クォーラム取得にディスクレスノードを用いた３ノードクラスタ

サイオステクノロジー株式会社

バージョン 1.1 2019/05/23

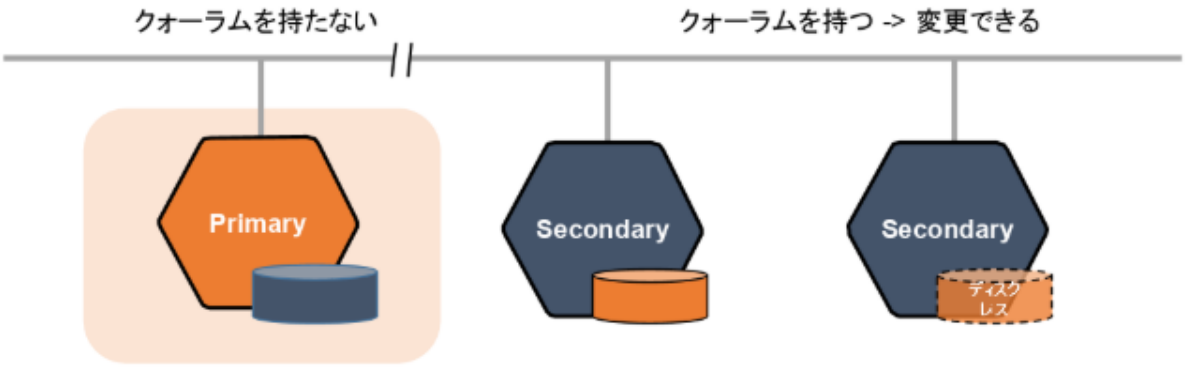
目次

1. はじめに

スプリットブレインを回避するためには、フェンシングを構成する必要がありますが、最後には冗長な通信路に依存し構成が複雑になりがちです。２ノードの構成にディスクレスノードを追加し、３ノード構成にすることで、スプリットブレインの回避に DRBD のクォーラム実装を使用することができます。ここでは実際の動作を確認してみて、その有効性を検証してみます。

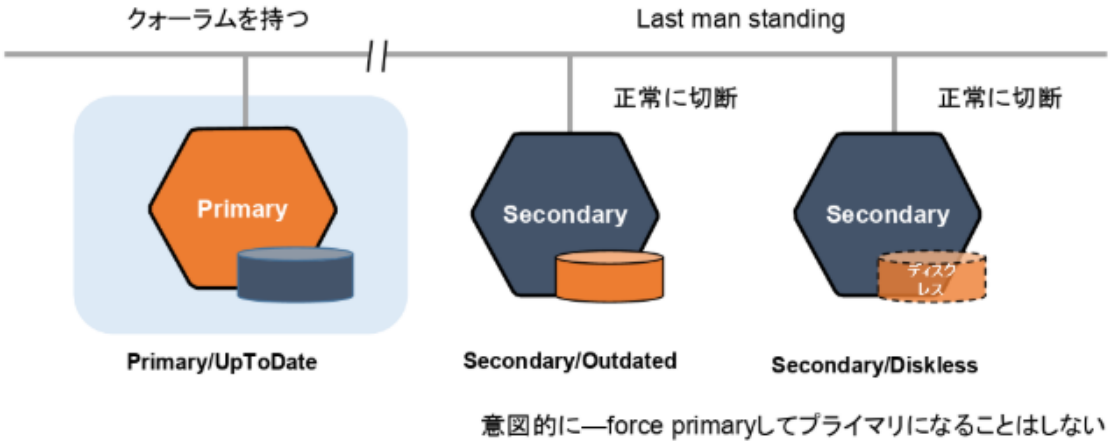
1.1. DRBDのクォーラム実装

基本的な考え方は、通信できるノードの数がノード総数の半分を超える場合、クラスタパーティションは複製されたデータセットを変更できるということです。そのようなパーティションのノードは、クォーラムを持つといいます。言い換えると、クォーラムを持たないノードは、複製されたデータを変更しないことを保証します。よってクォーラムを持つパーティションのみがデータを変更でき、データの相違が発生しないことになります。



DRBD のクォーラム実装は quorum リソースに majority , all または数値を設定することで有効にでき、前の文章の説明が majority の動作になります。今回の検証ではこの一般的な majority を使います。

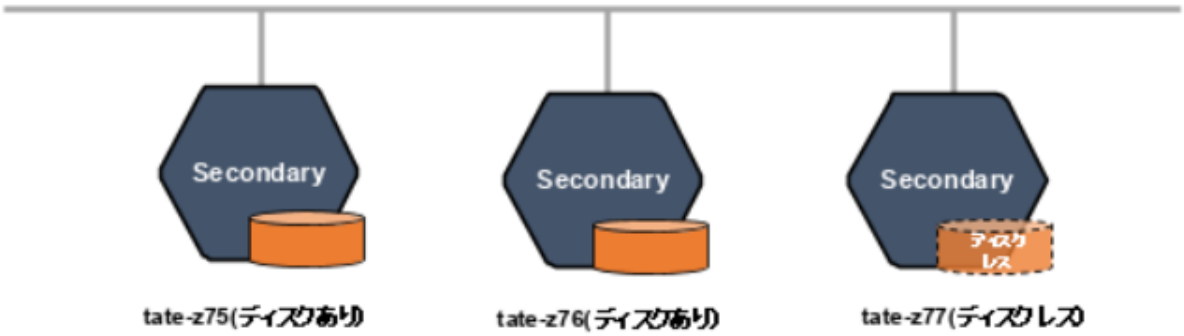
また１つ重要な概念 "Last man standing” というのがあります。これは DRBD の接続を正常に切断した場合は、これらのノードのディスクは Outdated とマークされていてこれらがクォーラムを取得することはないという前提で動作します。これにより、３ノードの１つのプライマリを残して２つのセカンダリが正常に切断したあとに、セカンダリとの回線が遮断された場合でも１つのプライマリだけで動作し続けることが可能になります。



2. インストール

2.1. 前提条件

以下のような３台のノードを仮定します。



検証の前提条件として

- すべてのノードに CentOS 7.6, DRBD/pacemaker/LINSTOR パッケージがインストール済み。
- /etc/drbd.d/r0.res リソースファイルは以下とする。
- tate-z75, tate-z76 はディスクフル、tate-z77 はディスクレスノードとする。

```
resource r0 {
  options
  {
    quorum majority;
    # This should be set after initial setup.
    #quorum-minimum-redundancy 2;
    on-no-quorum suspend-io;
  }

  # handlers {
  #   # Reboot
  #   quorum-lost "echo b > /proc/sysrq-trigger";
  # }

  net {
    protocol C;
  }

  on tate-z75
  {
    disk        /dev/drbdpool/disk1;
    meta-disk   internal;
    device      minor 0;
    address    192.168.192.175:7000;
    node-id    0;
  }

  on tate-z76
  {
    disk        /dev/drbdpool/disk1;
    meta-disk   internal;
    device      minor 0;
    address    192.168.192.176:7000;
    node-id    1;
  }

  on tate-z77
  {
    disk        none;
    meta-disk   internal;
    device      minor 0;
    address    192.168.192.177:7000;
    node-id    2;
  }

  connection-mesh {
    hosts tate-z75 tate-z76 tate-z77;
  }

}
```

2.2. 初期同期

まず、ディスクフルノードでメタデータ領域を初期化し、ディスクレスノードも含めて互いに接続します。接続後は、3 ノードでパーティションを構成し、ディスクが Inconsistent であり、どれか 1 つのディスクフルノードをプライマリにし初期同期を行います。

ディスクフルノードで以下を実行します。

```
# drbdadm create-md r0
```

各ノードで以下を実行します。

```
# drbdadm up r0
```

tate-z75 で status を確認します。

```
[root@tate-z75 ~]# drbdadm status
r0 role:Secondary
  disk:Inconsistent
  tate-z76 role:Secondary
    peer-disk:Inconsistent
  tate-z77 role:Secondary
    peer-disk:Diskless
```

tate-z75 で手動で強制的にプライマリにし、直ぐにセカンダリに戻します。以降は自動プロモーションでプライマリ、セカンダリが切り替わります。

```
[root@tate-z75 ~]# drbdadm --force primary r0
[root@tate-z75 ~]# drbdadm secondary r0
```

次に、自動プロモーションで tate-z75 をプライマリにし、ファイルシステムを構築します。

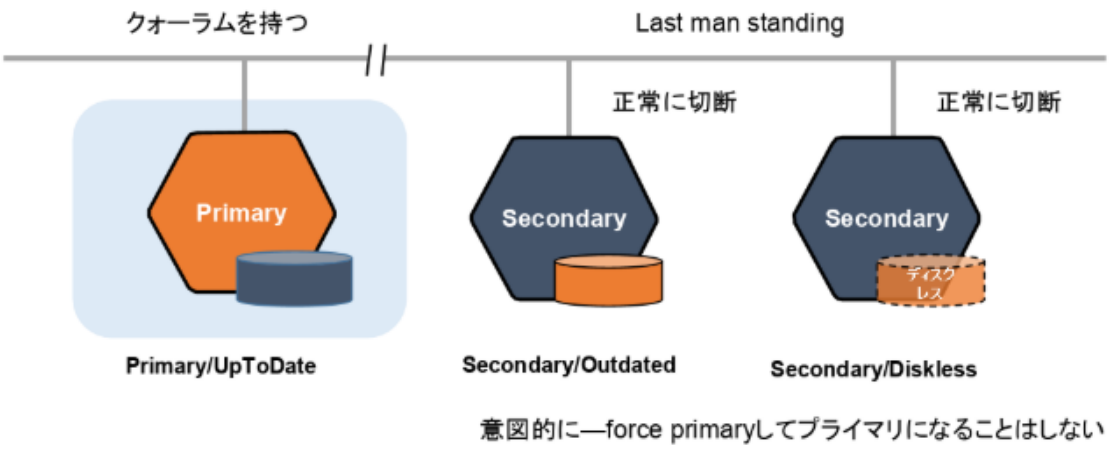
```
[root@tate-z75 ~]# mkfs.xfs -f /dev/drbd0
meta-data=/dev/drbd0             isize=512    agcount=4, agsize=314603 blks
         =                       sectsz=512   attr=2, projid32bit=1
         =                       crc=1        finobt=0, sparse=0
data     =                       bsize=4096   blocks=1258410, imaxpct=25
         =                       sunit=0      swidth=0 blks
naming   =version 2              bsize=4096   ascii-ci=0 ftype=1
log      =internal log           bsize=4096   blocks=2560, version=2
         =                       sectsz=512   sunit=0 blks, lazy-count=1
realtime =none                   extsz=4096   blocks=0, rtextents=0
[root@tate-z75 ~]# mount /dev/drbd0 /mnt
```

3. 検証

3.1. セカンダリノードを正常に切断

セカンダリノードを正常に切断すると、Last man standing 機能により1 つのプライマリだけで動作し続けます。

```
[root@tate-z75 ~]# ssh tate-z76 drbdadm down r0
[root@tate-z75 ~]# ssh tate-z77 drbdadm down r0
[root@tate-z75 ~]# drbdadm status
r0 role:Primary
  disk:UpToDate
  tate-z76 connection:Connecting
  tate-z77 connection:Connecting
[root@tate-z75 ~]# cd /mnt
[root@tate-z75 mnt]# echo "AAA"> BBB
[root@tate-z75 mnt]# ls
BBB
```



3.2. セカンダリノードとの経路を遮断

次に 2 つのセカンダリノードとの接続を維持したまま経路を切断してみます。この場合 セカンダリは UpToDate のままで、ディスクレスノードにより半数を超えるセカンダリ側がクォーラムを持ちます。よって、プライマリ側はクォーラムを失い、IO がフリーズします。

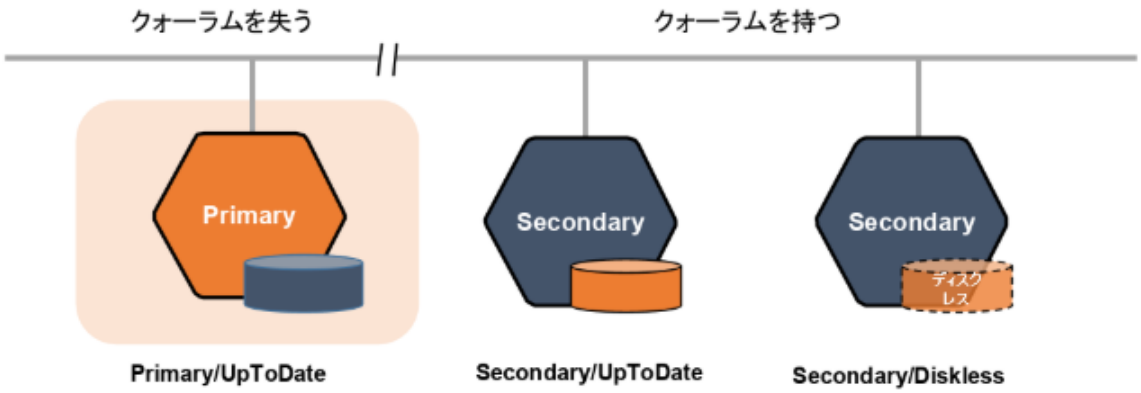
```
[root@tate-z75 ~]# iptables -I INPUT -i eth2 -j DROP
[root@tate-z75 ~]# drbdadm status
r0 role:Primary suspended:quorum
  disk:UpToDate quorum:no blocked:upper
  tate-z76 connection:Connecting
  tate-z77 connection:Connecting
[root@tate-z75 ~]# umount /mnt
  <Hang>...
```

tate-z76 のログには以下が出力されます。

```
May 22 15:28:35 tate-z76 kernel: drbd r0/0 drbd0: Would lose quorum, but using tiebreaker
logic to keep
```

そのまま接続を回復するとハング状態から復帰しますが、２つのセカンダリ側で一度プライマリにしてしまったあとに接続を回復すると、接続が拒否されます。この場合はリブート以外復旧できません。

```
May 22 16:29:41 tate-z75 kernel: drbd r0/0 drbd0: I shall become SyncTarget, but I am
primary!
```



3.3. handlers quorum-lost を設定

handlers quorum-lost を設定することで、クォーラムを失った時の動作を設定できます。アプリがハングしてしまって何もできなくなってしまう場合は、以下のようにしてリブートで復旧させるのがよいかもしれません。

```
[root@tate-z75 ~]# cat /etc/drbd.d/r0.res
resource r0 {
  :
  handlers {
    # Reboot if quorum lost.
    quorum-lost "echo b > /proc/sysrq-trigger";
  }
:
```

この設定で２つのセカンダリノードとの接続を遮断するとクォーラムを失い、リブートします。

```
[root@tate-z75 ~]# mount /dev/drbd0  /mnt
[root@tate-z75 ~]# iptables -I INPUT -i eth2 -j DROP
[root@tate-z75 ~]# Connection to 127.0.0.1 closed by remote host.
Connection to 127.0.0.1 closed.
```

3.4. on-no-quorum io-error を設定

on-no-quorum で io-error を設定することで、クォーラムを失った場合は、IO フリーズでなく IO エラーにすることもできます。この場合 handlers は使わないのでコメントにします。

```
[root@tate-z75 ~]# cat /etc/drbd.d/r0.res
resource r0 {
  options
  {
    quorum majority;
    #quorum-minimum-redundancy 2;
    on-no-quorum io-error;
  }

  #handlers {
  #  # Reboot
  #  quorum-lost "echo b > /proc/sysrq-trigger";
  #}
```

```
[root@tate-z75 ~]# mount /dev/drbd0  /mnt
[root@tate-z75 ~]# iptables -I INPUT -i eth2 -j DROP
[root@tate-z75 ~]# ls /mnt
BBB
[root@tate-z75 ~]# touch /mnt/CCC
touch: cannot touch '/mnt/CCC': Input/output error
```

io-error の場合 umount はできるようなので、その後に接続が回復すれば、復旧できます。

```
[root@tate-z75 ~]# umount /mnt
[root@tate-z75 ~]# iptables -D INPUT -i eth2 -j DROP
[root@tate-z75 ~]# drbdadm status
r0 role:Secondary
  disk:UpToDate
  tate-z76 role:Secondary
    peer-disk:UpToDate
  tate-z77 role:Primary
    peer-disk:UpToDate
```

3.5. quorum-minimum-redundancy を設定

quorum-minimum-redundancy 2 に設定することで、UpToDate のディスクを持つノードを最低２ノード必要と設定できます。

```
[root@tate-z75 ~]# cat /etc/drbd.d/r0.res
resource r0 {
  options
  {
    quorum majority;
    quorum-minimum-redundancy 2;
    on-no-quorum io-error;
  }
:
```

試しに、 invalidate-remote でセカンダリのメタデータを初期化し、フル同期を強制します。

```
[root@tate-z75 ~]# drbdadm invalidate-remote r0
[root@tate-z75 ~]# drbdadm status
r0 role:Secondary
  disk:UpToDate quorum:no
  tate-z76 role:Secondary
    replication:SyncSource peer-disk:Inconsistent done:3.48
  tate-z77 role:Secondary
    peer-disk:Diskless
```

この時点で、UpToDate が１つしかないのでプライマリにできません。

```
[root@tate-z75 ~]# mount /dev/drbd0 /mnt
mount: /dev/drbd0 is write-protected, mounting read-only
mount: mount /dev/drbd0 on /mnt failed: No data available

[root@tate-z75 ~]# drbdadm primary r0
r0: State change failed: (-25) No quorum
additional info from kernel:
1 of 2 nodes up_to_date, need 2 for quorum-minimum-redundancy
Command 'drbdsetup primary r0' terminated with exit code 11
```

同期が終わるとプライマリにできるようになります。常に UpToDate を2つ以上維持してから使用するような重要なデータに対しては有用でしょう。

4. pacemaker との連携

ここでは DRBD9 のクォーラムを使用した３ノード で corosync/pacemaker での mariadb の構築例を示します。

クォーラムを失った時に mariadb がハングするのを防ぐため、失った時はリブートするようにします。

```
[root@tate-z75 ~]# cat /etc/drbd.d/r0.res
resource r0 {
  options
  {
    quorum majority;
    # This should be set after initial setup.
    #quorum-minimum-redundancy 2;
    on-no-quorum suspend-io;
  }

  handlers {
    # Reboot
    quorum-lost "echo b > /proc/sysrq-trigger";
  }
  :
```

DRBD9 で自動プロモーションで drbd を使用する場合は、pacemaker から起動停止を管理を使用せず、あらかじめ drbd service で起動しておきます。

```
# systemctl enable drbd
# systemctl start drbd
```

corosync の３ノードの設定です。２ノードとの違いは node を１つ追加し、two_node を削除、expected_votes が３になっています。

```
# cat /etc/corosync/corosync.conf
totem {
        version: 2

        crypto_cipher: none
        crypto_hash: none
        cluster_name: cluster_test_76_77_78

        transport: udpu
        rrp_mode: passive
}

logging {
        fileline: off
        to_stderr: no
        to_logfile: yes
        logfile: /var/log/corosync.log
        to_syslog: yes
        debug: off
        timestamp: on
        logger_subsys {
                subsys: QUORUM
                debug: off
        }
}

quorum {
        provider: corosync_votequorum
        expected_votes: 3
}

nodelist {
        node {
                ring0_addr: 192.168.192.175
                ring1_addr: 192.168.193.175
                nodeid: 1
        }
        node {
                ring0_addr: 192.168.192.176
                ring1_addr: 192.168.193.176
                nodeid: 2
        }
        node {
                ring0_addr: 192.168.192.177
                ring1_addr: 192.168.193.177
                nodeid: 3
        }
}
```

pacemaker の設定です。自動プロモーションを使用するので、DRBD のリソース設定は不要で、Filesystem, mariadb, 仮想アドレスのリソースを設定します。

```
[root@tate-z75 ~]# crm configure show
node 1: tate-z75
node 2: tate-z76
node 3: tate-z77
primitive res_filesystem Filesystem \
    params device="/dev/drbd0" directory="/var/lib/mysql" fstype=xfs options=noatime \
    op start interval=0 timeout=60 \
    op stop interval=0 timeout=60 \
    op monitor interval=20 timeout=40
primitive res_mysql mysql \
    params binary="/usr/libexec/mysqld" pid="/var/run/mariadb/mariadb.pid" \
    op start interval=0 timeout=120 \
    op stop interval=0 timeout=120 \
    op monitor interval=20 timeout=30 OCF_CHECK_LEVEL=1
primitive vip IPaddr2 \
    params ip=10.1.12.182 cidr_netmask=16 nic=eth1 \
    op start interval=0 timeout=20 \
    op stop interval=0 timeout=20 \
    op monitor interval=10 timeout=20 \
    meta target-role=Started
group rg_service res_filesystem res_mysql vip
location l1 rg_service 100: tate-z75
property cib-bootstrap-options: \
    stonith-enabled=false \
    no-quorum-policy=ignore \
    have-watchdog=false \
    dc-version="1.1.19.linbit-8+20181129.el7.2-c3c624ea3d" \
    cluster-infrastructure=corosync \
    cluster-name=cluster_test_76_77_78 \
    last-lrm-refresh=1558513341
rsc_defaults rsc-options: \
    resource-stickiness=200
```

正しく設定されていると、以下のようになります。

```
[root@tate-z75 ~]# crm_mon -rf1
Stack: corosync
Current DC: tate-z76 (version 1.1.19.linbit-8+20181129.el7.2-c3c624ea3d) - partition with
quorum
Last updated: Wed May 22 17:48:54 2019
Last change: Wed May 22 17:43:10 2019 by root via cibadmin on tate-z75

3 nodes configured
3 resources configured

Online: [ tate-z75 tate-z76 tate-z77 ]

Full list of resources:

 Resource Group: rg_service
     res_filesystem (ocf::heartbeat:Filesystem):    Started tate-z75
     res_mysql  (ocf::heartbeat:mysql): Started tate-z75
     vip    (ocf::heartbeat:IPaddr2):   Started tate-z75

Migration Summary:
* Node tate-z75:
* Node tate-z77:
* Node tate-z76:
```

ここで、 tate-z75 で 192.168.192.175 のリンクを遮断してみます。

```
[root@tate-z75 ~]# iptables -I INPUT -i eth2 -j DROP
[root@tate-z75 ~]# Connection to 127.0.0.1 closed by remote host.
Connection to 127.0.0.1 closed.
```

rg_service は tate-z76 にフェイルオーバーし tate-z75 はクォーラムを失い、handler によりリブートします。リブート後、再度クラスタに参加します。

```
[root@tate-z77 ~]# crm_mon -rf1
Stack: corosync
Current DC: tate-z76 (version 1.1.19.linbit-8+20181129.el7.2-c3c624ea3d) - partition with
quorum
Last updated: Wed May 22 17:51:38 2019
Last change: Wed May 22 17:43:10 2019 by root via cibadmin on tate-z75

3 nodes configured
3 resources configured

Online: [ tate-z76 tate-z77 ]
OFFLINE: [ tate-z75 ]

Full list of resources:

 Resource Group: rg_service
     res_filesystem (ocf::heartbeat:Filesystem):    Started tate-z76
     res_mysql  (ocf::heartbeat:mysql): Started tate-z76
     vip    (ocf::heartbeat:IPaddr2):   Started tate-z76

Migration Summary:
* Node tate-z77:
* Node tate-z76:
```

5. LINSTOR での設定

LINSTOR でクォーラムを使う場合は drbd-options で設定します。

```
[root@tate-z60 ~]# linstor resource-definition create r0
[root@tate-z60 ~]# linstor resource-definition drbd-options --quorum majority r0
[root@tate-z60 ~]# linstor resource-definition drbd-options --on-no-quorum suspend-io r0
[root@tate-z60 ~]# linstor volume-definition create r0 1G
[root@tate-z60 ~]# linstor resource create r0 --storage-pool pool_loop --auto-place 2
[root@tate-z60 ~]# linstor resource create tate-z62 r0 --diskless
[root@tate-z60 ~]# linstor resource list
+-----------------------------------------------------+
| ResourceName | Node     | Port | Usage  |    State |
|-----------------------------------------------------|
| r0           | tate-z60 | 7000 | Unused | UpToDate |
| r0           | tate-z61 | 7000 | Unused | UpToDate |
| r0           | tate-z62 | 7000 | Unused | Diskless |
+-----------------------------------------------------+

[root@tate-z60 ~]# mkfs.xfs /dev/drbd1000
[root@tate-z60 ~]# mount /dev/drbd1000 /mnt
```

ここで network を切断するとクォーラムを失って書けなくります。

```
[root@tate-z60 ~]# iptables -I INPUT -i eth1 -j DROP

[root@tate-z60 ~]# drbdadm status
r0 role:Secondary suspended:quorum
  disk:UpToDate quorum:no blocked:upper
  tate-z61 connection:Connecting
  tate-z62 connection:Connecting

[root@tate-z60 ~]# touch /mnt/T
```

ディスクレスノードと通信できる tate-61 にクォーラムが移ります。

```
[root@tate-z61 ~]# drbdsetup -v status
r0 node-id:1 role:Secondary suspended:no
  volume:0 minor:1000 disk:UpToDate quorum:yes blocked:no
  tate-z60 node-id:0 connection:Connecting role:Unknown congested:no
      ap-in-flight:0 rs-in-flight:0
    volume:0 replication:Off peer-disk:DUnknown resync-suspended:no
  tate-z62 node-id:2 connection:Connected role:Secondary congested:no
      ap-in-flight:0 rs-in-flight:0
    volume:0 replication:Established peer-disk:Diskless peer-client:yes
        resync-suspended:no

[root@tate-z61 ~]# grep tie /var/log/messages
:
May 22 18:23:58 tate-z61 kernel: drbd r0/0 drbd1000: Would lose quorum, but using
tiebreaker logic to keep
```

6. 変更履歴

6.1. v1.1 2019/05/23

- ディスクレスノードを使用した 3 ノード構成に変更

6.2. v1.0 2019/01/31

- サイオステクノロジーロゴ追加