide** – This is a reference of commands that can be put in EXPRESSCLUSTER X scripts and maintenance commands that can

---

be executed from the server command prompt.

The above stated guides can also be found at

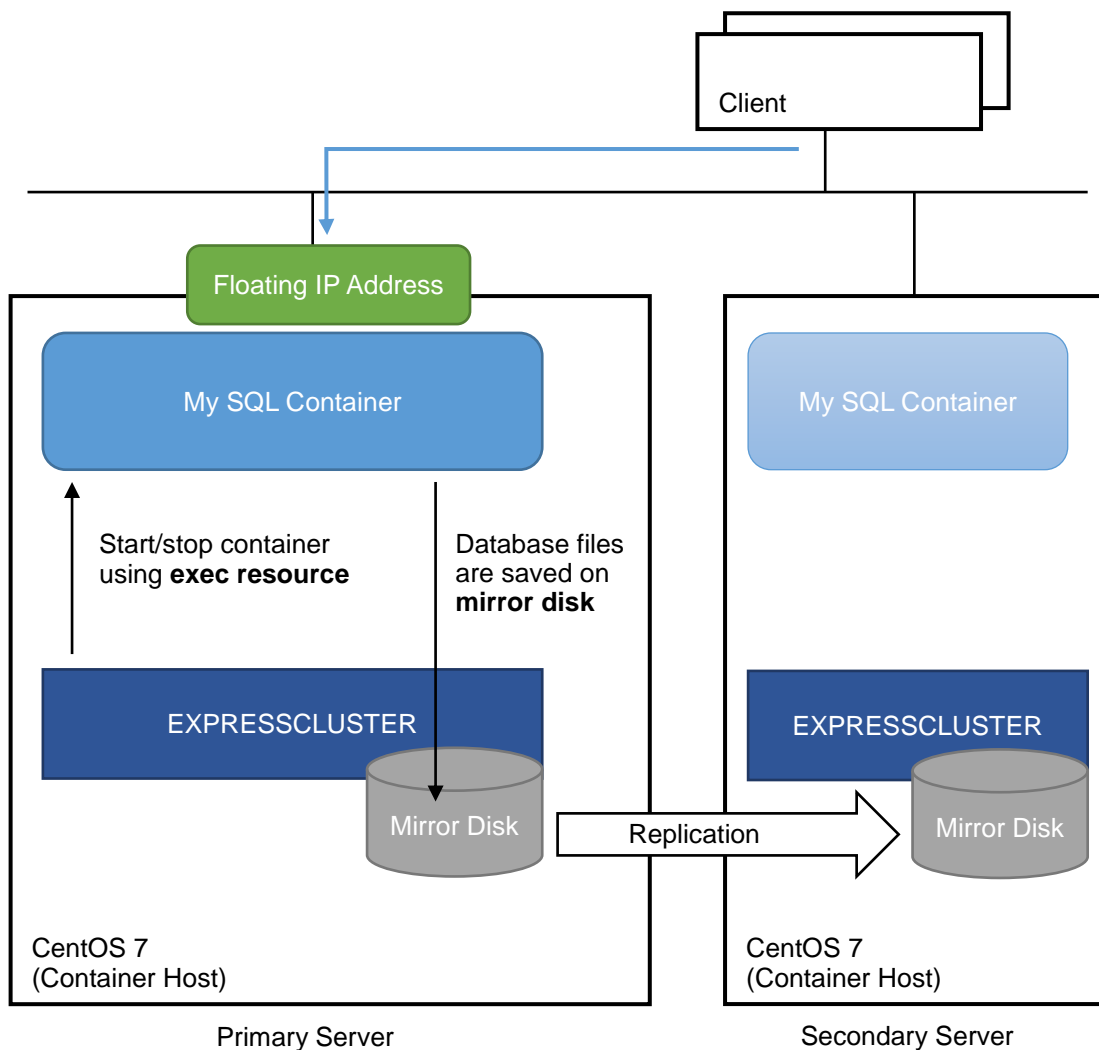
<http://www.nec.com/en/global/prod/expresscluster/en/support/manuals.html>

.

---

## 2 Overview

- The container OS image and the container are saved on the system drive of the primary and the secondary servers.
- EXPRESSCLUSTER is installed on the container host.
- MySQL is installed on the container.
- Database files are saved on a partition controlled by **mirror disk resource**.
- MySQL container is controlled by **exec resource**.
- Client machines use **floating IP address resource** to access to MySQL database.



---

## 3 System Requirements and Planning

### 3.1 System Requirements

- Docker  
Refer to Docker website.  
<https://docs.docker.com/engine/installation/>
- EXPRESSCLUSTER  
Refer to EXPRESSCLUSTER X *Getting Started Guide*.

### 3.2 System Planning

Here are sample parameters for a cluster. In section 9.1, there is **System Planning Worksheet**. Fill out all tables of **System Planning Worksheet** to create a cluster.

Machine #1: Primary Server

Machine #2: Secondary Server

Machine #3: Test Client Machine

**Table 1: Network Configuration**

Machine	Hostname/ Container Name	IP Address	Mirror Disk Connect
#1	server1	192.168.0.11/24	mdc1
		192.168.1.11/24	Do Not Use
#2	server2	192.168.0.12/24	mdc1
		192.168.1.12/24	Do Not Use
#3	client1	192.168.1.99/24	N/A

**Floating IP Address** : 192.168.1.21

---

**Table 2: OS and Disk Configuration**

Machine	OS	Mirror Disk
#1	CentOS 7.2.1511	<b>Cluster Partition:</b> Device: /dev/sdb1 Size: 17 MB
#2	CentOS 7.2.1511	<b>Data Partition:</b> Device: /dev/sdb2 Size: 10 GB
#3	CentOS 7.2.1511	N/A

**Table 3: Logins and Passwords**

Machine	Login	Password
#1	root	passw0rd
#2	root	passw0rd
#3	root	passw0rd



---

## 4 EXPRESSCLUSTER X Installation

Install EXPRESSCLUSTER on both the primary and the secondary server with reference to *Installation and Configuration Guide*.

## 5 Base Cluster Setup

Add the following resources with reference to *Chapter 5 Creating the cluster configuration data* in *Installation and Configuration Guide*.

- Exec resource
- Floating IP address resource
- Mirror disk resource

---

## 6 Docker Installation

### 6.1 Install Docker (Primary Server)

1. Open a terminal window with root account.
2. Run the following command to install Docker.

```
# yum -y install docker
```

3. If a proxy server is required to access to the Internet, run the following commands to modify docker.service file.

```
# cd /etc/systemd/system
# cp /usr/lib/systemd/system/docker.service .
# vi docker.service
```

Add the following lines to docker.service file.

```
Environment="HTTP_PROXY=<Your Proxy Server:Port>"
Environment="HTTPS_PROXY=<Your Proxy Server:Port>"
```

4. Run the following command to enable auto startup for docekr daemon.

```
# systemctl enable docker
```

5. Run the following command to start docker daemon.

```
# systemctl start docker
```

6. Run the following command to install MySQL container image.

```
# docker pull mysql:latest
```

7. Run the following command to check MySQL container images is installed.

```
# docker images
REPOSITORY          TAG          IMAGE ID      ...
docker.io/mysql     latest      18f13d72f7f0 ...
```

### 6.2 Install Docker (Secondary Server)

Perform the steps 1 to 7 in section 6.1 on the secondary server.



---

## 7 MySQL Cluster Setup

### 7.1 Create MySQL Container (Primary Server)

1. Check if the failover group is running on the primary server.
2. Create a directory (ex. mydata) on the mount point of the mirror disk (ex. /mnt/mysql) to share the directory with the container.

```
# mkdir /mnt/mysql/mydata
```

3. Create MySQL container.

```
# docker run -d -it --name mysql01 -v /mnt/mysql/mydata:/mydata -p 3306:3306 mysql:latest bash
```

4. Login to the container.

```
# docker attach mysql
```

5. Create the original file and copy mysqld.cnf file to /mydata directory.

```
# cd /etc/mysql/mysql.conf.d
# cp mysqld.cnf mysqld.cnf.org
# cp mysqld.cnf /mydata
```

6. Press Ctrl + P and Ctrl + Q to logout from the container.
7. Open mysqld.cnf with a text editor (ex. vi).

```
# vi /mydata/mysqld.cnf
```

8. Change datadir from /var/lib/mysql to /mydata.

```
[mysqld]
:
datadir = /mydata
```

9. Login the container, again.

```
# docker attach mysql
```

10. Replace the original mysqld.cnf with the modified one.

```
# mv /mydata/mysqld.cnf /etc/mysql/mysql.conf.d/
```

11. Initialize a database. Temporary password will be appeared in standard output. Note down the password. It is required to login to MySQL first.

---

```
# mysqld --initialize --user=mysql
```

12. Start MySQL server.

```
# service mysql start
```

13. Login to MySQL with the temporary password generated at database initialization.

```
# mysql -u root -p
Enter password:
```

14. Change the password.

```
mysql> ALTER USER 'root'@'localhost' IDENTIFIED BY 'new_password' ;
```

15. Create a database (ex. testdb).

```
mysql> CREATE DATABASE testdb;
```

16. Allow any remote clients to connect to the database.

```
mysql> GRANT ALL PRIVILEGES ON testdb.* TO root@"%" IDENTIFIED BY
'your password' WITH GRANT OPTION;
```

17. Logout from MySQL.

```
mysql> exit
Bye
```

18. Stop MySQL server.

```
# service mysql stop
```

19. Logout and stop the container.

```
# exit
```

20. Check if the container is not running.

```
# docker ps -a
CONTAINER ID   ...    STATUS                ... NAMES
8a61230eaced   ...    Exited (0) 30 seconds ago    mysql01
```

21. Move the failover group to the secondary server.

```
# clpgrp -m <failover group name>
```

---

## 7.2 Create MySQL Container (Secondary Server)

1. Create MySQL container.

```
# docker run -d -it --name mysql01 -v /mnt/mysql/mydata:/mydata -p 3306:3306 mysql:latest bash
```

2. Login to the container.

```
# docker attach mysql
```

3. Create the original file and copy mysqld.cnf file to /mydata directory.

```
# cd /etc/mysql/mysql.conf.d
# cp mysqld.cnf mysqld.cnf.org
# cp mysqld.cnf /mydata
```

4. Press Ctrl + P and Ctrl + Q to logout from the container.
5. Open mysqld.cnf with a text editor (ex. vi).

```
# vi /mydata/mysqld.cnf
```

6. Change datadir from /var/lib/mysql to /mydata.

```
[mysqld]
:
datadir = /mydata
```

7. Login the container, again.

```
# docker attach mysql
```

8. Replace the original mysqld.cnf with the modified one.

```
# mv /mydata/mysqld.cnf /etc/mysql/mysql.conf.d/
```

9. Start MySQL server.

```
# service mysql start
```

10. Login to MySQL with the new password.

```
# mysql -u root -p
Enter password:
```

11. Check if the same databases are available.

---

```
mysql> show databases;
+-----+
| Database          |
+-----+
| information_schema|
| mysql             |
| :                 |
| testdb           |
+-----+
6 rows in set (0.10 sec)
```

12. Logout from MySQL.

```
mysql> exit
Bye
```

13. Stop MySQL server.

```
# service mysql stop
```

14. Logout and stop the container.

```
# exit
```

15. Check if the container is not running.

```
# docker ps -a
CONTAINER ID   ...    STATUS                               ...    NAMES
8a3bbeacbe35   ...    Exited (0) 30 seconds ago           ...    mysql01
```

16. Stop exec resource using WebManager or clprsc command.

## 7.3 Create the Script to Control MySQL Container

1. Save the samples script files on some director that WebManager can access.
2. Launch WebManager and check if the exec resource is not running.
3. Change to [Config Mode].
4. Click the failover group on the left pane. Right click the exec resource and click [Properties].
5. Click [Details] tab.
6. Click [start.sh] and click [Replace] button. Search the sample start.sh

---

file and click [Open]. Click [Yes].

7. Click [stop.sh] and click [Replace] button. Search the sample stop.sh file and click [Open]. Click [Yes].
8. Click [OK].
9. Click [File] on menu bar. Click [Apply the Configuration File].
10. Back to [Operation Mode] and start the exec resource.
11. Check if the container and MySQL is running.

```
# docker ps
CONTAINER ID   ...    STATUS           ...    NAMES
8a61230eaced   ...    Up 30 seconds    ...    mysql01

# docker exec mysql01 service mysql status
MySQL Community Server 5.7.15 is running.
```

## 7.4 Setup MySQL Monitor Resource

1. Install MySQL library file on both the primary and the secondary server.

```
# yum install mysql-libs
```

2. Register EXPRESSCLUSTER X Database Agent license on the both the primary and the secondary server with reference to ***Installation and Configuration Guide***.
3. Launch WebManager and change to [Config Mode].
4. Right click [Monitors] and click [Add Monitor Resource].
5. Click [Get License Info]. Select [MySQL monitor] from [Type] list box. Click [Next].
6. Click [Browse] button to setup [Target Resource]. Click exec resource to control the container. Click [OK] and [Next].
7. Set each parameters as below. Click [Next].



IP Address : Floating IP Address  
 User Name : Database login user (ex. root)  
 Password : Database login password  
 Library Path : /usr/lib/mysql/libmysqlclient.so.15

8. Click [Browse] button to setup [Recovery Target]. Click failover group. Click [OK] and [Finish].
9. Click [File] on menu bar. Click [Apply the Configuration File]. Message box suspend the cluster click [OK]. Resume the cluster.

---

## 8 Verify Functionality

### 8.1 Move the Failover Group

1. Using WebManager, move the **%failover\_group%** from the primary server to the secondary server and confirm the following.
  - The client can access to the database.
  - The container on the primary server should be in a stopped state.
  - The container on the secondary server should be started.
2. Using WebManager, move the **%failover\_group%** back to the primary server and confirm the following.
  - The client can access to the database.
  - The container on the primary server should be started.
  - The container on the secondary server should be in a stopped state.

### 8.2 Failover on Server Shutdown

1. Restart the primary server and confirm the following.
  - The client can access to the database.
  - The container on the secondary server should be started.
2. After the primary server comes back to the cluster, restart the secondary server and confirm the following.
  - The client can access to the database.
  - The container on the primary server should be started.

---

## 9 Appendix

### 9.1 System Planning Worksheet

Machine #1: Primary Server

Machine #2: Secondary Server

Machine #3: Test Client Machine

**Table 1: Network Configuration**

Machine	Hostname	IP Address	Mirror Disk Connect
#1	server1		
#2	server2		
#3	client1		N/A

**Floating IP Address** : \_\_\_\_\_

**Table 2: OS and Disk Configuration**

Machine	OS	Mirror Disk
#1		<b>Cluster Partition:</b> Device: Size:
#2		<b>Data Partition:</b> Device: Size:
#3		N/A

**Table 3: Logins and Passwords**

Machine	Login	Password
#1		
#2		
#3		