

第2回 Linux-HA Japan 勉強会

2011年6月3日
NTTデータ先端技術株式会社

池田淳子



本日のお題

crm の歩き方

目次

- 1.crm シェルとは
- 2.crm シェルのインストール
- 3.crm シェルでリソースを設定する
- 4.crm シェルでクラスタを管理する



1. crm シェルとは



crm シェルとは

- Pacemakerに付属したコマンドラインツールです
- リソースの設定やクラスタの管理を行うことができます



Pacemaker の設定ファイル

/var/lib/heartbeat/crm/cib.xml

```
<cib validate-with="pacemaker-1.0" crm_feature_set="3.0.1" have-quorum="1" dc-uuid="22222222-2222-2222-2222-222222222222"
dmin_epoch="0" epoch="7" num_updates="0" cib-last-written="Tue May 31 14:33:00 2011">
  <configuration>
    <crm_config>
      <cluster_property_set id="cib-bootstrap-options">
        <nvpair id="cib-bootstrap-options-dc-version" name="dc-version" value="1.0.11-db98485d06ed stable=1.0 tip"/>
        <nvpair id="cib-bootstrap-options-cluster-infrastructure" name="cluster-infrastructure" value="Heartbeat"/>
        <nvpair id="cib-bootstrap-options-no-quorum-policy" name="no-quorum-policy" value="ignore"/>
        <nvpair id="cib-bootstrap-options-stonith-enabled" name="stonith-enabled" value="false"/>
        <nvpair id="cib-bootstrap-options-startup-fencing" name="startup-fencing" value="false"/>
      </cluster_property_set>
    </crm_config>
    <nodes>
      <node id="11111111-1111-1111-1111-111111111111" uname="xen01" type="normal"/>
      <node id="22222222-2222-2222-2222-222222222222" uname="xen02" type="normal"/>
    </nodes>
    <resources>
      <group id="grpHA">
        <primitive class="ocf" id="prmFS" provider="heartbeat" type="Filesystem">
          <instance_attributes id="prmFS-instance_attributes">
            <nvpair id="prmFS-instance_attributes-fstype" name="fstype" value="ext3"/>
            <nvpair id="prmFS-instance_attributes-run_fsck" name="run_fsck" value="force"/>
            <nvpair id="prmFS-instance_attributes-device" name="device" value="/dev/sda1"/>
            <nvpair id="prmFS-instance_attributes-directory" name="directory" value="/mnt"/>
          </instance_attributes>
          <operations>
            <op id="prmFS-start-0s" interval="0s" name="start" on-fail="restart" timeout="480s"/>
            <op id="prmFS-monitor-10s" interval="10s" name="monitor" on-fail="restart" timeout="60s"/>
          </operations>
        </primitive>
      </group>
      ...
    </resources>
    <rsc_location>
      <rsc_colocation id="colocation-01" rsc="grpHA" score="INFINITY" with-rsc="clnPingd"/>
      <rsc_order first="clnPingd" id="order-01" score="0" then="grpHA"/>
    </rsc_location>
    <constraints>
      <rsc_defaults>
        <meta_attributes id="rsc-options">
          <nvpair id="rsc-options-resource-stickiness" name="resource-stickiness" value="INFINITY"/>
          <nvpair id="rsc-options-migration-threshold" name="migration-threshold" value="1"/>
        </meta_attributes>
      </rsc_defaults>
    </constraints>
  </configuration>
</cib>
```

Heartbeat v2時代は、かなり頑張って cib.xml を作ったり編集したりしていました。
(これがイヤで、v1モードを使っている人も多いのでは...)

また、クラスタの状態を変更するためには、crm_resourceやcibadminなど複雑な管理コマンドを使いこなす必要がありました。

Pacemaker を使用したクラスタ環境では

crm シェルが使えます

- XMLの構文に悩まされることなく、リソースの設定ができます。
 - crmシェルがこっそり cib.xml を編集してくれます。
- 管理コマンドがうろ覚えでも、クラスタの管理ができます
 - crmシェルがこっそり管理コマンドを呼び出してくれます。
- 対話モードとバッチモードが使えます。
- TABキーでコマンドやオプションの補完ができます。
- クラスタが起動していなくても使えます。
 - クラスタの管理はできませんが、いろいろ遊べます。



2. crm シェルのインストール



Pacemaker がインストールされた環境であれば
crm シェルはすぐに使い始めることができます。

```
# which crm
/usr/sbin/crm

# file /usr/sbin/crm
/usr/sbin/crm: python script text executable

# rpm -qlp pacemaker-1.0.10-1.4.el6.x86_64.rpm | grep "/usr/sbin/crm$"
/usr/sbin/crm

# crm
crm(live)# help
```

This is the CRM command line interface program.

Available commands:

cib	manage shadow CIBs	
resource	resources management	リソースの管理を行います
node	nodes management	ノードの管理を行います
options	user preferences	
configure	CRM cluster configuration	クラスタの設定や管理を行います
ra	resource agents information center	RAの情報を表示します
status	show cluster status	
quit,bye,exit	exit the program	
help	show help	
end,cd,up	go back one level	

勉強会で使用する試験環境

Host OS(Domain0)

OS

- RHEL 5.6 x86_64 (xen)

Dell precision 370

- Intel(R) Pentium(R) 4 CPU 3.20GHz x2
- Memory 2GB

Guset OS(DomainU)

OS

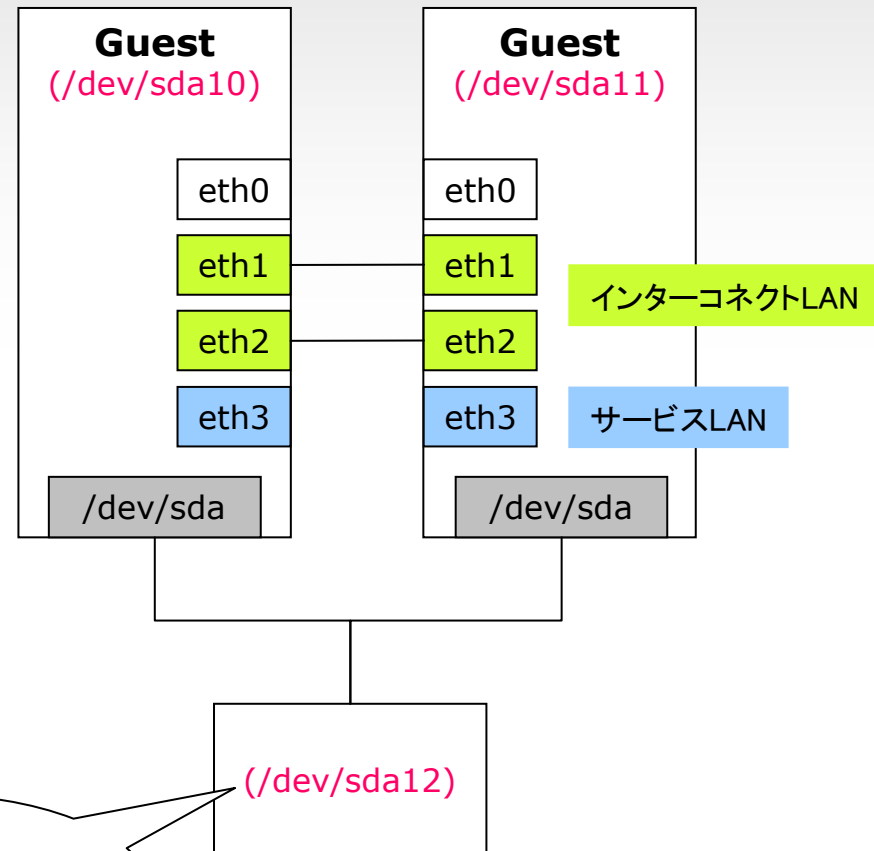
- RHEL 5.6 x86_64

Cluster software

- pacemaker 1.0.11
- cluster-glue 1.0.7
- resource-agents 1.0.4
- heartbeat 3.0.4
- pm-extras 1.02-1

CPU x1

Memory 512MB



共有ディスクのふり
(tap:aio)

勉強会で使用する試験環境

```
# cat /etc/ha.d/ha.cf
```

```
pacemaker on  
logfacility local1
```

```
debug 0  
udpport 694  
keepalive 2  
warntime 20  
deadtime 24  
initdead 48
```

```
bcast eth1  
bcast eth2
```

```
node xen01  
node xen02
```

```
watchdog /dev/watchdog  
respawn root /usr/lib64/heartbeat/ifcheckd
```



3. crm シェルでリソースを設定する



```
# vi sample.crm
```

```
property ¥
```

```
no-quorum-policy="ignore" ¥  
stonith-enabled="false" ¥  
startup-fencing="false" ¥
```

```
rsc_defaults ¥
```

```
resource-stickiness="INFINITY" ¥  
migration-threshold="1"
```

```
group grpHA prmFS prmIP
```

```
clone clnPingd prmPingd
```

```
primitive prmFS ocf:heartbeat:Filesystem ¥
```

```
params ¥
```

```
fstype="ext3" ¥
```

```
run_fsck="force" ¥
```

```
device="/dev/sda1" ¥
```

```
directory="/mnt" ¥
```

```
op start interval="0s" timeout="480s" on-fail="restart" ¥
```

```
op monitor interval="10s" timeout="60s" on-fail="restart" ¥
```

```
op stop interval="0s" timeout="60s" on-fail="block"
```

```
...
```

Linux-HA Japanのサイトでもよく見る
crm ファイル

でも、そもそも property って？
primitive ? group ?

```
crm(live)# configure  
crm(live)configure# help
```

設定関連はとりあえず
configure !

Commands for resources are:

リソースの設定

- **`primitive`**
- **`monitor`**
- **`group`**
- **`clone`**
- **`ms` / **`master` (master-slave)****

There are three types of constraints:

制約の設定

- **`location`**
- **`colocation`**
- **`order`**

Finally, there are the cluster properties, resource meta attributes defaults, and operations defaults.

クラスタの設定

- **`property`**
- **`rsc_defaults`**
- **`op_defaults`**

primitive

- 1コのリソースです。
例) 仮想IP、ファイルシステム、データベース
- RA(Resource Agent)を登録します。
 - RAに必要なパラメータ (params) を設定します。
 - RAに必要なオペレーション (op) を設定します。

設定例

```
primitive prmIP ocf:heartbeat:IPaddr2 ¥  
  params ¥  
    ip="192.168.201.129" ¥  
    nic="eth3" ¥  
    cidr_netmask="24" ¥  
  op start interval="0s" timeout="60s" on-fail="restart" ¥  
  op monitor interval="10s" timeout="60s" on-fail="restart" ¥  
  op stop interval="0s" timeout="60s" on-fail="block"
```

でも、RAに必要な
パラメータとかオペレーションとか
わからない

```
# crm
```

RAの情報表示

```
crm(live)# ra
```

```
crm(live)ra# list
```

```
usage: list <class> [<provider>]
```

さらに help

```
crm(live)ra# help list
```

```
crm(live)ra# list ocf
```

```
crm(live)ra# info vmware
```

Manages VMWare Server 2.0 virtual machines (ocf:heartbeat:vmware)

OCF compliant script to control vmware server 2.0 virtual machines.

Parameters (* denotes required, [] the default):

vmxpath* (string): VMX file path
VMX configuration file path

vimshbin (string, [/usr/bin/vmware-vim-cmd]): vmware-vim-cmd path
vmware-vim-cmd executable path

Operations' defaults (advisory minimum):

start	timeout=600
stop	timeout=600
monitor	interval=300 timeout=30

group

- 1コ以上のリソースを登録します。
- start/stop の順序が保証されます。
- group まるごと F/O できます。

設定例

group sister kana kayo

グループの名前

グループの登録するリソースの名前

グループの登録するリソースの名前

```
crm(live)configure# property ¥  
> no-quorum-policy="ignore" ¥  
> stonith-enabled="false" ¥  
> startup-fencing="false"
```

```
crm(live)configure# primitive ¥  
> prmIP ocf:heartbeat:IPaddr2 ¥  
> params ¥  
> ip="192.168.201.129" ¥  
> nic="eth3" ¥  
> cidr_netmask="24" ¥  
> op start interval="0s" timeout="60s" on-fail="restart" ¥  
> op monitor interval="10s" timeout="60s" on-fail="restart" ¥  
> op stop interval="0s" timeout="60s" on-fail="block"
```

```
crm(live)configure# show
```

```
crm(live)configure# commit
```

```
crm_verify[7851]: 2011/06/03_11:41:44 WARN: unpack_nodes: Blind faith: not fencing unseen nodes
```

```
crm(live)configure# cd
```

```
crm(live)# resource
```

```
crm(live)resource# show
```

```
prmIP (ocf::heartbeat:IPaddr2) Stopped
```

```
crm(live)resource# start prmIP
```

STONITHを設定していないと警告が
でるけど、とりあえず無視



```
crm(live)resource# cd

crm(live)# configure
crm(live)configure# primitive ¥
> kana ocf:pacemaker:Dummy ¥
> op start interval="0s" timeout="100s" on-fail="restart" ¥
> op monitor interval="10s" timeout="100s" on-fail="restart" ¥
> op stop interval="0s" timeout="100s" on-fail="block"

crm(live)configure# commit
crm_verify[2699]: 2011/06/06_14:13:37 WARN: unpack_nodes: Blind faith: not fencing unseen nodes

crm(live)configure# cd
crm(live)# resource show
prmIP (ocf::heartbeat:IPaddr2) Started
kana (ocf::pacemaker:Dummy) Started
```



```
crm(live)# configure
crm(live)configure# edit

node $id="11111111-1111-1111-1111-111111111111" xen01
node $id="22222222-2222-2222-2222-222222222222" xen02
primitive kana ocf:pacemaker:Dummy ¥
  op start interval="0s" timeout="100s" on-fail="restart" ¥
  op monitor interval="10s" timeout="100s" on-fail="restart" ¥
  op stop interval="0s" timeout="100s" on-fail="block"
primitive kayo ocf:pacemaker:Dummy ¥
  op start interval="0s" timeout="100s" on-fail="restart" ¥
  op monitor interval="10s" timeout="100s" on-fail="restart" ¥
  op stop interval="0s" timeout="100s" on-fail="block"
primitive prmIP ocf:heartbeat:IPaddr2 ¥
  params ip="192.168.201.129" nic="eth3" cidr_netmask="24" ¥
  op start interval="0s" timeout="60s" on-fail="restart" ¥
  op monitor interval="10s" timeout="60s" on-fail="restart" ¥
  op stop interval="0s" timeout="60s" on-fail="block"
property $id="cib-bootstrap-options" ¥
  dc-version="1.0.11-db98485d06ed stable-1.0 tip" ¥
  cluster-infrastructure="Heartbeat" ¥
  no-quorum-policy="ignore" ¥
  stonith-enabled="false" ¥
  startup-fencing="false"
```

コピー！

!wq

edit の結果を書き込みます。
(今回の例では vi ですが好みのエディタに変更することも可能です)

edit 操作は予想外の変更もクラスタに反映されてしまうことがあるため
注意して実行してください。
編集には後述の rename, delete コマンドが推奨されています。

```
crm(live)configure# cd
crm(live)# resource show
prmIP (ocf::heartbeat:IPaddr2) Started
kana (ocf::pacemaker:Dummy) Started
kayo (ocf::pacemaker:Dummy) Started

crm(live)# resource stop kayo
crm(live)# resource stop kana
crm(live)# resource show
prmIP (ocf::heartbeat:IPaddr2) Started
kana (ocf::pacemaker:Dummy) Stopped
kayo (ocf::pacemaker:Dummy) Stopped

crm(live)# configure
crm(live)configure# group sister kana kayo
```



```
crm(live)configure# show
node $id="11111111-1111-1111-1111-111111111111" xen01
node $id="22222222-2222-2222-2222-222222222222" xen02
primitive kana ocf:pacemaker:Dummy ¥
    op start interval="0s" timeout="100s" on-fail="restart" ¥
    op monitor interval="10s" timeout="100s" on-fail="restart" ¥
    op stop interval="0s" timeout="100s" on-fail="block" ¥
    meta target-role="Stopped"
primitive kayo ocf:pacemaker:Dummy ¥
    op start interval="0s" timeout="100s" on-fail="restart" ¥
    op monitor interval="10s" timeout="100s" on-fail="restart" ¥
    op stop interval="0s" timeout="100s" on-fail="block" ¥
    meta target-role="Stopped"
primitive prmIP ocf:heartbeat:IPaddr2 ¥
    params ip="192.168.201.129" nic="eth3" cidr_netmask="24" ¥
    op start interval="0s" timeout="60s" on-fail="restart" ¥
    op monitor interval="10s" timeout="60s" on-fail="restart" ¥
    op stop interval="0s" timeout="60s" on-fail="block"
group sister kana kayo
property $id="cib-bootstrap-options" ¥
    dc-version="1.0.11-db98485d06ed stable-1.0 tip" ¥
    cluster-infrastructure="Heartbeat" ¥
    no-quorum-policy="ignore" ¥
    stonith-enabled="false" ¥
    startup-fencing="false"
crm(live)configure# commit
crm_verify[2899]: 2011/06/06_14:21:11 WARN: unpack_nodes: Blind faith: not fencing unseen nodes
```



```
crm(live)configure# cd
crm(live)# resource start sister
crm(live)# show
ERROR: syntax: show
crm(live)# resource show
prmIP (ocf::heartbeat:IPaddr2) Started
Resource Group: sister
  kana      (ocf::pacemaker:Dummy) Started
  kayo      (ocf::pacemaker:Dummy) Started
```



```
crm(live)configure# cd
crm(live)# resource stop prmIP
crm(live)# configure rename
usage: rename <old_id> <new_id>
crm(live)# configure rename prmIP monja
crm_verify[3041]: 2011/06/06_14:24:09 WARN: unpack_nodes: Blind faith: not fencing unseen nodes

crm(live)# resource start monja
crm(live)# resource show
Resource Group: sister
    kana      (ocf::pacemaker:Dummy) Started
    kayo      (ocf::pacemaker:Dummy) Started
monja (ocf::heartbeat:IPaddr2) Started
```

configureレベルの外から
configureの変更を実施すると
「自動的に commit」されて
しまうので注意してください！



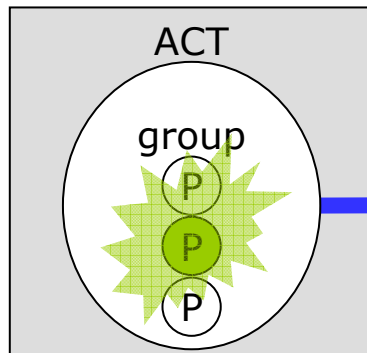
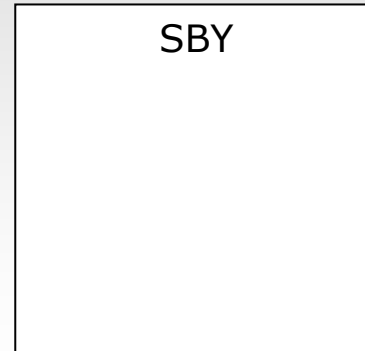
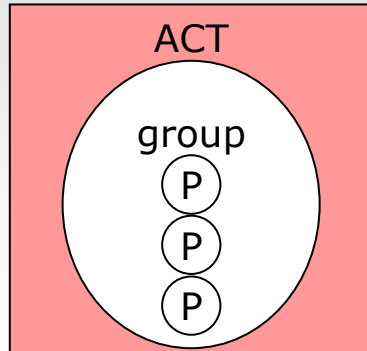

```
crm(live)# resource stop monja
crm(live)# configure delete monja
crm_verify[3897]: 2011/06/06_14:28:09 WARN: unpack_nodes: Blind faith: not fencing unseen nodes
crm(live)# configure show
node $id="11111111-1111-1111-1111-111111111111" xen01
node $id="22222222-2222-2222-2222-222222222222" xen02
primitive kana ocf:pacemaker:Dummy ¥
    op start interval="0s" timeout="100s" on-fail="restart" ¥
    op monitor interval="10s" timeout="100s" on-fail="restart" ¥
    op stop interval="0s" timeout="100s" on-fail="block"
primitive kayo ocf:pacemaker:Dummy ¥
    op start interval="0s" timeout="100s" on-fail="restart" ¥
    op monitor interval="10s" timeout="100s" on-fail="restart" ¥
    op stop interval="0s" timeout="100s" on-fail="block"
group sister kana kayo ¥
    meta target-role="Started"
property $id="cib-bootstrap-options" ¥
    dc-version="1.0.11-db98485d06ed stable-1.0 tip" ¥
    cluster-infrastructure="Heartbeat" ¥
    no-quorum-policy="ignore" ¥
    stonith-enabled="false" ¥
    startup-fencing="false"
```

monja の設定が削除されています

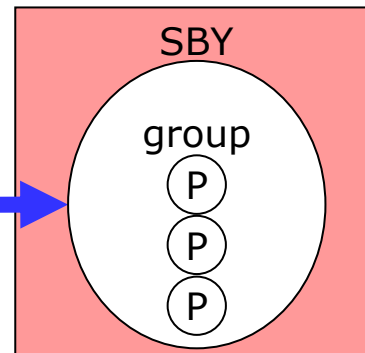


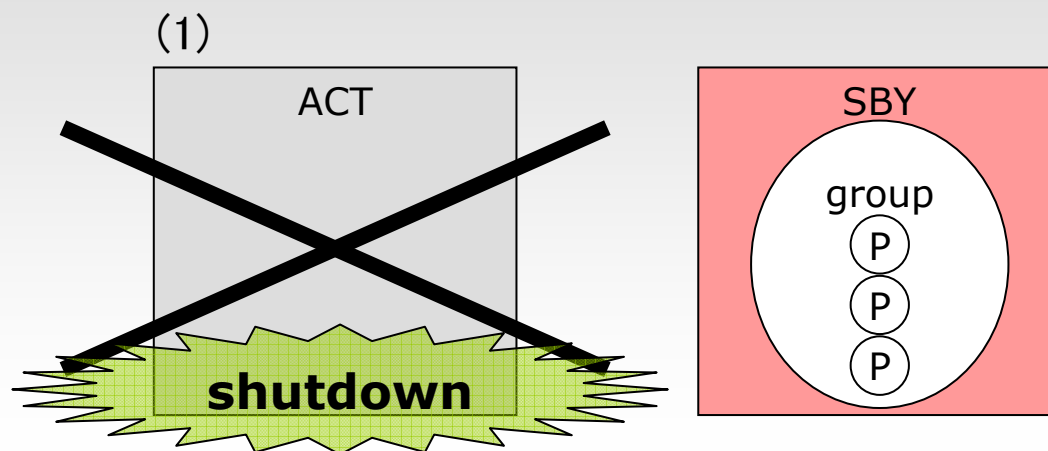
```
crm(live)# resource stop sister  
crm(live)# configure  
crm(live)configure# erase  
crm(live)configure# commit  
crm(live)configure# show  
node $id="11111111-1111-1111-1111-111111111111" xen01  
node $id="22222222-2222-2222-2222-222222222222" xen02
```

クラスタの設定が全て削除されます。
delete と erase は注意して
使い分けてください。



F/O



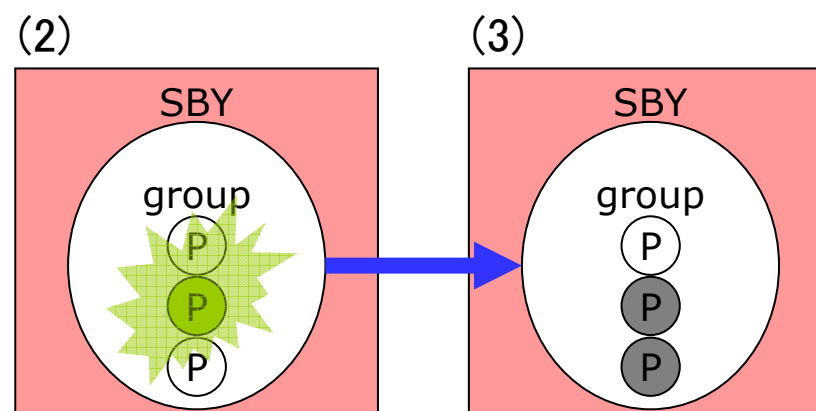


- (1) 故障したACTノードを停止
- (2) SBYで2番目のprimitiveが故障
- (3) SBYで2,3番目のprimitiveが停止

この場合、1番目のprimitiveは起動継続する

→ **仕様**

ホントは1番目も停止してほしいところですが、あんどりゅーくんととの闘いに敗れました



clone

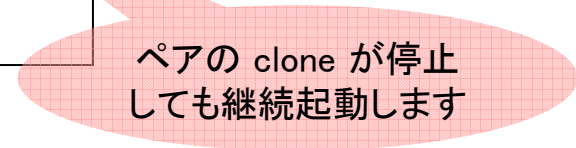
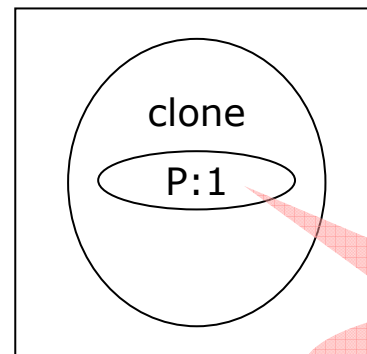
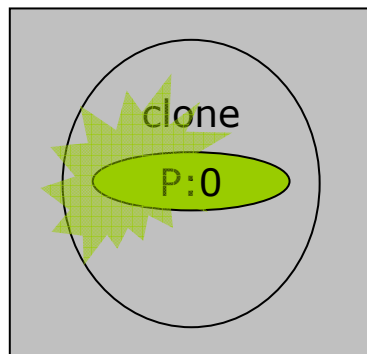
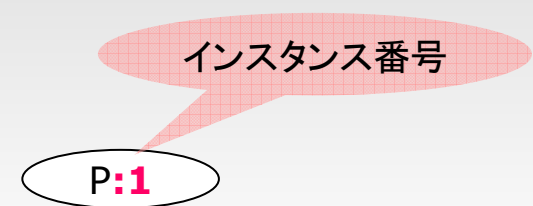
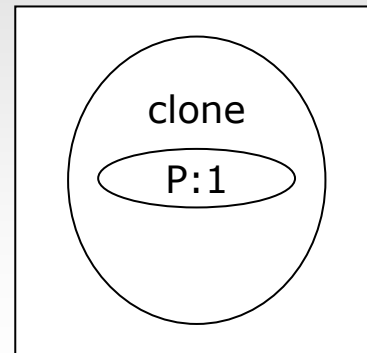
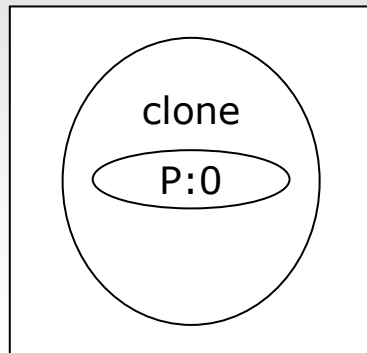
- 1コの primitive または group を登録します。

設定例

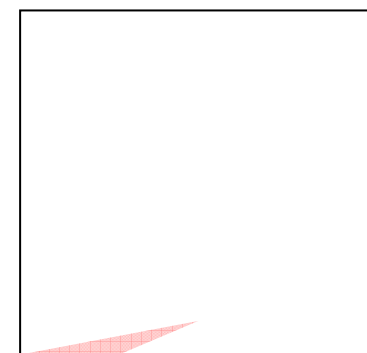
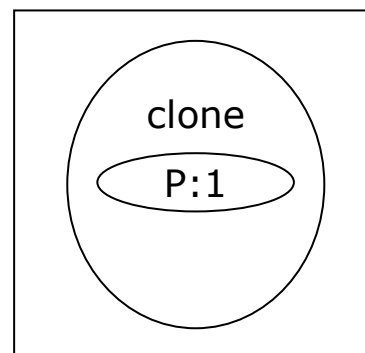
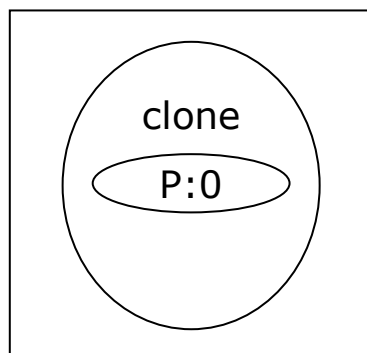
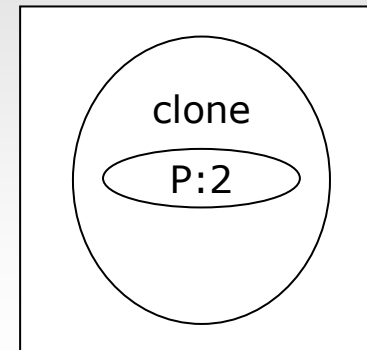
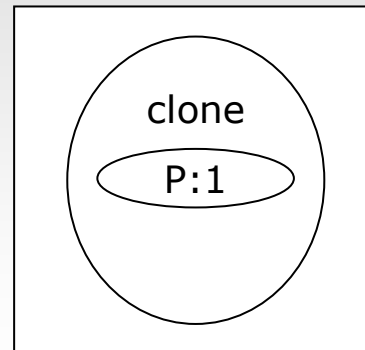
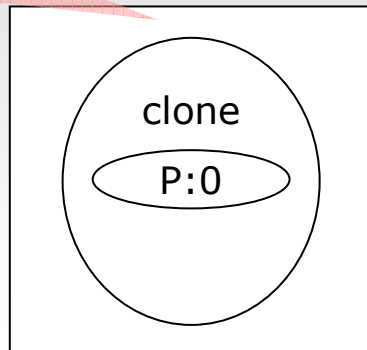
```
clone clnPingd prmPingd ¥  
clone-max="2" clone-node-max="1"
```

clone を配置するノード数
※ デフォルト値: クラスタ内のノード数

1ノードあたりの clone の数
※ デフォルト値: 1



3ノード以上でもOK



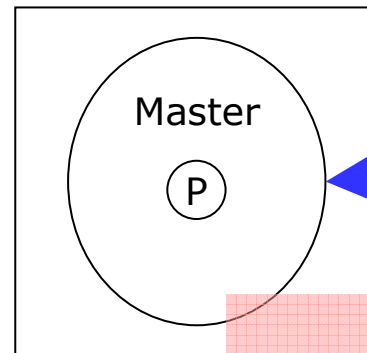
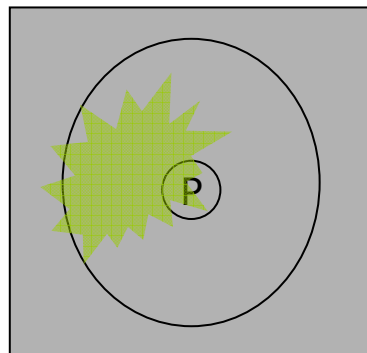
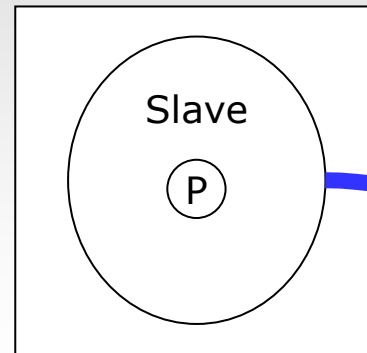
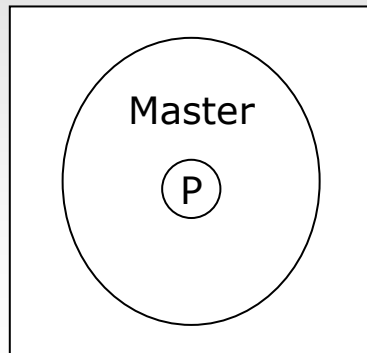
全ノードに配置しなくてもOK

Master/Slave

- 1コの primitive または group を登録します。

設定例

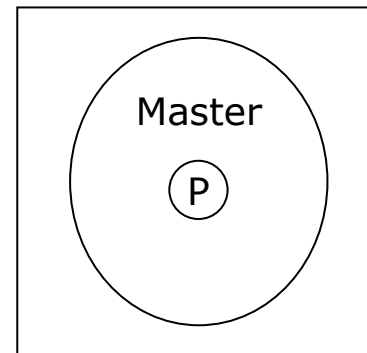
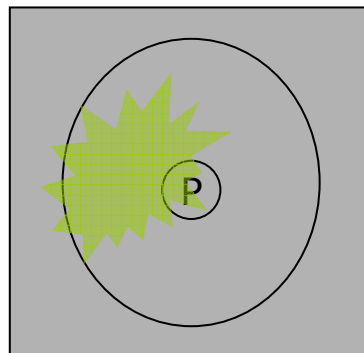
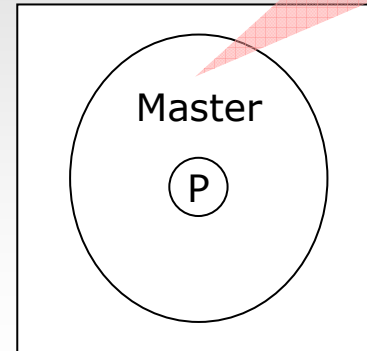
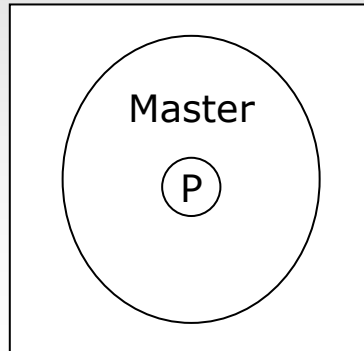
```
ms msDrbd grpDrbd ¥  
  meta master-max="1" ¥  
    master-node-max="1" ¥  
    clone-max="2" ¥  
    clone-node-max="1" ¥  
    notify="true" ¥  
    globally-unique="false"
```

promote

Master → Slave ... promote
Slave → Master ... demote

両ノードとも Master でも OK



制約の設定

location
colocation
order

配置制約
同居制約
順序制約

location (配置制約)

crm(live)configure# location

usage: location <id> <rsc> {node_pref|rules}

node_pref :: <score>: <node>

rules ::

rule [id_spec] [\$role=<role>] <score>: <expression>

[rule [id_spec] [\$role=<role>] <score>: <expression> ...]

id_spec :: \$id=<id> | \$id-ref=<id>

score :: <number> | <attribute> | [-]inf

expression :: <simple_exp> [bool_op <simple_exp> ...]

bool_op :: or | and

simple_exp :: <attribute> [type:]<binary_op> <value>

| <unary_op> <attribute>

| date <date_expr>

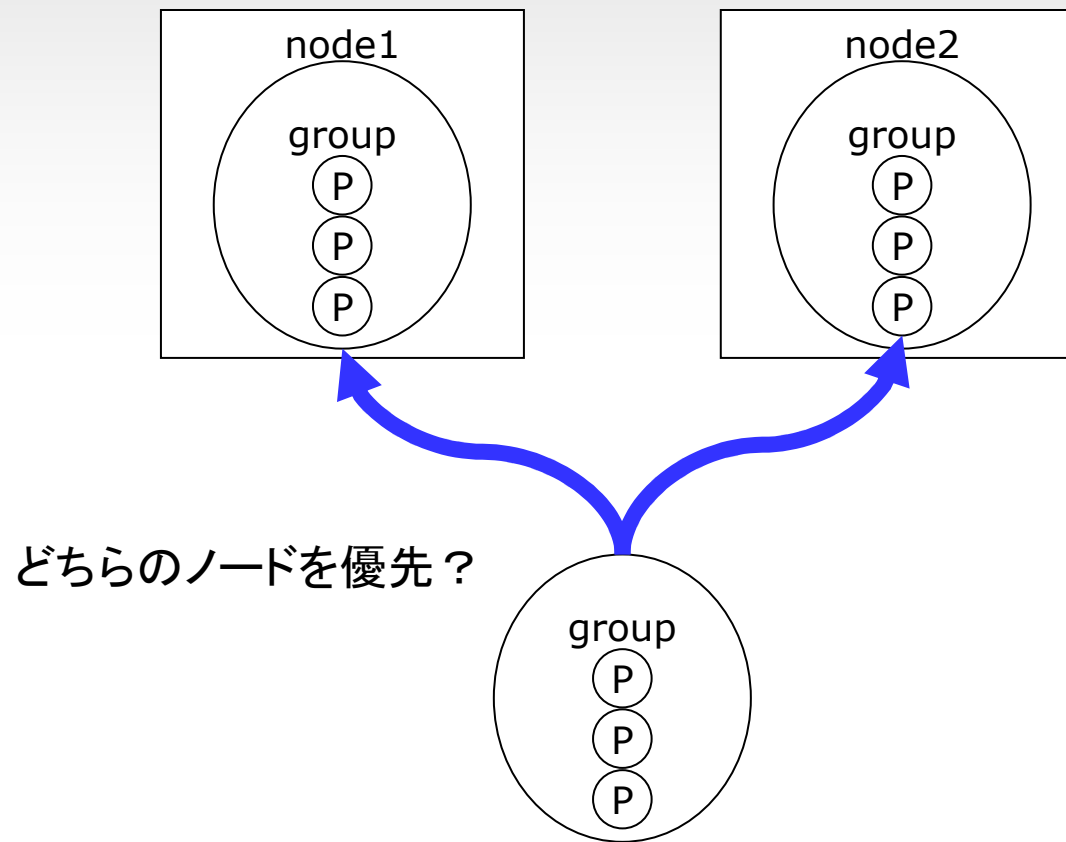
type :: string | version | number

binary_op :: lt | gt | lte | gte | eq | ne

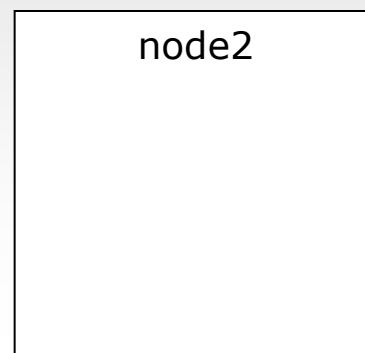
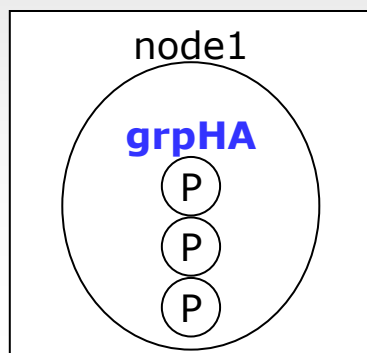
unary_op :: defined | not_defined



location (配置制約)



location (配置制約)



location の名前

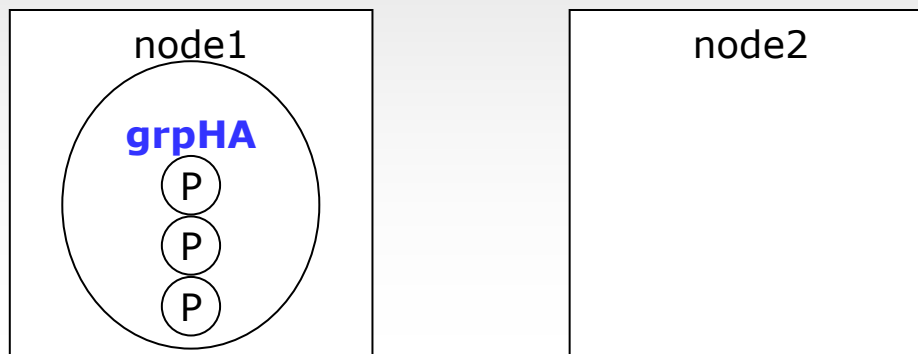
配置制約を
設定したいリソース名

```
location location-01 grpHA ¥  
rule 200: #uname eq node1 ¥  
rule 100: #uname eq node2
```

スコア値

ノード名

location (配置制約)



location location-02 **grpHA** ¥
rule **-INFINITY**: #uname eq **node2**

node2で起動させたくない場合は
-INFINITYを設定すればよい

colocation（同居制約）

crm(live)configure# colocation

usage: colocation <id> <score>: <rsc>[:<role>] <rsc>[:<role>]

colocation の名前

制約に関連する
リソース名

制約対象となる
リソース名

colocation nakayoshi-colocation **INFINITY**: kana kayo

kana は kayo が起動しているノードに同居できる

colocation nakawarui-colocation **-INFINITY**: kana kayo

kana は kayo が起動しているノードに同居できない

order（順序制約）

crm(live)configure# order

usage: order <id> score-type: <first-rsc>[:<action>] <then-rsc>[:<action>]
[symmetrical=<bool>]

order の名前

起動する順番にリソース名を並べる

order order-01 0: kana kayo

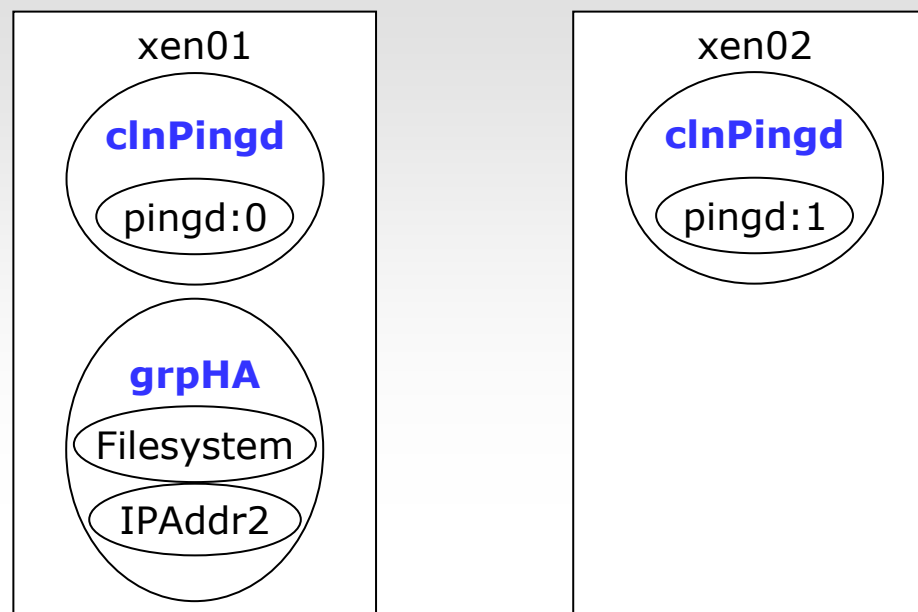
kana が起動した後に、kayo が起動する

停止順序は、起動順序の逆となります。

ただし、symmetrical=falseを設定すると、起動順序と停止順序は同じになります。

symmetricalのデフォルト値はtrueです。

location, colocation, order の設定例



clnPingd が起動した後に、grpHA を起動させたい → **order**

grpHA は clnPingd と同じノードで起動させたい → **colocation**

grpHA は、xen01 で優先的に起動させたい → **location**

clnPingdの属性値が設定されていない

または clnPindの属性値が100より小さい場合

grpHAはそのノードで起動してはならない → **location**

設定例

<省略>

group grpHA prmFS prmIP

clone clnPingd prmPingd meta clone-max="2" clone-node-max="1"

primitive prmFS ocf:heartbeat:Filesystem ¥

<省略>

primitive prmIP ocf:heartbeat:IPaddr2 ¥

<省略>

primitive prmPingd ocf:pacemaker:pingd ¥
params ¥

name="default_ping_set" ¥

host_list="192.168.201.81" ¥

multiplier="100" ¥

dampen="1" ¥

op start interval="0s" timeout="90s" on-fail="restart" ¥

op monitor interval="10s" timeout="60s" on-fail="restart" ¥

op stop interval="0s" timeout="100s" on-fail="ignore"

location location-01 grpHA ¥

rule 200: #uname eq xen01 ¥

rule 100: #uname eq xen02 ¥

rule -INFINITY: not_defined default_ping_set or default_ping_set lt 100

colocation colocation-01 INFINITY: grpHA clnPingd

order order-01 0: clnPingd grpHA

clnPingdの属性値が設定されていない
または
clnPingdの属性値が100より小さい
場合は起動しない(-INFINITY)

property

クラスタ全体の設定

```
crm(live)configure# property  
usage: property [$id=<set_id>] <option>=<value>
```

res_defaults

リソース全体の設定

```
crm(live)configure# rsc_defaults  
usage: rsc_defaults [$id=<set_id>] <option>=<value>
```

op_defaults

オペレーション全体の設定

```
crm(live)configure# op_defaults  
usage: op_defaults [$id=<set_id>] <option>=<value>
```

設定例

Cluster Option

property ¥

no-quorum-policy="ignore" ¥

stonith-enabled="false" ¥

startup-fencing="false" ¥

STONITHを有効にする場合は
stonith-enabled="true"

stonith-enabled="true" の場合
op stop on-fail="fence"

stonith-enabled="false" の場合
op stop on-fail="block"

Resource Defaults

rsc_defaults ¥

resource-stickiness="INFINITY" ¥

migration-threshold="1"

Linux-HA Japan
月刊あんどりゅーくん(6月号)の
「知恵袋」を参照

時間の都合上、説明を省略します。

バッチモード用のファイルを作成する

dummy.crm
(拡張子は .crm でなくてもよい)

```
### Cluster Option ###
property no-quorum-policy="ignore" ¥
        stonith-enabled="false" ¥
        startup-fencing="false" ¥

### Resource Defaults ###
rsc_defaults resource-stickiness="INFINITY" ¥
        migration-threshold="1"

### Primitive Configuration ###
primitive prmDummy ocf:pacemaker:Dummy ¥
        op start interval="0s" timeout="100s" on-fail="restart" ¥
        op monitor interval="10s" timeout="100s" on-fail="restart" ¥
        op stop interval="0s" timeout="100s" on-fail="block"

location location-01 prmDummy ¥
        rule 200: #uname eq xen01 ¥
        rule 100: #uname eq xen02
```

バッチモードでリソースを設定する

ファイル名

```
[root@xen01 hastudy]# crm configure load update dummy.crm
crm_verify[6592]: 2011/06/01_16:06:12 WARN: unpack_nodes: Blind faith: not fencing unseen nodes
```

```
[root@xen01 hastudy]# crm_mon -1 -A
```

```
=====
```

```
Last updated: Wed Jun  1 16:06:50 2011
```

```
Stack: Heartbeat
```

```
Current DC: xen02 (22222222-2222-2222-2222-222222222222) - partition with quorum
```

```
Version: 1.0.11-db98485d06ed stable-1.0 tip
```

```
2 Nodes configured, unknown expected votes
```

```
1 Resources configured.
```

```
=====
```

```
Online: [ xen01 xen02 ]
```

```
prnDummy      (ocf::pacemaker:Dummy): Started xen01
```

Node Attributes:

* Node xen01:

```
+ xen02-eth1      : up
```

```
+ xen02-eth2      : up
```

* Node xen02:

```
+ xen01-eth1      : up
```

```
+ xen01-eth2      : up
```

STONITHを設定していないと警告が
でるけど、とりあえず無視

ifcheckd をインストールして、ha.cfに
設定しておく、インターコネクトLAN
の情報も表示できます(crm_mon -A)

4. crm シェルでクラスタを管理する



(1) リソースの起動停止

```
# crm
crm(live)# resource
crm(live)resource# show
Resource Group: grpHA
  prmFS      (ocf::heartbeat:Filesystem) Started
  prmIP      (ocf::heartbeat:IPaddr2) Started
Clone Set: clnPingd
  Started: [ xen01 xen02 ]
```

```
crm(live)resource# start
usage: start <rsc>
crm(live)resource# stop
usage: stop <rsc>
```

```
crm(live)resource# stop prmIP
crm(live)resource# start prmIP
crm(live)resource# stop prmFS
crm(live)resource# start prmFS
crm(live)resource# stop clnPingd
crm(live)resource# start clnPingd
```

clone はインスタンス番号を
指定できません

(2) リソースの移動

```
# crm
crm(live)# resource
crm(live)resource# show
Resource Group: grpHA
    prmFS      (ocf::heartbeat:Filesystem) Started
    prmIP      (ocf::heartbeat:IPaddr2) Started
Clone Set: clnPingd
    Started: [ xen01 xen02 ]
crm(live)resource# move
usage: migrate <rsc> [<node>] [<lifetime>] [force]

crm(live)resource# move prmIP xen02 force

crm(live)resource# unmove
usage: unmigrate <rsc>

crm(live)resource# unmove prmIP
```

「move force」は移動元のノードにスコア値(**-INFINITY**)を追加するため、実行時に警告が表示されます。
「move force」でリソースを移動させた後は、移動元のスコアを正常に戻すため、必ず「**unmove**」を実行してください。

(3) 故障からの復旧

```
# ip addr del 192.168.201.129/24 dev eth3
```

```
# crm_mon -i1 -f
```

```
Online: [ xen01 xen02 ]
```

```
Resource Group: grpHA
```

```
  prmFS      (ocf::heartbeat:Filesystem):    Started xen02
```

```
  prmIP      (ocf::heartbeat:IPaddr2):        Started xen02
```

```
Clone Set: clnPingd
```

```
Started: [ xen01 xen02 ]
```

```
Migration summary:
```

```
* Node xen02:
```

```
* Node xen01:
```

```
prmIP: migration-threshold=1 fail-count=1
```

```
Failed actions:
```

```
  prmIP_monitor_10000 (node=xen01, call=41, rc=7, status=complete): not running
```



```
# crm
crm(live)# resource
crm(live)resource# failcount
usage:
    failcount <rsc> set <node> <value>
    failcount <rsc> delete <node>
    failcount <rsc> show <node>
crm(live)resource# show
Resource Group: grpHA
    prmFS      (ocf::heartbeat:Filesystem) Started
    prmIP      (ocf::heartbeat:IPaddr2) Started
Clone Set: clnPingd
    Started: [ xen01 xen02 ]

crm(live)resource# failcount prmIP show xen01
scope=status name=fail-count-prmIP value=1

crm(live)resource# failcount prmIP delete xen01

crm(live)resource# failcount prmIP show xen01
scope=status name=fail-count-prmIP value=0
crm(live)resource# quit
bye
```



バッチモード

```
# crm resource failcount prmIP show xen01  
scope=status name=fail-count-prmIP value=0
```

crmシェルがこっそり
呼び出しているコマンドを表示

```
# crm -R  
crm(live)# resource failcount prmIP show xen01
```

```
.INP: resource failcount prmIP show xen01  
.EXT crm_failcount -r 'prmIP' -N 'xen01' -G
```

```
scope=status name=fail-count-prmIP value=0
```



pingd によるサービスLAN切断検知

```
# iptables -A INPUT -i eth3 -j DROP; iptables -A OUTPUT -o eth3 -j DROP
```

```
# crm_mon -i1 -Af
```

Online: [xen01 xen02]

Resource Group: grpHA

prnFS (ocf::heartbeat:Filesystem): Started xen02

prnIP (ocf::heartbeat:IPaddr2): Started xen02

Clone Set: clnPingd

Started: [xen01 xen02]

Node Attributes:

* Node xen01:

+ **default_ping_set**

: 0

: **Connectivity is lost**

+ xen02-eth1

: up

+ xen02-eth2

: up

* Node xen02:

+ default_ping_set

: 100

+ xen01-eth1

: up

+ xen01-eth2

: up

Migration summary:

* Node xen02:

* Node xen01:

fail-count は増加しない

location 設定に従ってF/O

スコア値が100 → 0



```
# iptables -F
```

```
# crm_mon -i1 -Af
```

```
Online: [ xen01 xen02 ]
```

```
Resource Group: grpHA
```

```
  prmFS      (ocf::heartbeat:Filesystem):   Started xen02
```

```
  prmIP      (ocf::heartbeat:IPaddr2):       Started xen02
```

```
Clone Set: clnPingd
```

```
Started: [ xen01 xen02 ]
```

```
Node Attributes:
```

```
* Node xen01:
```

```
  + default_ping_set : 100
```

```
  + xen02-eth1         : up
```

```
  + xen02-eth2         : up
```

```
* Node xen02:
```

```
  + default_ping_set   : 100
```

```
  + xen01-eth1         : up
```

```
  + xen01-eth2         : up
```

```
Migration summary:
```

```
* Node xen02:
```

```
* Node xen01:
```

スコア値が0 → 100

(4) ノードのスタンバイ化/オンライン化

```
crm(live)# node
crm(live)node# help standby
```

Set a node to standby status. The node parameter defaults to the node where the command is run. Additionally, you may specify a lifetime for the standby---if set to `reboot`, the node will be back online once it reboots. `forever` will keep the node in standby after reboot.

Usage:

```
.....
    standby [<node>] [<lifetime>]

    lifetime :: reboot | forever
.....
```

ノード名を指定しなかった場合は、
crmシェルを実行中のノードが
スタンバイ化

おわり

