

# MySQL 데이터베이스 동기화 (리플리케이션)

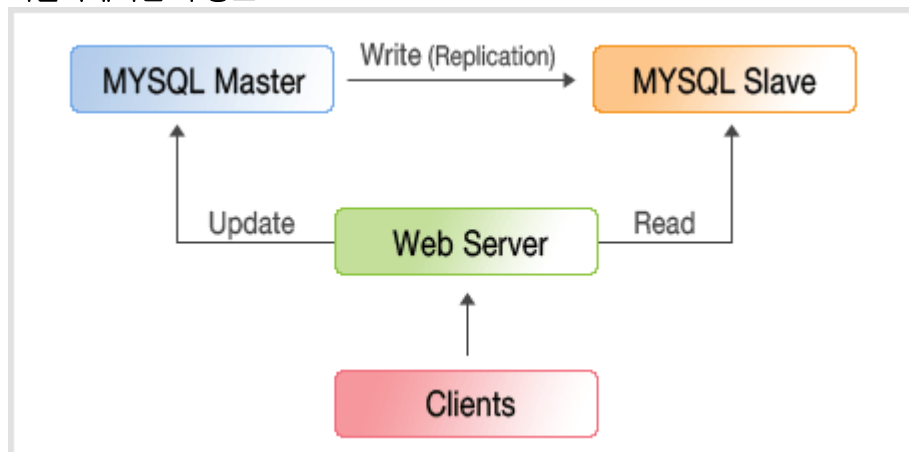
작성자 : 벅스트라인 고객기술지원부 최보근

작성일 : 2009년 02월 23일

## 리플리케이션(Replication)이란?

리플리케이션이란 하나의 **MySQL** 데이터베이스 서버(**Master**)에 있는 데이터를 하나 이상의 다른 **MySQL** 데이터베이스 서버(**Slave**)에 복제하는 것을 의미합니다. **MySQL**: 리플리케이션은 비-동기적으로 구현되어 있다. 비-동기화는 지속적 연결이 아닌 이벤트 발생시에 연결하여 처리하는 것이긴 하나, 실제로 거의 동기화적 수준이라고 봐도 무방하다. 단, 크리티컬한 데이터와 대용량의 데이터 처리를 요구하는 경우에는 **MySQL**의 비-동기화적 방식은 적절치 못하므로 사용을 권장하지 않는다. 리플리케이션의 사용유무는 어디까지나 사용자의 몫이며, 책임도 사용자에게 따른다. 그러나, 이러한 밝혀지지 않은 문제에도 불구하고, **MySQL**의 리플리케이션은 성능이 매우 좋은 편임에는 틀림이 없다. 모든 데이터베이스, 선택한 데이터베이스, 선택한 테이블만 복제하도록 리플리케이션을 구성할 수 있습니다.

## 리플리케이션 구성도



## 리플리케이션의 목적

**시스템 확장** : 로드 분산으로 인하여 전체 시스템의 성능을 개선합니다. 이와 같은 환경에서는 쓰기 및 업데이트가 마스터 서버에서만 실행되어야 하지만, 읽기 연산은 하나 이상의 슬레이브에서 진행 되므로 쓰기, 연산 속도를 개선시키며, 슬레이브 서버 수가 늘어 날수록 읽기 속도가 개선 됩니다.

**데이터 안정성** : 마스터에서 슬레이브로 복제된 데이터를 마스터와 상관없이 슬레이브에서 안전하게 백업을 실행할 수 있습니다.

**분석** : 실제 데이터는 마스터에서 생성되기 때문에 마스터 서버의 성능에 영향을 주지 않은 채로 슬레이브에서 데이터 분석을 실행 할 수 있습니다.

## 리플리케이션 구성

**MySQL** 리플리케이션은 바이너리 로깅 메커니즘을 사용합니다. 마스터로 구동하는 **MySQL** 인스턴스는 데이터베이스 업데이트와 변경 사항을 바이너리 로그에 기록하며, 바이너리 로그에 있는 정보는 기록해야 할 데이터베이스 변경 사항에 따라서 서로 다른 로깅 포맷으로 저장 됩니다. 슬레이브는 마스터에서 바이너리 로그를 읽은 후에 바이너리 로그를 읽은 후에 바이너리 로그에 있

는 이벤트를 자신의 로컬 데이터베이스에서 실행하도록 구성됩니다.

## Master 와 Slave 설치

### 설치환경

CentOS release 5.2, mysql 4.1.21, mysql경로 : /usr/local/mysql

MySQL을 master 와 slave 서버에 설치합니다. 안정성을 위해 두 서버의 버전을 맞춰주는 것이 좋으며, Replication 기능은 3.23.15부터 지원되기 시작하였으나 3.23.32부터 안정화되었다고 알려져 있으므로, 그 이상 혹은 최신 버전의 MySQL 을 설치하길 권장합니다.

### Master 설정

Slave 서버에서 master 서버에 접속 할 REPLICATION SLAVE 권한을 가진 계정을 생성합니다.

```
mysql> grant replication slave on *.*  
-> to 'user_name'@'user_host' identified by 'user_password';
```

4.02 이전 버전의 MySQL에서는 REPLICATION SLAVE 권한이 없으므로 FILE 권한으로 대신합니다.

```
mysql> grant file on *.*  
-> to 'user_name'@'user_host' identified by 'user_password';
```

User\_name : master 서버에 접속할 계정 명

User\_host : slave 서버의 주소/도메인 네임

### Master 데이터를 Slave에 복사

HOT 백업(tar, zip, rsync)

master

```
mysql> flush tables with read lock;  
[root@localhost ~]# tar -cvf mysql-snapshot.tar /usr/local/mysql/data/
```

slave

```
[root@localhost ~]# tar -xvf mysql-snapshot.tar
```

master

```
mysql> unlock tables;
```

mysqldump 백업

```
[root@localhost mysql]# mysqldump -u root -p -A > database.sql
```

### 환경설정

#### Master 환경설정

Master 서버의 /etc/my.cnf 파일의 [mysqld] 섹션을 수정합니다.

```
25 [mysqld]  
26 server-id = 1  
27 log-bin
```

#### Slave 환경설정

Slave 서버의 /etc/my.cnf 파일의 [mysqld] 섹션을 수정합니다.(server-id는 master 서버의 server-id와 다르게 설정하며, slave가 여러 대인 경우 각 slave 서버의 server-id는 각각 달라야 하며 값은 2^32- 1까지 설정 가능합니다.)

```
[mysqld]
server-id      = 2
master-host    = xxx.xxx.xxx.xxx
master-port    = 3306
master-user    = user_name
master-password = user_password
```

HOT 백업을 받은 경우 **slave** 서버의 데이터 디렉토리에 **master** 서버의 데이터를 복사하며, **mysqldump**를 사용한 경우에는 **slave** 서버를 시작한 후에 백업 받은 자료를 복구 합니다.

```
[root@localhost ~]# tar -xvf mysql-snapshot.tar
```

```
[root@localhost ~]# /usr/local/mysql/bin/mysqld_safe &
[root@localhost ~]# mysql -u root -p < database.sql
```

### slave 쓰레드 스타트

```
mysql> start slave;
```

```
090218 14:38:46 [Note] Slave I/O thread: connected to master 'master_user@master_host:3306', replication started in log 'FIRST' at position 4
```

슬레이브 쓰레드가 시작된 후에 **mysql/var/localhost.localdomain.err** 파일을 확인하여 위와 같은 메시지가 있으면 성공적으로 연결이 된 것 입니다.

### 리플리케이션 옵션과 변수

- **master-host**
- **master-user**
- **master-password**
- **master-port**
- **master-connect-retry**
- **master-ssl**
- **master-ssl-ca**
- **master-ssl-capath**
- **master-ssl-cert**
- **master-ssl-cipher**
- **master-ssl-key**

MySQL 5.1.17 버전 이후에는 위의 옵션들을 사용하지 않습니다. 위 옵션들이 변경하는 설정 값은 **mysqld**가 시작될 때 무시되고 **mysqld** 로그에 있는 경고문이 나오게 됩니다. 리플리케이션으로 구성하기 위해서는 반드시 **change master to...** 명령문을 사용해야 합니다. MySQL 5.1.16 이전 버전에서는 **master.info** 파일이 존재한다면 위 옵션들이 무시됩니다.(즉, MySQL 서버가 이미 리플리케이션으로 구성되어 있는 경우) 이 파일이 존재하고 위 옵션들이 **my.cnf** 파일에 들어 있거나 또는 명령어 라인 옵션으로 **mysqld**에 제공되더라도, 그것들은 무시되고 **master.info**에 들어 있는 정보를 대신 사용합니다.

아래의 옵션들은 리플리케이션을 제어하는데 사용하는 스타트업 옵션입니다. **Change master to** 명령문을 사용해서 리셋 시킬 수 있으며, --**replicate-\*** 옵션은 슬레이브 서버가 시작될 때에만 설정 될 수 있습니다.

-- **master-host=host\_name**

마스터 리플리케이션 서버의 호스트 이름 또는 IP

-- **master-info-file=file\_name**

슬레이브가 마스터 정보를 기록하는 파일 이름. 디폴트는 데이터 디렉토리에 있는 **mysql.info**

**--master-user=user\_name**

슬레이브가 서버에 접속할 때 슬레이브 쓰레드가 인증용으로 사용하는 계정의 사용자 이름. 이 계정은 반드시 **REPLICATION SLAVE** 권한을 가지고 있어야 합니다. **master.info** 파일 값이 우선권.

**--master-port=port\_number**

마스터가 슬레이브 접속을 기다리는 **(listening) TCP/IP** 포트 번호. **master.info** 파일에 있는 값이 우선권.

**--master-password=password**

슬레이브가 마스터에 접속을 할 때 슬레이브 쓰레드가 인증용으로 사용하는 계정의 패스워드. **master.info** 파일에 있는 값이 우선권

**--replicate-do-db=db\_name**

리플리케이션을 제한하도록 서버에게 지시한다. 한 개 이상의 데이터 베이스를 지정하기 위해서는, 각각의 데이터 베이스 별로 한 번씩 이 옵션을 사용.

**--replicate-do-table=db\_name.tbl\_name**

지정한 테이블로만 복제하도록 슬레이브 쓰레드에게 지시한다. 한 개 이상의 테이블을 지정하기 위해서는, 각각의 테이블 별로 한 번씩 이 옵션을 사용

**--replicate-ignore-db=db\_name**

디폴트 데이터 베이스가(즉, **USE**가 선택하는 것) **db\_name**인 곳에서는 모든 명령문을 복제하지 않도록 서버에게 지시. 한 개 이상의 데이터 베이스를 무시하도록 하기 위해서는, 각각의 데이터 베이스 별로 한번씩 이 옵션을 사용

**--replicate-ignore-table=db\_name.tbl\_name**

동일한 명령문이 다른 모든 테이블을 업데이트하더라도 지정 테이블에 대한 업데이트 명령문은 모두 복제하지 않도록 슬레이브에게 지시. 한 개 이상의 테이블을 무시하도록 지정하기 위해서는, 각각의 테이블 별로 한번씩 이 옵션을 사용

**--log-slave-updates**

**SQL** 쓰레드가 실행한 업데이트를 자신의 바이너리 로그에 기록하도록 하는 옵션

**--log-warnings**

에러 로그에 보다 자세한 메시지를 기록하도록 만드는 옵션

**--master-connect-retry=seconds**

마스터가 다운 되거나 또는 접속이 끊어지는 경우에 마스터에 재 접속을 시도하기 전에 슬레이브 쓰레드가 슬립(**sleep**)하는 시간. **master.info** 파일에 있는 값이 우선권을 가집니다. 디폴트는 **60**. 재 연결 시도 횟수는 **--master-retry-count** 옵션

**--master-retry-count=count**

슬레이브가 마스터에 접속을 시도하는 횟수 ( 접속 포기 전까지 ). 디폴트는 **86400**.

**--master-ssl, --master-ssl-ca= file\_name, --master-ssl-capath= directory\_name, --master-ssl-cert= file\_name, --master-ssl-cipher= cipher\_list, --master-ssl-key= file\_name**

이 옵션은 **SSL**을 사용하는 마스터 서버 보안 리플리케이션 접속 설정용으로 사용. **master.info** 파일에 있는 값들이 우선권

**--max-relay-log-size=size**

서버가 자동으로 릴레이 로그 파일을 순환시키는 크기. 디폴트 크기는 **1GB**

**--read-only**

슬레이브 서버가 클라이언트에서 오는 업데이트를 받아들이지 못하도록 만드는 옵션

**--relay-log=file\_name**

릴레이 로그용 이름. 디폴트 이름은 **host\_name-relay-bin.nnnnnn** (**host\_name** = 슬레이브 서버 호스트이름, **nnnnnn** = 릴레이 로그가 이만큼의 시퀀스로 생성되었음을 가리키는 것)

**--relay-log-index=file\_name**

릴레이 로그 인덱스 파일의 이름. 디폴트 이름은 데이터 디렉토리 내에 있는 **host\_name-relay-bin.index**이며, 여기에서 **host\_name**은 슬레이브 서버의 이름입니다.

**--relay-log-info-file=file\_name**

릴레이 로그 정보를 기록하기 위한 파일의 이름. 디폴트 이름은 데이터 디렉토리 내에 있는 **relay-log.info**

**--relay-log-purge={0/1}**

릴레이 로그가 더 이상 필요 없게 되는 순간에 자동으로 이 로그를 비워 버리는(**purging**) 것을 활성화 또는 비 활성화. 디폴트는 **1**(활성화)

**--relay-log-space-limit=size**

이 옵션은 슬레이브에 있는 모든 릴레이 로그 전체의 최대 크기(바이트 단위)를 지정합니다. 이 값이 **0**이면 “제한 없음(**no limit**)”을 의미.

**--replicate-rewrite-db=from\_name -> to\_name**

마스터상에서 디폴트 데이터 베이스(즉, **use**가 선택하는 것)가 **from\_name** 이었다면, 그것을 **to\_name**으로 해석하도록 슬레이브에게 지시함.

**--replicate-same-server-id**

슬레이브 서버에서 사용됨. 일반적으로는 디폴트 설정 값인 **0**을 사용하는데, 이 값은 순환 리플리케이션(**circular replication**)에 의한 무한 루프를 방지합니다. 이 값을 **1**로 설정한다면, 슬레이브는 자신의 서버 **ID**를 가지고 있는 이벤트를 건너 띄지 않게 됩니다.

**--replicate-wild-do-table=db\_name.tbl\_name**

업데이트된 모든 테이블이 지정 데이터 베이스와 테이블 이름 패턴을 매치(**match**) 하는 곳에서는 명령문 리플리케이션을 제한하도록 슬레이브 스레드에게 지시합니다. 이러한 패턴에는 ‘%’ 및 ‘\_’ 와일드 카드 문자가 포함될 수 있는데, 이것들은 **LIKE** 패턴 - 매칭 연산자에 대한 것과 동일한 의미를 가집니다. 한 개 이상의 테이블을 지정하기 위해서는, 각각의 테이블 별로 한번씩 이 옵션을 사용합니다.

**--replicate-wild-ignore-table=db\_name.tbl\_name**

입력한 와일드카드 패턴과 테이블이 하나라도 매치가 되는 곳에서는 명령문을 복제하지 못하도록 슬레이브 스레드에게 지시. 한 개 이상의 테이블을 무시하도록 지정하기 위해서는, 각각의 테이블 별로 한번씩 이 옵션을 사용. 이것은 크로스 - 데이터 베이스 업데이트를 실행한다.

**--report-host=slave\_name**

슬레이브를 등록 (**registration**) 하는 동안에 마스터에 보고되는 슬레이브 호스트 이름 또는 **IP** 번

호. 이것은 마스터 서버에서 **SHOW SLAVE HOSTS** 실행하면 얻을 수 있습니다. 슬레이브를 마스터에 등록하고 싶지 않을 경우에는 이 값을 설정하지 않습니다.

**--report-port=slave\_port\_num**

슬레이브를 등록하는 동안에 마스터에 보고되는 슬레이브 접속용 **TCP/IP** 포트 번호. 슬레이브가 디폴트 포트가 아닌 것에 존재하거나, 또는 마스터 서버 또는 다른 클라이언트에서 슬레이브로 접속을 하기 위한 특별 터널(**tunnel**)을 가지고 있는 경우에만 이 값을 설정합니다.

**--report-password=password**

슬레이브가 등록되는 동안 마스터에 보고되는 슬레이브 계정 패스워드. **--show-slave-auth-info** 옵션을 사용하면, 마스터에서 **SHOW SLAVE HOSTS** 명령문으로 이 값을 확인할 수 있습니다.

**--report-user=user\_name**

슬레이브가 등록되는 동안 마스터에 보고되는 슬레이브 계정 사용자 이름. **--show-slave-auth-info** 옵션을 사용하면, 마스터에서 **SHOW SLAVE HOSTS** 명령문으로 이 값을 확인할 수 있습니다.

**--show-slave-auth-info**

**--report-user** 및 **--report-password** 옵션과 함께 실행된 슬레이브의 사용자 이름과 패스워드를 마스터 서버의 **SHOW SLAVE HOSTS** 명령문 결과로 출력합니다.

**--skip-slave-start**

서버가 시작될 때 슬레이브 쓰레드를 시작하지 말도록 서버에게 지시하는 옵션.

**--slave-compressed-protocol={0/1}**

이것을 **1**로 설정하면, 슬레이브 / 마스터 프로토콜에 대해서 압축을 사용(양쪽 서버가 모두 지원을 할 경우)

**--slave-load-tmpdir=file\_name**

슬레이브가 임시 파일을 만드는 디렉토리의 이름. 이 옵션은 **tmpdir** 시스템 변수 값과 동일한 것을 디폴트로 사용. 이 옵션은 충분한 사용 가능 공간을 가지고 있는 파일 시스템 디렉토리에 임시 파일을 저장하도록 슬레이브에 지시할 때 사용을 권장.

**--slave-net-timeout=seconds**

접속이 끊어져서 읽기가 중단 되었기 때문에 재 접속을 시도해야 한다고 슬레이브가 판단을 하기 전에 마스터에서 오는 데이터를 기다리는 대기 시간. 첫 번째 재 접속은 이 타임 아웃이 끝나면 즉시 시도됩니다. 재시도 간격(**interval**)은 **--master-connect-retry** 옵션으로 제어할 수 있습니다. 디폴트는 **3600초(1 시간)**

**--slave-skip-errors=[ err\_code1 , err\_code2 ,... /all]**

일반적으로, 슬레이브에서 에러가 발생하면 리플리케이션은 중단되며, 이러한 경우에는 데이터 일관성을 수동으로 해결해야 합니다. 이 옵션은 옵션 값에 열거되어 있는 에러 중의 하나를 명령문이 리턴 하더라도 리플리케이션을 계속 진행하도록 슬레이브 **SQL** 쓰레드에게 지시합니다. 에러 발생 원인에 대해 충분히 이해하고 있지 않으면 이 옵션을 사용하지 말도록 합니다.

주로 빨간색 표시된 부분의 옵션이 많이 사용되며, 그 아래의 옵션은 경우에 따라서 필요하실 때 사용하시면 됩니다.

감사합니다.