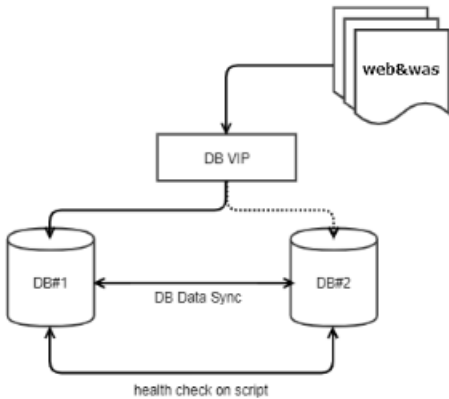


corosync / pacemaker 기반의 DB이중화

1. 패키지 설치
2. DB이중화 구성



3. DB 모니터링을 위한 스크립트 설치

```
$> vi /etc/init.d/db_check
#!/bin/bash

case "$1" in
"start")
    echo "start"
    ;;
"stop")
    echo "stop"
    ;;
"status")
    mysql --connect-timeout 2 -u monitor -p'모니터 계정패스워드' -h localhost -Nse 'select now()'
    result_code=$?

    if [[ $result_code -eq 1 ]]
    then
        echo "[$(date +%Y-%m-%d-%H:%M:%S)] DB Connection Error" >> /var/log/db_check
        pcs resource move DB_GROUP {{ 상대방 IP }}
    fi
    ;;
*)
    echo "start|stop|status"
    exit 1
    ;;
esac
```

4. DB이중화 구성을 위한 리소스 설정

```
$> pcs cluster setup db_ha {{ DB1_IP }} {{ DB2_IP }} --force
$> pcs cluster start --all
$> pcs property set stonith-enabled=false
$> pcs property set no-quorum-policy=ignore
$> pcs resource defaults update resource-stickiness=100
$> pcs resource create DB_CHECK service:db_check op status timeout=10s interval=10s --group DB_GROUP
$> pcs resource create DB_VIP ocf:heartbeat:IPaddr2 ip={{ DB_VIP }} cidr_netmask=32 op monitor interval=10s --group DB_GROUP
```

