

CLUSTERPRO[®] X *for Linux*

PPガイド
(MySQL)

2019.5.14
第01版

CLUSTERPRO

改版履歴

版数	改版日付	内容
01	2019/05/14	新規作成

© Copyright NEC Corporation 2019. All rights reserved.

免責事項

本書の内容は、予告なしに変更されることがあります。

日本電気株式会社は、本書の技術的もしくは編集上の間違い、欠落について、一切責任をおいませぬ。

また、お客様が期待される効果を得るために、本書に従った導入、使用および使用効果につきましては、お客様の責任とさせていただきます。

本書に記載されている内容の著作権は、日本電気株式会社に帰属します。本書の内容の一部または全部を日本電気株式会社の許諾なしに複製、改変、および翻訳することは禁止されています。

商標情報

CLUSTERPRO® X は日本電気株式会社の登録商標です。

Linuxは、Linus Torvalds氏の米国およびその他の国における登録商標あるいは商標です。

本書に記載されたその他の製品名および標語は、各社の商標または登録商標です。

その他のシステム名、社名、製品名等はそれぞれの会社の商標及び登録商標です。

目次

はじめに.....	i
対象読者と目的.....	i
適用範囲.....	i
CLUSTERPRO マニュアル体系.....	ii
本書の表記規則.....	iii
最新情報の入手先.....	iv
第 1 章 MySQL.....	1
機能概要.....	1
機能範囲.....	3
動作環境.....	4
インストール手順.....	5
動作確認手順.....	21

はじめに

対象読者と目的

『CLUSTERPRO® PPガイド』は、クラスタシステムに関して、システムを構築する管理者、およびユーザサポートを行うシステムエンジニア、保守員を対象にしています。

本書では、CLUSTERPRO環境下での動作確認が取れたソフトウェアをご紹介します。ここで紹介するソフトウェアや設定例は、あくまで参考情報としてご提供するものであり、各ソフトウェアの動作保証をするものではありません。

適用範囲

本書は、以下の製品を対象としています。

CLUSTERPRO X 3.3 for Linux

CLUSTERPRO X 4.0 for Linux

CLUSTERPRO マニュアル体系

CLUSTERPRO のマニュアルは、以下の 4 つに分類されます。各ガイドのタイトルと役割を以下に示します。

『CLUSTERPRO X スタートアップガイド』(Getting Started Guide)

CLUSTERPRO を使用するユーザを対象読者とし、製品概要、動作環境、アップデート情報、既知の問題などについて記載します。

『CLUSTERPRO X インストール & 設定ガイド』(Install and Configuration Guide)

CLUSTERPRO を使用したクラスタ システムの導入を行うシステム エンジニアと、クラスタシステム導入後の保守・運用を行うシステム管理者を対象読者とし、CLUSTERPRO を使用したクラスタ システム導入から運用開始前までに必須の事項について説明します。実際にクラスタ システムを導入する際の順番に則して、CLUSTERPRO を使用したクラスタ システムの設計方法、CLUSTERPRO のインストールと設定手順、設定後の確認、運用開始前の評価方法について説明します。

『CLUSTERPRO X リファレンス ガイド』(Reference Guide)

管理者、およびCLUSTERPRO を使用したクラスタ システムの導入を行うシステム エンジニアを対象とし、CLUSTERPRO の運用手順、各モジュールの機能説明、メンテナンス関連情報およびトラブルシューティング情報等を記載します。『インストール & 設定ガイド』を補完する役割を持ちます。

『CLUSTERPRO X 統合WebManager 管理者ガイド』(Integrated WebManager Administrator's Guide)

CLUSTERPRO を使用したクラスタシステムを CLUSTERPRO 統合WebManager で管理するシステム管理者、および統合WebManager の導入を行うシステムエンジニアを対象読者とし、統合WebManager を使用したクラスタシステム導入時に必須の事項について、実際の手順に則して詳細を説明します。

本書の表記規則

本書では、「注」および「重要」を以下のように表記します。

注: は、重要ではあるがデータ損失やシステムおよび機器の損傷には関連しない情報を表します。

重要: は、データ損失やシステムおよび機器の損傷を回避するために必要な情報を表します。

関連情報: は、参照先の情報の場所を表します。

また、本書では以下の表記法を使用します。

表記	使用方法	例
[] 角かっこ	コマンド名の前後 画面に表示される語 (ダイアログ ボックス、メニューなど) の前後	[スタート] をクリックします。 [プロパティ] ダイアログ ボックス
モノスペース フォント (courier)	コマンド ライン、関数、パラメータ	clpstat -s
モノスペース フォント 太字 (courier)	ユーザが実際にコマンドプロンプト から入力する値を示します。	以下を入力します。 clpctl -s -a
モノスペース フォント (courier) <i>斜体</i>	ユーザが有効な値に置き換えて入 力する項目	clpstat -s [-h host_name]

最新情報の入手先

最新の製品情報については、以下のWebサイトを参照してください。

<https://jpn.nec.com/clusterpro/>

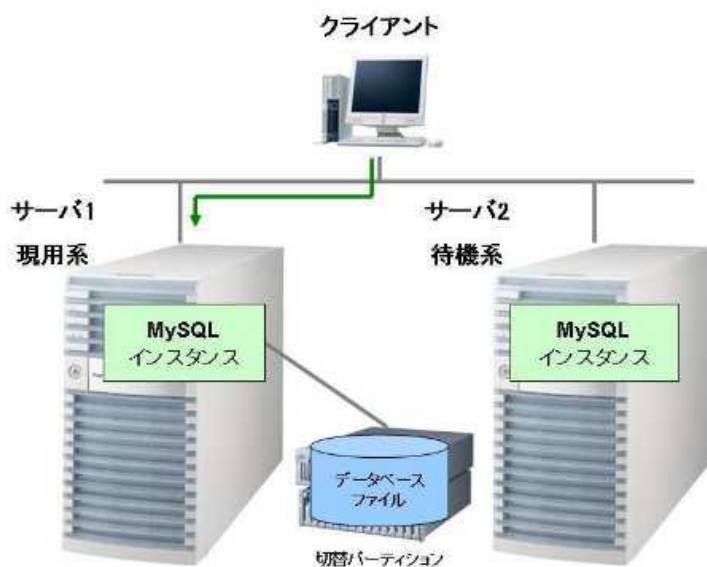
第1章 MySQL

機能概要

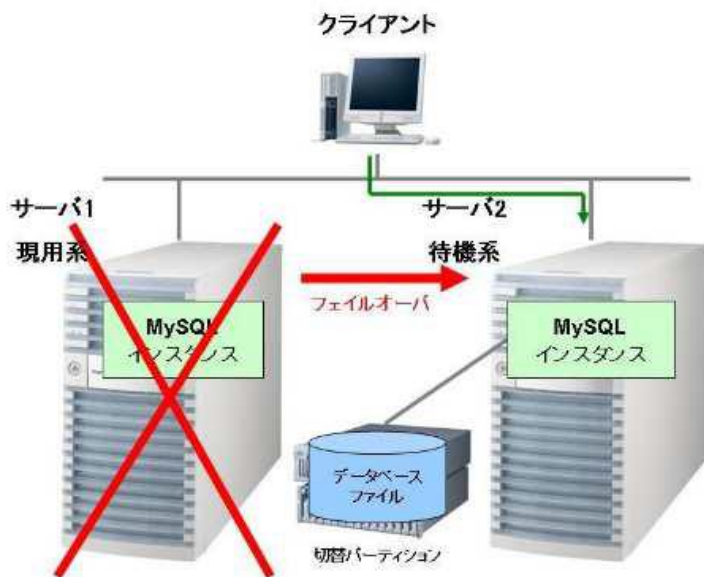
MySQLとCLUSTERPROを連携させることで、以下のクラスタを構成することができます。

【片方向スタンバイ型】

下図は、片方向スタンバイ型の構成例です。サーバ1を現用系、サーバ2を待機系として構成しています。クライアントからは、フローティングIP や仮想コンピュータ名を使用して接続します。

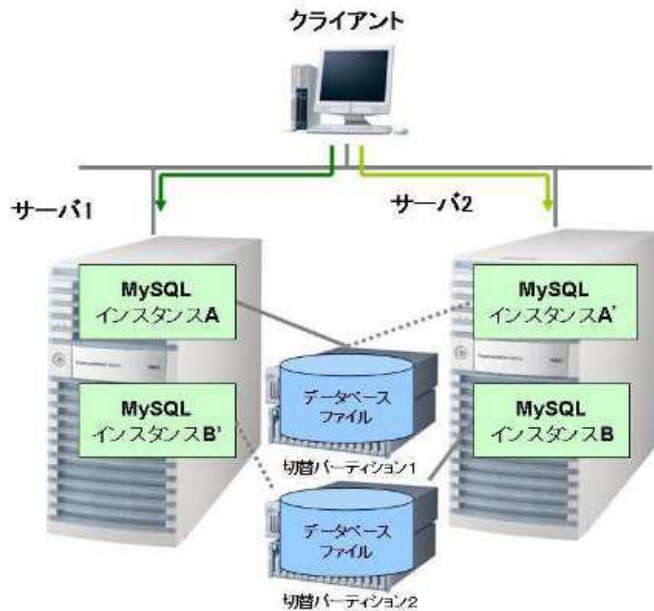


サーバ1で障害が発生すると、切替パーティションがサーバ1からサーバ2に移行します。移行後、サーバ2でMySQL インスタンスが立ち上がり、フェイルオーバーが完了します。フェイルオーバー後も、同一のフローティングIP や仮想コンピュータ名で接続が可能です。

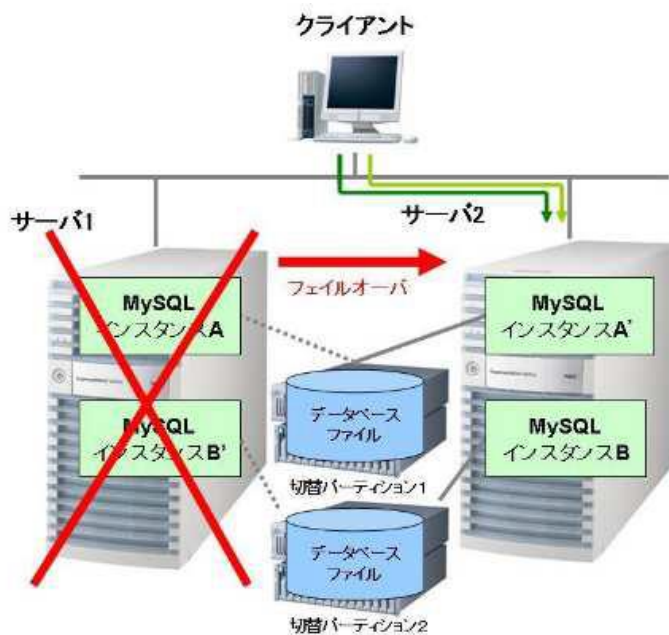


【双方向スタンバイ型】

下図は、双方向スタンバイ型の構成例です。サーバ1はMySQL インスタンスA の現用系であると同時にMySQL インスタンスB の待機系です。サーバ2はMySQL インスタンスA の待機系であると同時にMySQL インスタンスB の現用系です。クライアントからは、フローティングIP や仮想コンピュータ名を使用して接続します。



サーバ1で障害が発生すると、切替パーティション1がサーバ1からサーバ2に移行します。移行後、サーバ2でインスタンスA' が立ち上がり、フェイルオーバーが完了します。この時、サーバ2では2つのMySQL インスタンスが起動している状態になります。フェイルオーバー後も、同一のフローティングIP や仮想コンピュータ名で接続が可能です。



機能範囲

InnoDB Cluster、MySQL NDB Cluster 構成は対象外となります。

動作環境

下記のMySQL、CLUSTERPRO、OSのバージョンの組み合わせで動作確認を行っておりません。これ以外の組み合わせについては、お客様にて十分に動作確認を行ってください。

- MySQLのバージョン
5.7.21

- ※MySQLモニタリソースを使用する場合は、MySQLの有償製品（MySQL サブスクリプションまたはMySQL OEMライセンスのうち、お客様のMySQL利用状況に合致したもの）をご用意ください。

- CLUSTERPROのバージョン

- CLUSTERPRO X 3.3 for Linux (内部バージョン 3.3.4-1)
CLUSTERPRO X Database Agent 3.3

- CLUSTERPRO X 4.0 for Linux (内部バージョン 4.0.0-1)
CLUSTERPRO X Database Agent 4.0

- OSの種類

- Red Hat Enterprise Linux 7.3

インストール手順

MySQLのCLUSTERPRO環境構築は以下の流れで行います。

1. CLUSTERPROのインストール
2. MySQLのインストール
3. インスタンスの作成
4. サービスリソースの設定

1. CLUSTERPROのインストール

CLUSTERPROのインストール&設定ガイドに従い、クラスタ、フェイルオーバーグループ、モニタリソースの作成を行ってください。

フェイルオーバーグループの設定例は下記です。

【片方向スタンバイ型】

クラスタ環境

	サーバ1(現用系)	サーバ2(待機系)
実IPアドレス	192.168.120.1	192.168.120.2
フローティングIP	192.168.120.3	
リッスンポート	3306	
サフィクス	@mysql1	
共有ディスク (切り替えディスク)	/dev/sdb1	
マウントポイント	/mysql1	

【双方向スタンバイ型】

クラスタ環境(グループ1)

	サーバ1(現用系)	サーバ2(待機系)
実IPアドレス	192.168.120.1	192.168.120.2
フローティングIP	192.168.120.3	
リッスンポート	3306	
サフィクス	@mysql1	
共有ディスク (切り替えディスク)	/dev/sdb1	
マウントポイント	/mysql1	

クラスタ環境(グループ2)

	サーバ1(待機系)	サーバ2(現用系)
実IPアドレス	192.168.120.1	192.168.120.2
フローティングIP	192.168.120.4	
リッスンポート	3307	
サフィクス	@mysql2	
共有ディスク (切り替えディスク)	/dev/sdb2	
マウントポイント	/mysql2	

2. MySQLのインストール

各サーバのローカルディスク上にMySQLをインストールします。

3. インスタンスの作成

【片方向スタンバイ型】

- (1) OS起動時のインスタンス起動を無効化する
下記コマンドをすべてのノードで実行します。

```
systemctl disable mysqld
```

- (2) my.cnfファイルの準備
すべてのノードに /etc/my.cnf ファイルを作成します。

datadir	データディレクトリです。共有ディスク上に設定します。
port	クライアントから接続する際に使用するポート番号です
socket	ローカル接続で使用するソケットファイルを指定します。
log-error	エラーログを出力するのファイル名を指定します。
log_timestamps	ログに記録される時刻のフォーマットを指定します。UTC、SYSTEM(ローカルシステム時刻)のいずれかを指定します。

```
[mysqld@mysql1]
port=3306
datadir=/mysql1/data
socket=/mysql1/data/mysql.sock
log-error=/mysql1/data/mysqld.log
log_timestamps=SYSTEM

[mysql@mysql1]
socket=/mysql1/data/mysql.sock
```

- (3) データベースを作成する
下記コマンドを現用系のみで実行してください。

```
systemctl start mysqld@mysql1
```

- (4) エラーログからrootユーザの初期パスワードを確認する
エラーログの出力が下記例の場合、初期パスワードは「3w*b;w)w7y4N」になります。

```
[Note] A temporary password is generated for root@localhost: 3w*b;w)w7y4N
```

- (5) rootユーザでログインする

```
mysql --defaults-group-suffix=@mysql1 -u root -p
```


- (6) rootユーザのパスワードを変更する

```
alter user root@localhost identified by '<パスワード>';
```

- (7) MySQLモニタリソース用のユーザを作成する

```
create user watchuser@server1 identified by '<パスワード>';  
create user watchuser@server2 identified by '<パスワード>';  
grant all on watchdb.* to watchuser@server1;  
grant all on watchdb.* to watchuser@server2;
```

- (8) MySQLモニタリソース用のデータベース、テーブルを作成する

```
create database watchdb;  
use watchdb;  
create table mysqlwatch (num int not null primary key)  
ENGINE=innodb;  
insert into mysqlwatch values (0);  
commit;
```

- (9) インスタンスを停止する

```
systemctl stop mysqld@mysql1
```

【双方向スタンバイ型】

- (1) OS起動時のインスタンス起動を無効化する
下記コマンドをすべてのノードで実行します。

```
systemctl disable mysqld
```

- (2) my.cnfファイルの準備
すべてのノードに /etc/my.cnf ファイルを作成します。

datadir	データディレクトリです。共有ディスク(切り替えディスク)上に設定します。
port	クライアントから接続する際に使用するポート番号です
socket	ローカル接続で使用するソケットファイルを指定します。
log-error	エラーログを出力するのファイル名を指定します。
log_timestamps	ログに記録される時刻のフォーマットを指定します。UTC、SYSTEM(ローカルシステム時刻)のいずれかを指定します。

```
[mysqld@mysql1]
port=3306
datadir=/mysql1/data
socket=/mysql1/data/mysql.sock
log-error=/mysql1/data/mysqld.log
log_timestamps=SYSTEM

[mysql@mysql1]
socket=/mysql1/data/mysql.sock

[mysqld@mysql2]
port=3307
datadir=/mysql2/data
socket=/mysql2/data/mysql.sock
log-error=/mysql2/data/mysqld.log
log_timestamps=SYSTEM

[mysql@mysql2]
socket=/mysql2/data/mysql.sock
```

- (3) データベースを作成する
本手順以降はすべて各クラスタグループの現用系で実行してください。

サーバ1

```
systemctl start mysqld@mysql1
```

サーバ2

```
systemctl start mysqld@mysql2
```

- (4) エラーログからrootユーザの初期パスワードを確認する
エラーログの出力が下記例の場合、初期パスワードは「3w*b;w)w7y4N」になります。

```
[Note] A temporary password is generated for root@localhost: 3w*b;w)w7y4N
```

- (5) rootユーザでログインする

サーバ1

```
mysql --defaults-group-suffix=@mysql1 -u root -p
```

サーバ2

```
mysql --defaults-group-suffix=@mysql2 -u root -p
```

- (6) rootユーザのパスワードを変更する

```
alter user root@localhost identified by '<パスワード>';
```

- (7) MySQLモニタリソース用のユーザを作成する

```
create user watchuser@server1 identified by '<パスワード>';  
create user watchuser@server2 identified by '<パスワード>';  
grant all on watchdb.* to watchuser@server1;  
grant all on watchdb.* to watchuser@server2;
```

- (8) MySQLモニタリソース用のデータベース、テーブルを作成する

```
create database watchdb;  
use watchdb;  
create table mysqlwatch (num int not null primary key)  
ENGINE=innodb;  
insert into mysqlwatch values (0);  
commit;
```

- (9) インスタンスを停止する

サーバ1

```
systemctl stop mysqld@mysql1
```

サーバ2

```
systemctl stop mysqld@mysql2
```

2. EXECリソースの設定

MySQLのEXECリソースをフェイルオーバーグループに設定します。
双方向スタンバイ型の場合、各フェイルオーバーグループに1つずつEXECリソースを設定してください。

- (1) Cluster Manager でサービスを追加するフェイルオーバーグループを選択し、リソースの追加画面を開く
- (2) [タイプ]ボックスでグループリソースのタイプ(execute resource)を選択し、[次へ]をクリックする。

グループ(failover1)のリソース定義

グループリソースの定義

タイプ(T) execute resource

名前(N) exed

コメント(C)

ライセンス情報取得(L)

説明

グループリソースの種類を選択して名前を入力してください。

<戻る(B) 次へ(N)> キャンセル

(3) 依存関係のページが表示されます。[既定の依存関係に従う]のチェックを解除し、ディスクリソース、FIPリソースを設定します。[次へ]をクリックする。

グループ(failover1)のリソース定義

ステップ

- 情報
- ⇒ 依存関係
- 復旧動作
- 詳細

既定の依存関係に従う (E)

依存するリソース (E)

名前	リソースのタイプ
disk1	disk resource
fip1	floating ip resource

< 追加 (D)

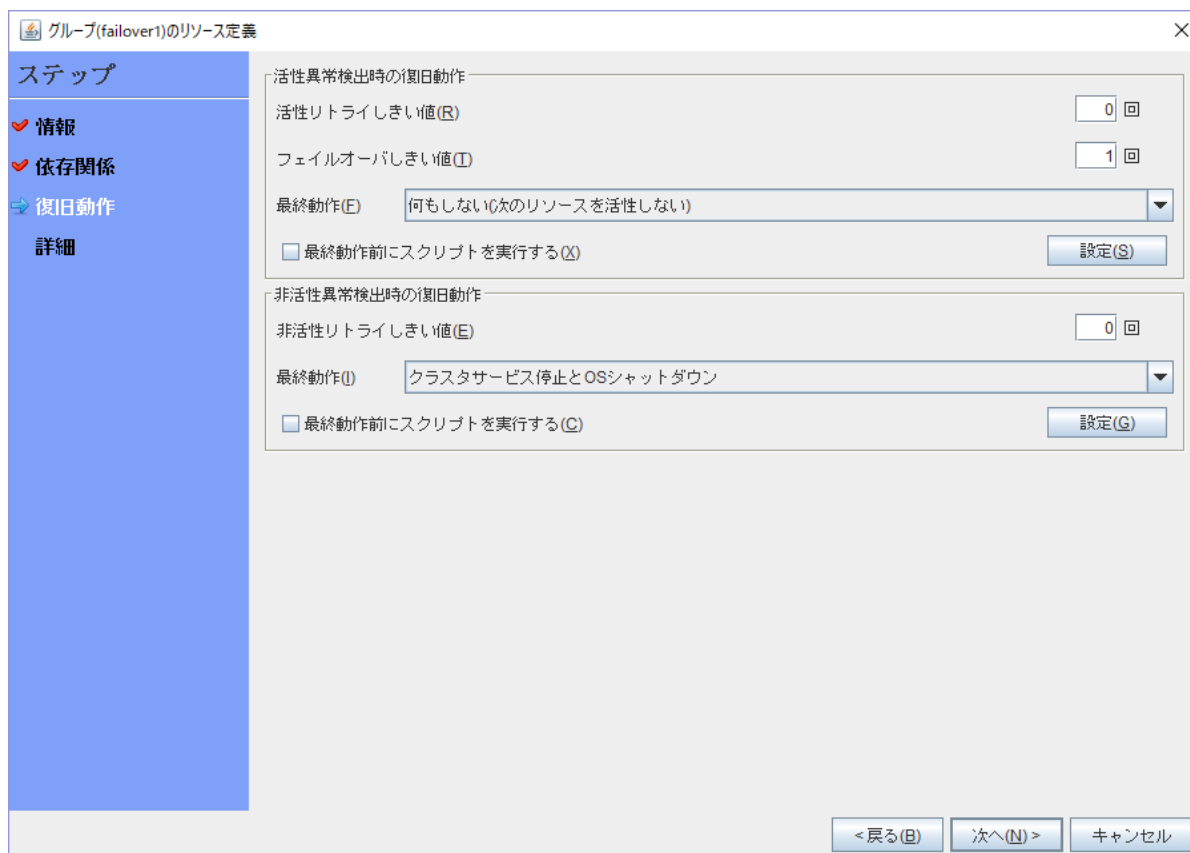
削除 (R) >

利用可能なリソース (V)

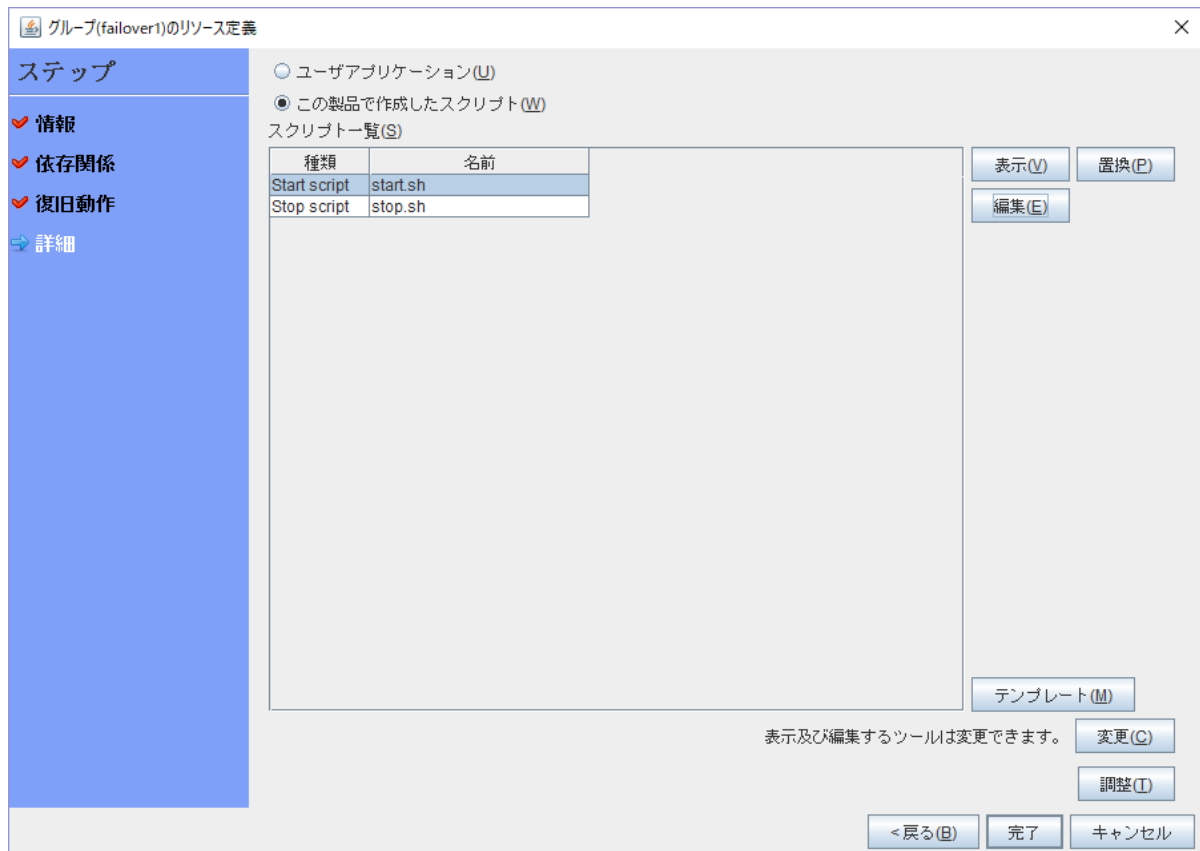
名前

< 戻る (B) 次へ (N) > キャンセル

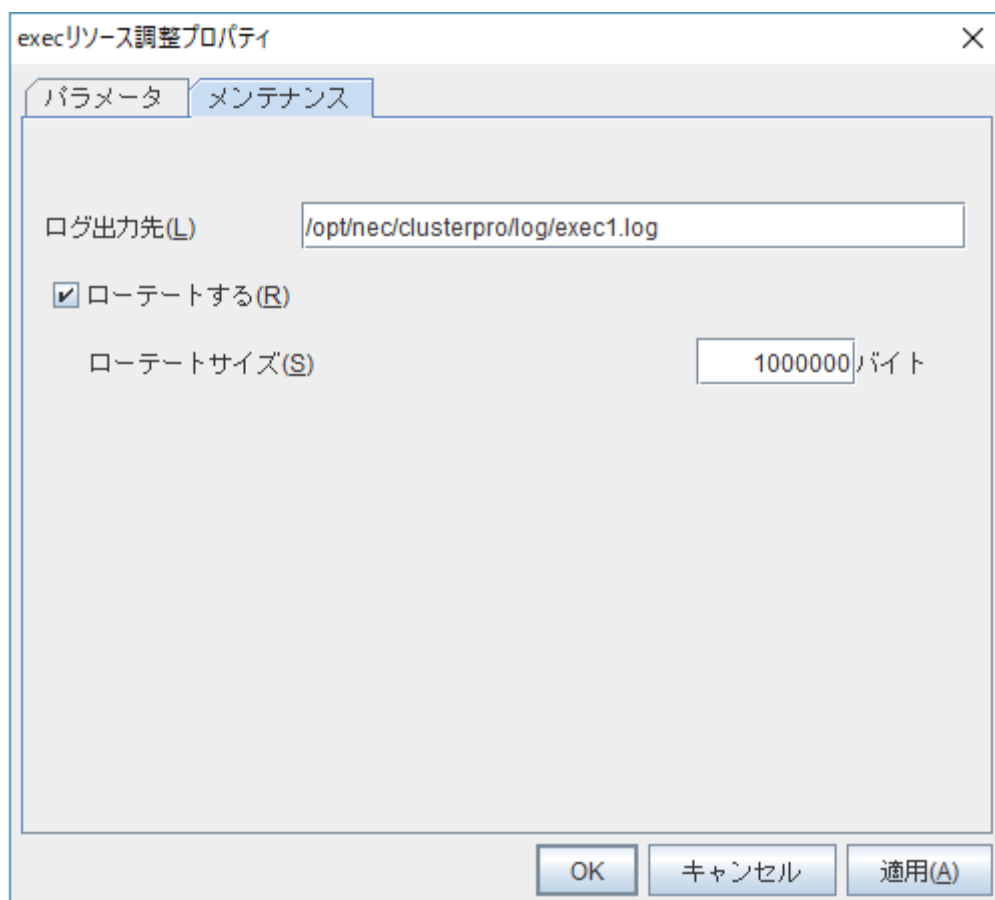
(4) [活性異常検出時の復旧動作]、[非活性異常検出時の復旧動作]が表示されます。[次へ]をクリックする。



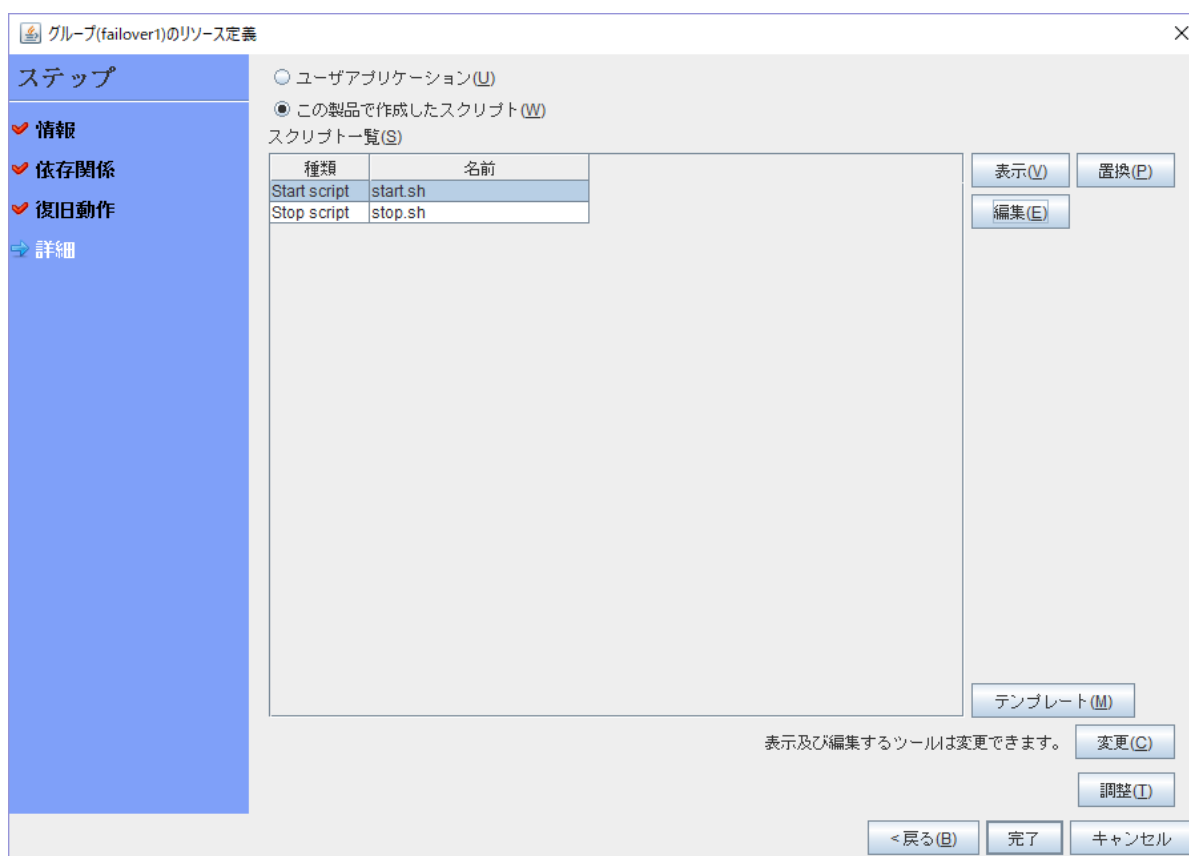
(5) [調整]ボタンをクリックする。



(6) [メンテナンス]タブを開き、ログ出力先、ローテートの設定を行い[OK]をクリックする。



- (7) [編集]ボタンをクリックし、[Start script]、[Stop script]を編集します。スクリプトの記述は後述のサンプルを参考にしてください。編集後[完了]をクリックする。



Start scriptサンプル(サフィクス@mysql1の例)

```
#!/bin/sh
#*****
#*                start.sh                *
#*****
#ulimit -s unlimited
echo "-----start-----"
if [ "$CLP_EVENT" = "START" ]
then
    if [ "$CLP_DISK" = "SUCCESS" ]
    then
        date
        echo "NORMAL1"
        systemctl start mysqld@mysql1
        if [ "$CLP_SERVER" = "HOME" ]
        then
            date
            echo "NORMAL2"
        else
            date
            echo "ON_OTHER1"
        fi
    else
        date
        echo "ERROR_DISK from START"
        exit 1
    fi
elif [ "$CLP_EVENT" = "FAILOVER" ]
then
    if [ "$CLP_DISK" = "SUCCESS" ]
    then
        date
        echo "FAILOVER1"
        systemctl start mysqld@mysql1
        if [ "$CLP_SERVER" = "HOME" ]
        then
            date
            echo "FAILOVER2"
        else
            date
            echo "ON_OTHER2"
        fi
    else
        date
        echo "ERROR_DISK from FAILOVER"
        exit 1
    fi
else
    date
    echo "NO_CLP"
    exit 1
fi
date
echo "EXIT"
echo "-----start-----"
exit 0
```

Stop scriptサンプル(サフィクス@mysql1の例)

```
#!/bin/sh
#*****
#*                stop.sh                *
#*****
#ulimit -s unlimited
echo "-----stop-----"
if [ "$CLP_EVENT" = "START" ]
then
    if [ "$CLP_DISK" = "SUCCESS" ]
    then
        date
        echo "NORMAL1"
        systemctl stop mysqld@mysql1
        if [ "$CLP_SERVER" = "HOME" ]
        then
            date
            echo "NORMAL2"
        else
            date
            echo "ON_OTHER1"
        fi
    else
        date
        echo "ERROR_DISK from START"
        exit 1
    fi
elif [ "$CLP_EVENT" = "FAILOVER" ]
then
    if [ "$CLP_DISK" = "SUCCESS" ]
    then
        date
        echo "FAILOVER1"
        systemctl stop mysqld@mysql1
        if [ "$CLP_SERVER" = "HOME" ]
        then
            date
            echo "FAILOVER2"
        else
            date
            echo "ON_OTHER2"
        fi
    else
        date
        echo "ERROR_DISK from FAILOVER"
        exit 1
    fi
else
    date
    echo "NO_CLP"
    exit 1
fi
date
echo "EXIT"
echo "-----stop-----"
exit 0
```

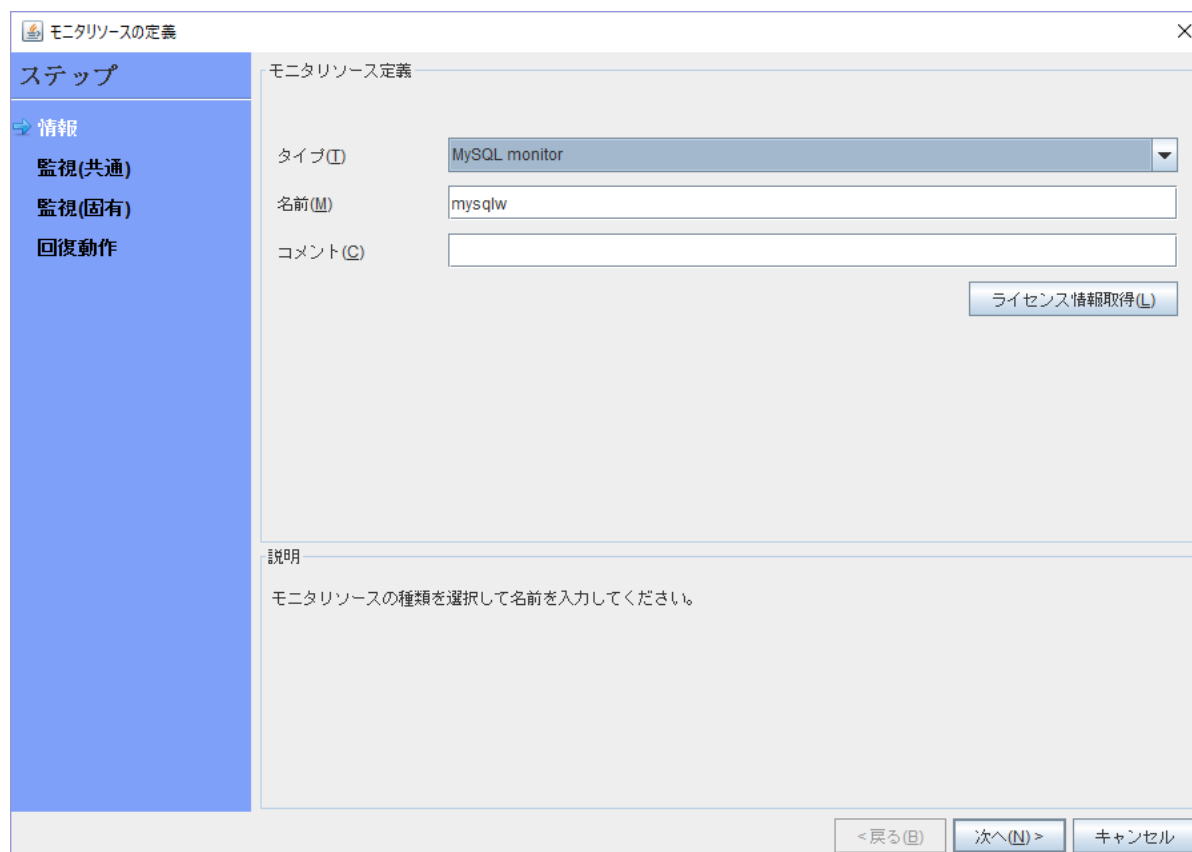
3. モニタリソースの設定

MySQLのモニタリソースを設定します。

双方向スタンバイ型の場合、インスタンスごとにモニタリソースを設定してください。

(1) Cluster Manager で[Monitors]から[モニタリソースの追加]を開く

(2) [タイプ]で[MySQL monitor]を選択し、[次へ]をクリックする。



(3) 対象リソースとして、監視対象のEXECリソースを選択し、[次へ]をクリックする。

モニタリソースの定義

ステップ

情報

監視(共通)

監視(固有)

回復動作

インターバル(I) 60 秒

タイムアウト(T) 120 秒

タイムアウト発生時に監視プロセスのダンプを採取する(D)

タイムアウト発生時にリトライしない(M)

タイムアウト発生時に回復動作を実行しない(O)

リトライ回数(R) 2 回

監視開始待ち時間(S) 0 秒

監視タイミング

常時(L)

活性時(C)

対象リソース exec 参照(W)

nice値(E) 0

監視を行うサーバを選択する サーバ(V)

<戻る(B) 次へ(N)> キャンセル

(4) 表示されたすべての項目を入力する

[データベース名]、[ユーザ名]、[パスワード]、[監視テーブル名]、[ストレージエンジン]には、先に作成したMySQLモナリソース用のユーザ、データベース、テーブルを設定します。

[ライブラリパス]はインストールされているライブラリを指定します。設定後、[次へ]をクリックする。

ステップ	監視レベル(W)	レベル2(update/selectでの監視)
情報	データベース名(D)	watchdb
監視(共通)	IPアドレス(R)	173.14.210.3
監視(固有)	ポート番号(Q)	3306
回復動作	ユーザ名(U)	watchuser
	パスワード(P)	password
	監視テーブル名(T)	mysqlwatch
	ストレージエンジン(E)	InnoDB
	ライブラリパス(L)	/usr/lib64/mysql/libmysqlclient.so.20.3.8

<戻る(B) 次へ(N)> キャンセル

(5) 対象リソースとしてフェイルオーバーグループを選択し、[完了]をクリックする。

モニタリソースの定義

ステップ

回復動作(E) カスタム設定

回復対象 failover1 参照(W)

回復スクリプト実行回数(Y) 0回

再活性化前にスクリプトを実行する(V)

最大再活性化回数(R) 0回

フェイルオーバー実行前にスクリプトを実行する(Q)

フェイルオーバー実行前にマイグレーションを実行する(Q)


最大フェイルオーバー回数(I) 1回

最終動作前にスクリプトを実行する(X)

最終動作(E) クラスタサービス停止とOSシャットダウン

スクリプト設定(S)

<戻る(B) 完了 キャンセル

(6) [設定の反映 ]をクリックし、設定を反映する。

(7) Cluster Managerを操作モードに変更し、フェイルオーバーグループを起動する

動作確認手順

フェイルオーバーグループを起動した後、以下の動作確認を実施してください。

1. クラスタの正常起動
2. クライアントからの接続確認
3. 現用系サーバダウンの疑障(フェイルオーバー)
4. クライアントからの接続確認
5. 現用系サーバ復旧
6. フェイルオーバーグループの移動
7. クライアントからの接続確認