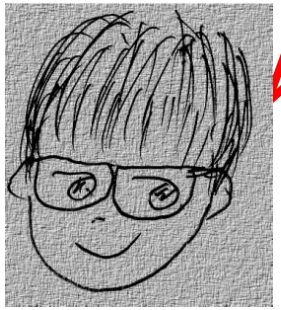
HAクラスタサポートの日々
～Pacemaker導入･運用の勘所～

2012年8月4日　OSC2012 Kansai/Kyoto

Linux-HA Japan
赤松　洋

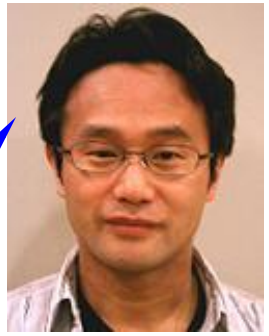
ある日…



三井さん

赤松君、最近 Pacemaker の問い合わせが多いね

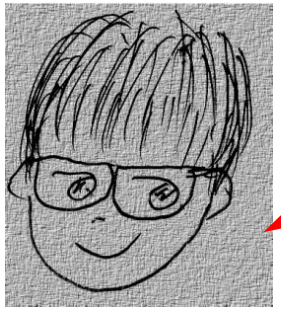
今週5件ですね、今年度だけですでに50件超えてます



赤松

わが社だけでこれだけあるということは、コミュニティでも困ってる人が多いんじゃないかなぁ？

ある日…

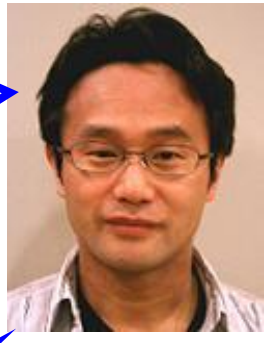


三井さん

まぁ、そうかもしれませんね

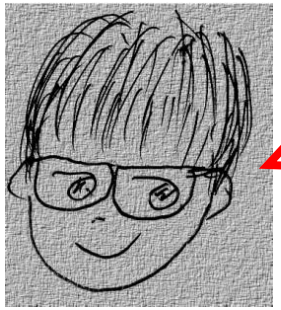
よし決めた！保守運用ノウハウを
OSC で発表しよう！

・・・あの
半年前も、OSC東京で
しましたよ？



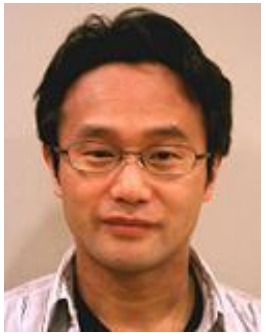
赤松

ある日…



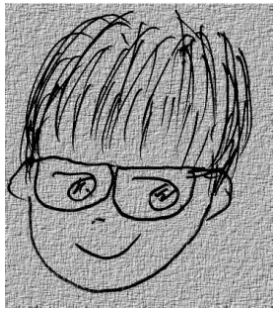
三井さん

赤松くん！
あとよろしく！

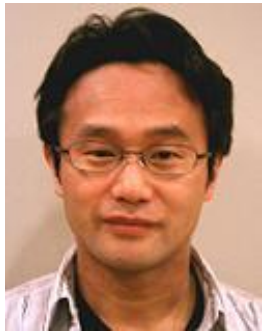


赤松

登場人物



三井さん
Linux-HA の重鎮の一人
何でも答えてくれます！



赤松
今回の演者、日々汗を流して
サポート業務に励んでいます

更に…

登場人物

かなさんとかよさんにも
登場して頂きます

本日のお話

① フェイルオーバーに関する運用

② Pacemaker の自動起動・停止・リストア

③ stonith について

ちなみに

- インストール・環境構築の話はありません
- GUI・Corosync の話もありません
- 仮想化の話は少し触れるかもしれません
- DRBD の話もありません

ちなみに

- 環境は Pacemaker ＋ Heartbeat（1.0.12）に特化しています
- 個人的な見解に沿っている所もあります
- ご不明な点は後ほどブースや、メール等でお問い合わせ下さい

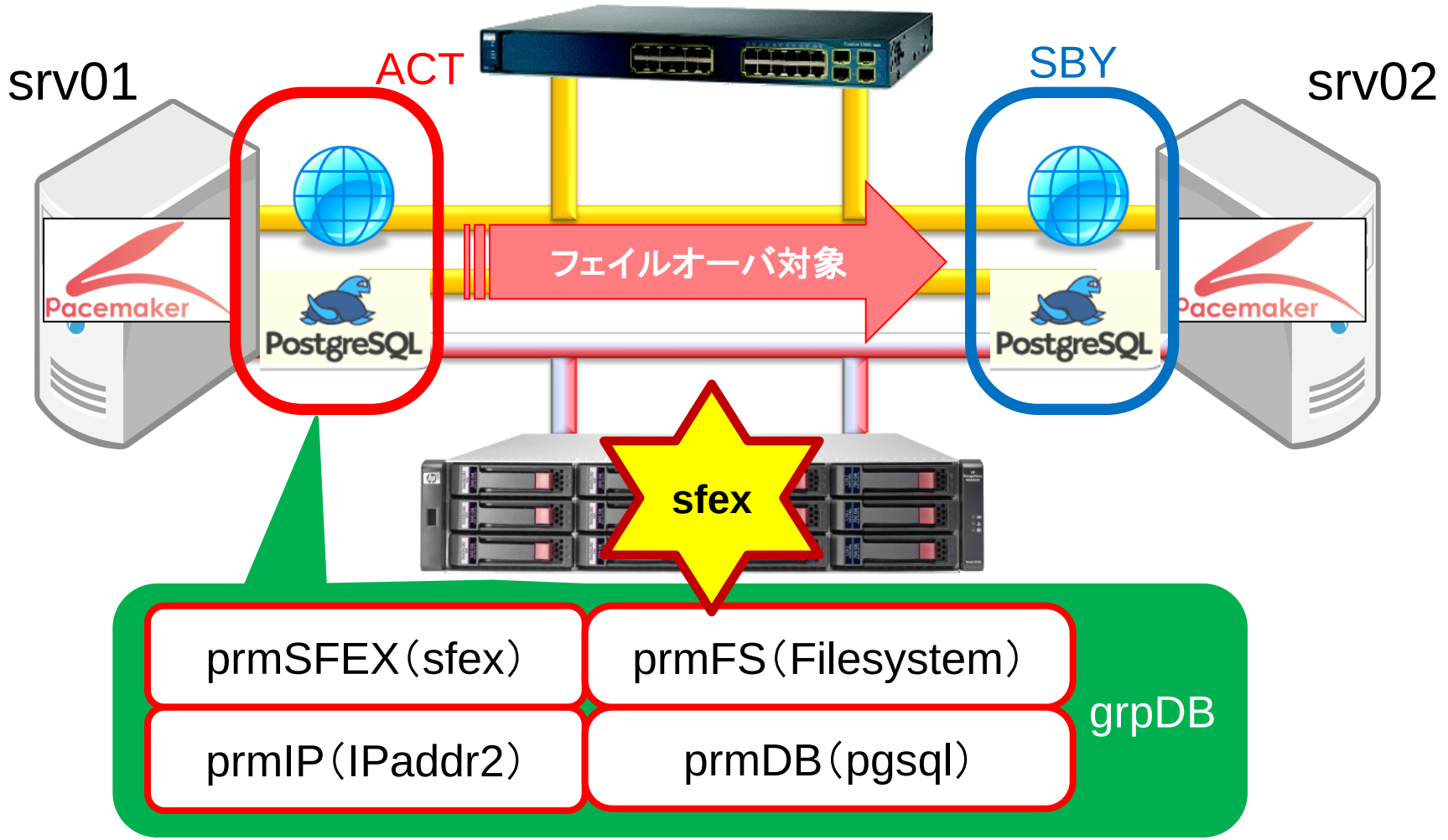
本日のお話

① フェイルオーバーに関する運用

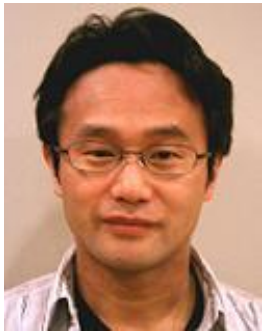
② Pacemaker の自動起動・停止・リストア

③ 最後に stonith について

某社のシステム構成


srv01
ACT
SBY
srv02
Pacemaker
PostgreSQL
フェイルオーバ対象
PostgreSQL
Pacemaker
sfex
prmSFEX（sfex）
prmFS（Filesystem）
prmIP（IPaddr2）
prmDB（pgsql）
grpDB

大変です！



どうされましたか？

リソースがフェイルオーバーしてます！

何をしたらよいでしょうか！？

- まずは現状認識
  - srv01 と srv02 の状況
  - srv02 ではリソースが正常稼働しているか

- 具体的には crm_mon コマンドを実行

```
[srv02 ]# crm_mon△-fA△-1
```

    - srv01 が稼働しているか
    - srv02 で正常稼働中か
    - ha-log, messages ファイルなどを保存・分析
  - その結果から…

現状の確認

サーバの電源入ってない

サーバの電源ON

Pacemaker が起動していない

ハードやログの確認
- コミュニティへ質問等

リソース異常が出力

Pcemaker 起動

Standby実施

異常の調査・復旧
- NW経路
- ディスク異常
- リソース異常

- サーバの電源が入っていない
  - 各ケーブルの結線の確認
  - 電源押下、ディスクの状況確認
  - ログファイルを分析用に保存
    - 調査・コミュニティへ質問

- Pacemaker が起動していない時
  - 上記同様の確認
  - ログファイルを分析用に保存
    - 調査・コミュニティへ質問
  - Pacemaker の起動

リソース異常が出力

- srv01を Standby 化します

```
[srv02 ]# crm△-R△node△standby△srv01
```

- 再度、現状認識

```
[srv02 ]# crm_mon△-fA△-1
```

- その結果から、（主に）下記の異常が判定
  - NW経路監視に異常
  - ディスク監視に異常
  - リソースに異常

リソース異常が出力

- srv01を Standby 化します

```
[srv02 ]# crm△-R△node△standby△srv01
```

- 再度、現状認識

```
[srv02 ]# crm_r
```

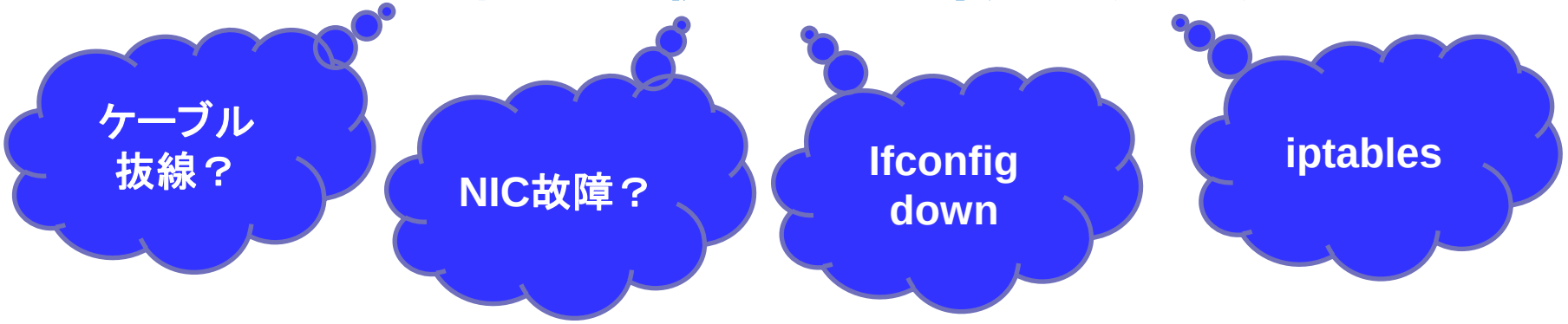
クイズ！
-R をつけると、どんな効果があるでしょうか？

- その結果から、(
  - NW経路監視に異常
  - ディスク監視に異常
  - リソースに異常

NW経路監視に異常

```
[srv01 ]# crm_mon△-fA△-1
・・・
* Node srv01:
    + default_ping_set : 0 : Connectivity is lost
```

□ **srv01 から経路監視先への導通で異常発生！**



□ **問題が解決したらクラスタメンバに復帰**

```
[srv01 ]# crm△-R△node△online△srv01
```

ディスク経路監視に異常

```
[srv01 ]# crm_mon△-fA△-1
・・・
 + diskcheck_status : ERROR
```

□ サーバと共有ディスク間の導通で異常発生！

□ 問題が解決したらクラスタメンバに復帰

```
[srv01 ]# crm△-R△node△online△srv01
```

リソース異常

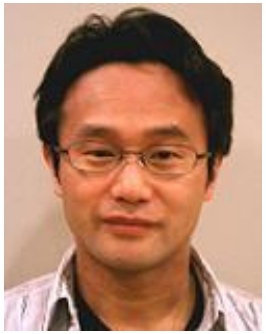
```
[srv01 ]# crm_mon△-fA△-1
・・・
* Node srv01:
   prmDB: migration-threshold=1 fail-count=1

Failed actions:
  prmDB_monitor_10000 ¥
    (node=srv01, call=XXX, rc=-2, status=Timed Out): ¥
     unknown exec error
```

□ **異常情報をクリアして、クラスタメンバに復帰！**

```
[srv01 ]# crm△-R△resource△cleanup△prmDB△srv01
[srv01 ]# crm△-R△node△online△srv01
```

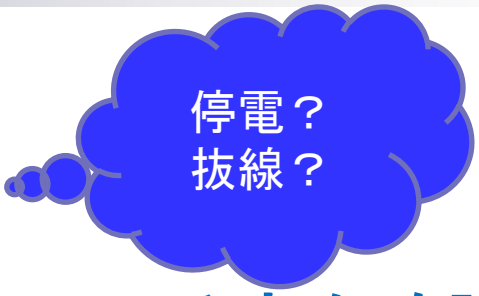
大変です！

どうされましたか？



リソースが両系共にいません！
どうしたらよいでしょうか！？

- まずは現状認識
  - 両系ともサーバ自体の電源確認
  - 両系ともに Pacemaker が稼働している事を確認



- srv02 を standby化
  - 最終的に srv01 でリソースを稼働させるため

```
[srv02 ]# crm△-R△node△standby△srv02
```

- 次に crm_mon コマンドを実行

```
[srv02 ]# crm_mon△-fA△-1
```

現状の確認

NW経路異常

ディスク異常

両系でリソース異常

リソース制約未解除

電源断

Pacemaker 未稼働

監視先確認等

Pacemaker 停止

データ復旧

srv01・srv02 の順で異常解除

制約解除

NW経路監視が両系ともに異常

- 経路監視先（デフォゲー等）が落ちてる可能性あり

```
[srv01 ]# crm_mon△-fA△-1
・・・
* Node srv01:
    + default_ping_set : 0 : Connectivity is lost
* Node srv02:
    + default_ping_set : 0 : Connectivity is lost
```

- NW関係の機材等を確認
- 問題が解決したらクラスタメンバに自動で復帰 standby を解除

```
[srv01 ]# crm△-R△node△online△srv02
```

ディスク監視が両系ともに異常

□共有ディスクに異常が発生している可能性あり

```
[srv01 ]# crm_mon△-fA△-1
・・・
* Node srv01:
    + diskcheck_status                : ERROR
* Node srv02:
    + diskcheck_status                : ERROR
```

□FCケーブル等に異常が無い場合

- Pacemaker を直ちに停止
- データの調査・復旧等を実施

■ 両系でリソース異常

□ 片系で監視異常、対向でも起動失敗等

```
[srv01 ]# crm_mon△-fA△-1
・・・
Failed actions:
  prmDB_monitor_10000 ¥
    (node=srv01, call=XXX, rc=7,  status=complete): ¥
      not running
  prmDB_start_0  ¥
    (node=srv02, call=XXX, rc=-2, status=Timed Out): ¥
      unknown exec error
```

- 同時刻に重い処理（ウィルススキャンとか）が走った…
- マウント対象デバイスの設定に問題がある…

両系でリソース異常

□異常情報をクリアしてサービス再開！

① srv01 のリソース異常を解除、リソース再開

- # crm△-R△resource△cleanup△prmDB△srv01

② srv01 でリソース再開を確認

- # crm_mon△-fA△-1

③ srv02 のリソース異常を解除

- # crm△-R△resource△cleanup△prmDB△srv02

④ srv02 のリソース解除を確認

- # crm_mon△-fA△-1

⑤ srv02 の standby 解除

- # crm△-R△node△online△srv02

リソース移動制約が効いている（解除忘れ）

- 以前、リソースを意図して対向サーバへ移動させた
  - crm_mon コマンドでは判定できず
  - ログファイルに残るが、ローテートされて消えてしまう
  - 下記コマンドで制約の存在確認が可能

```
[srv01 ]# cibadmin△-Q | grep△cli-standby | grep△srv02
 <expression id=“cli-standby-expr-grpDB” ¥
   attribute="#uname" ¥
   operation="eq" ¥
   value=“srv02" ¥
   type="string"/>
```

  - 例えば srv01 にてリソース異常が発生したが srv02 で上記制約が効いていると、両系で起動されない

リソース移動制約が効いている

□制約情報をクリアしてサービス再開！

① srv01 のリソース異常を解除、リソース再開

- # crm△-R△resource△cleanup△prmDB△srv01

② srv01 でリソースが再開された事を確認

- # crm_mon△-fA△-1

③ リソース起動制約を解除

- # crm△-R△resource△unmove△prmDB

④ 解除された事を確認

- # cibadmin△-Q | grep△cli | grep△srv02

⑤ srv02 の standby 解除

- # crm△-R△node△online△srv02

Pacemaker あるある ： リソース異常の原因

- 高負荷だった（バッチ処理・ウィルススキャン等）
- /tmp 配下のファイルが消された
- max_connections を超えていた
- pg_hba.conf（認証用ファイル）を編集した or 消した
- multipathd, iSCSI の起動漏れによるデバイス無効
- マウントする時に fsck の完全チェックが走った
- ${DocumentRoot}/index.html が無い or grep で失敗
- ログファイルのパーミッションが root だった
- その他：cib.xml の場所をド忘れ

お互い、気をつけましょう

さらにおまけ：起動スクリプトによる制御

- LSB（Linux Standard Base）の仕様に則ったスクリプトである事
  - http://refspecs.linuxfoundation.org/LSB_4.1.0/LSB-Core-generic/LSB-Core-generic/iniscrptact.html

① start / stop / status の各メソッドが実装されている。
② 停止中に start メソッドが実行され、正常起動した場合は "0" を返す
③ 停止中に start メソッドが実行され、起動失敗した場合は "0" 以外を返す
④ 稼動中に stop メソッドが実行され、正常停止した場合は "0" を返す
⑤ 稼動中に stop メソッドが実行され、停止失敗した場合は "0" 以外を返す
⑥ 稼動中に status メソッドが実行された場合は "0" を返す
⑦停止中に status メソッドが実行された場合は "0" 以外を返す
⑧停止中に stop メソッドが実行された場合は "0" を返す

さらにおまけ：起動スクリプトによる制御

- ただし！監視処理が緩い！
  - pid ファイルの存在確認・/proc/$PID 確認程度
  - プロセスがサスペンドしてても気づかない
  - RA であれば wget, select文等、より確実で高度な動作確認が可能
- 基本的には RA でリソース管理する事を勧めます
- 且つ、コミュニティ提供の RA を利用される際は、事前に必ず目を通される事を勧めます

個人的な見解：
snmpd, ntpd, multipathd 等、両系必ず動いていなくてはいけないリソースをクローンとして稼働させる時には 起動スクリプトでも良いのかなと思います
尚、クローンはリソースが稼働した状態でも停止させずにそのまま組み込めます

本日のお話

① フェイルオーバーに関する運用

② Pacemaker の自動起動・停止・リストア

③ 最後に stonith について

shutdown コマンドでサーバが停止しません！

大変です！

どうされましたか？

- 実は…Pacemakerを手動停止する前にサーバを緩やかに停止(※)する事は少し危険
  - リソース停止異常が発生すると、ダンマリしちゃう！
  - ゲストOSを管理対象にしていると、対向でゲストOSがまともに起動しない！
    - 詳細については別途…

※ # shutdown△-h△now

- 保守者はPacemakerを事前に停止させ、停止を確認した後にサーバの停止を行って下さい
- もしくは、上記状態になったら…

reboot△-f△-r

- 更に…Pacemaker の自動起動も、あんまりお勧めしません
  - 各ネットワーク、ちゃんと繋がってるか
  - 共有ディスクとの接続、問題無いか
  - リソースの設定ファイル等がキチンと用意されてるか
  - これらを保守運用者さんが確認してから起動しないとヘンなところで止まっちゃう…
- 保守者は Pacemaker を起動する前に、環境の確認を行って下さい

Pacemaker を起動したのに、ずっと
OFFLINE のままで、クラスタメンバに
入ってきません！

大変です！

どうされましたか？

- 実は…
  リストア手順を誤ると、クラスタに組みこまれない
  - 片系で正常稼働、もう片系を停止後、丸ごとリストア
  - リストア後に Pacemaker を起動すると、陥ります
  - 原因 : /var/lib/heartbeat/hb_generation
    （世代管理ファイル）の不一致によるもの
    - 起動時に 1 上がります
  - こうなると、リストアしたサーバを再起動するしかない

裏:簡単な再現方法(srv01 がリストアと想定)

- 両系で正常稼働の状態にする
- 片系(srv01)の Pacemaker を正常に停止
- 片系(srv01)の世代管理ファイル内の数値を少なくして保存
- 片系(srv01)の Pacemaker を起動

2 以上

- **srv01 のログ**

450 からERROR

```
srv01 heartbeat: [XXXXX]: WARN: ¥
  Message hist queue is filling up (376 messages in queue)
```

- **srv02 のログ**

500 まで
カウントアップ

```
srv02 heartbeat: [XXXXX]: ¥
  ERROR: should_drop_message: attempted replay attack [srv01]? ¥
  [gen = 1336642803, curgen = 1336642852]
```

裏：簡単な再現方法（srv01 がリストアと想定）

- 両系で正常稼働の状態にする
- 片系（srv01）の Pacemaker を正常に停止

2 以上

- 片
- 片
- s

保存

srv01 h
Messa

- s

srv02 he
ERROR: should_drop_message: attempted replay attack [srv01]? ¥
[gen = 1336642803, curgen = 1336642852]

！注意！

この手順を行うと、srv01 を再起動しなくてはいけません

- 対策
  - リストア前に世代管理ファイル等（※）を削除する
  - もしくはリストア後（且つ Pacemaker 起動前）に世代管理ファイル等（※）を削除する
    - 当ファイルが無い状態で起動するのは、問題ない
    - ちなみに以前よりも大きな値になっても、問題ない
  - もし発生したら、リストアしたサーバを再起動させ、世代管理ファイル等（※）を削除して Pacemaker を起動

- リストアする運用には上記運用を徹底して下さい

※:
[srv01]# rm△-f△/var/lib/heartbeat/crm/*
[srv01]# rm△-f△/var/lib/heartbeat/hb_generation

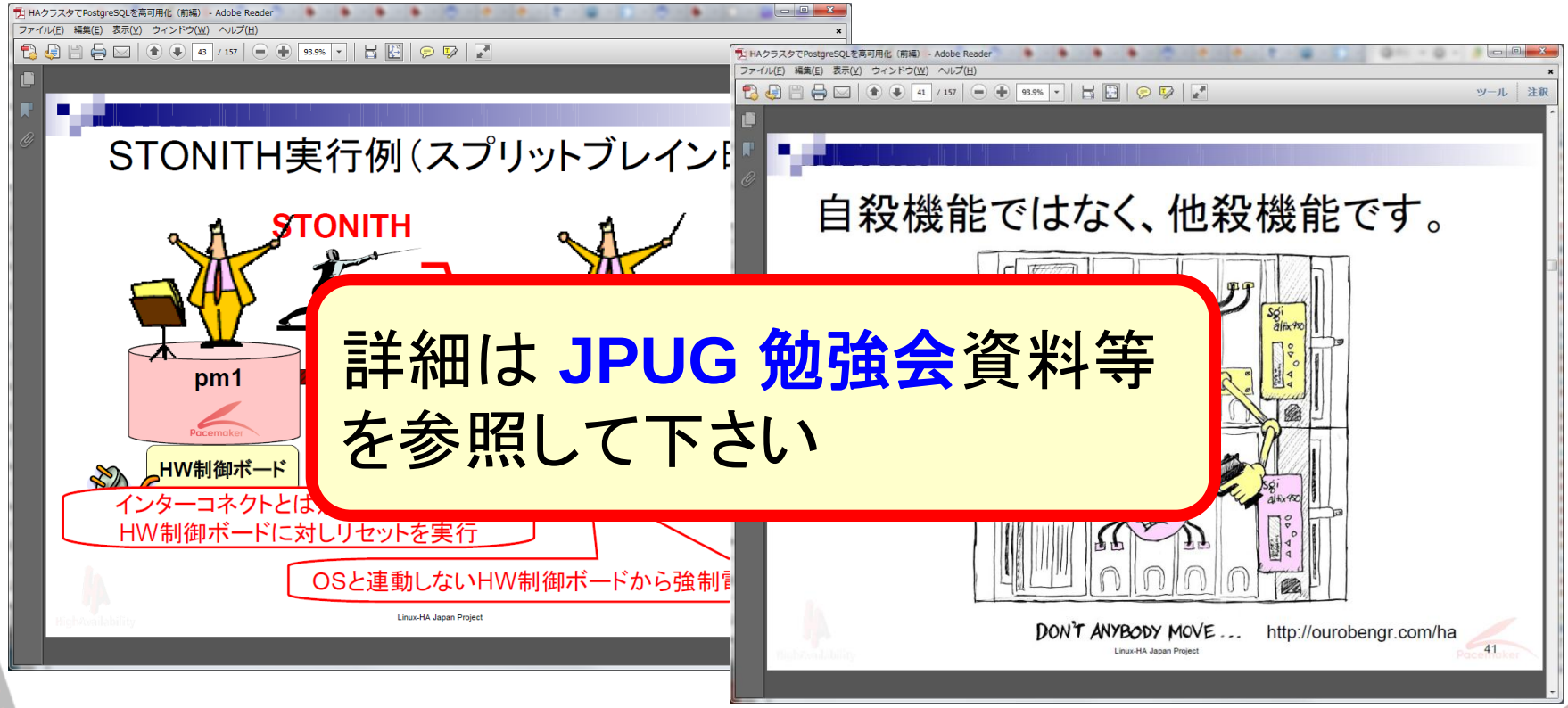
本日のお話

① フェイルオーバーに関する運用

② Pacemaker の自動起動・停止・リストア

③ stonith について

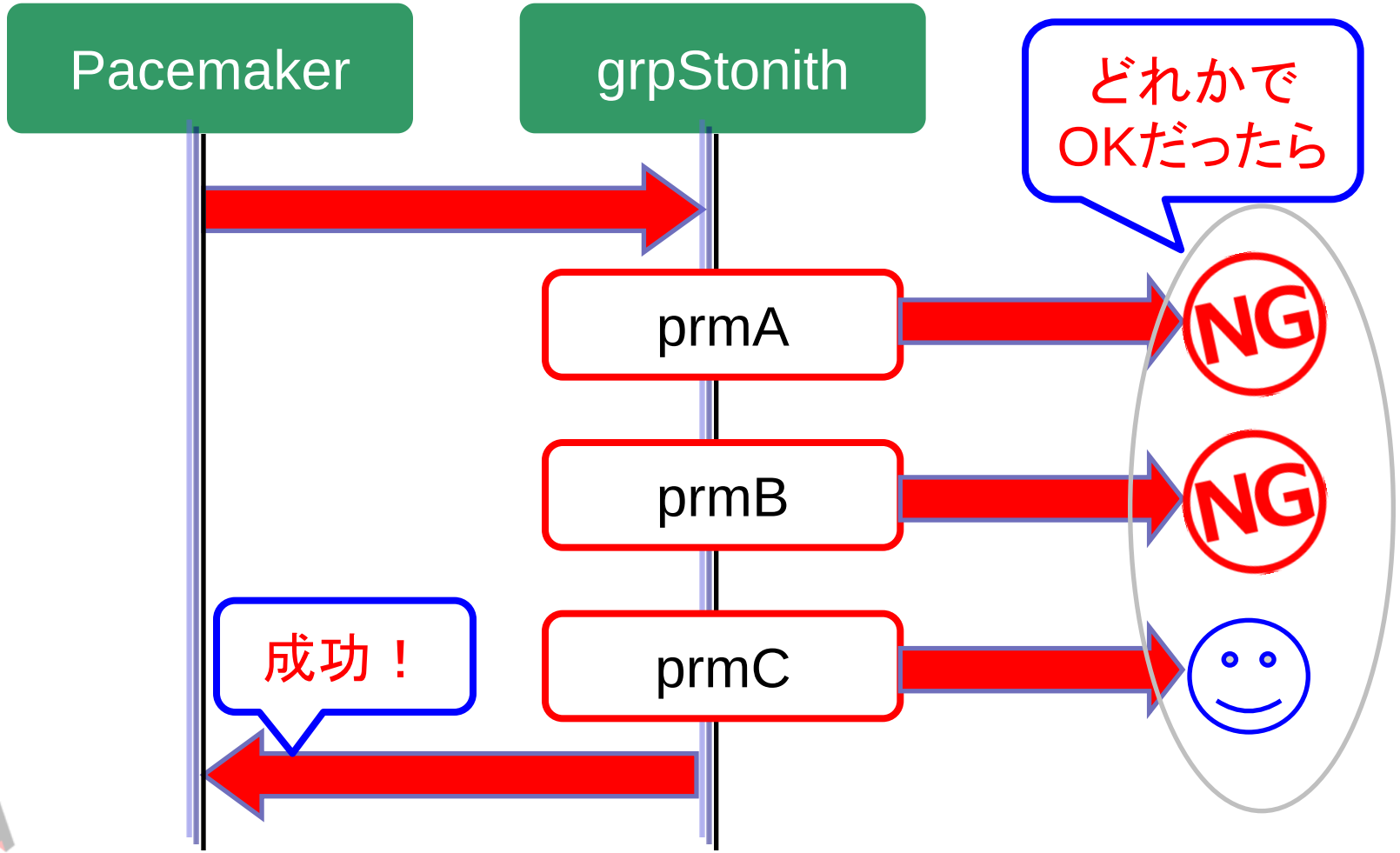
- stonith について大まかに触れておきます
  - stonith はスプリットブレイン回避のためのしくみ
    - 発動契機1： リソース停止失敗
    - 発動契機2： インターコネクトLAN抜けてた



詳細は JPUG 勉強会資料等を参照して下さい

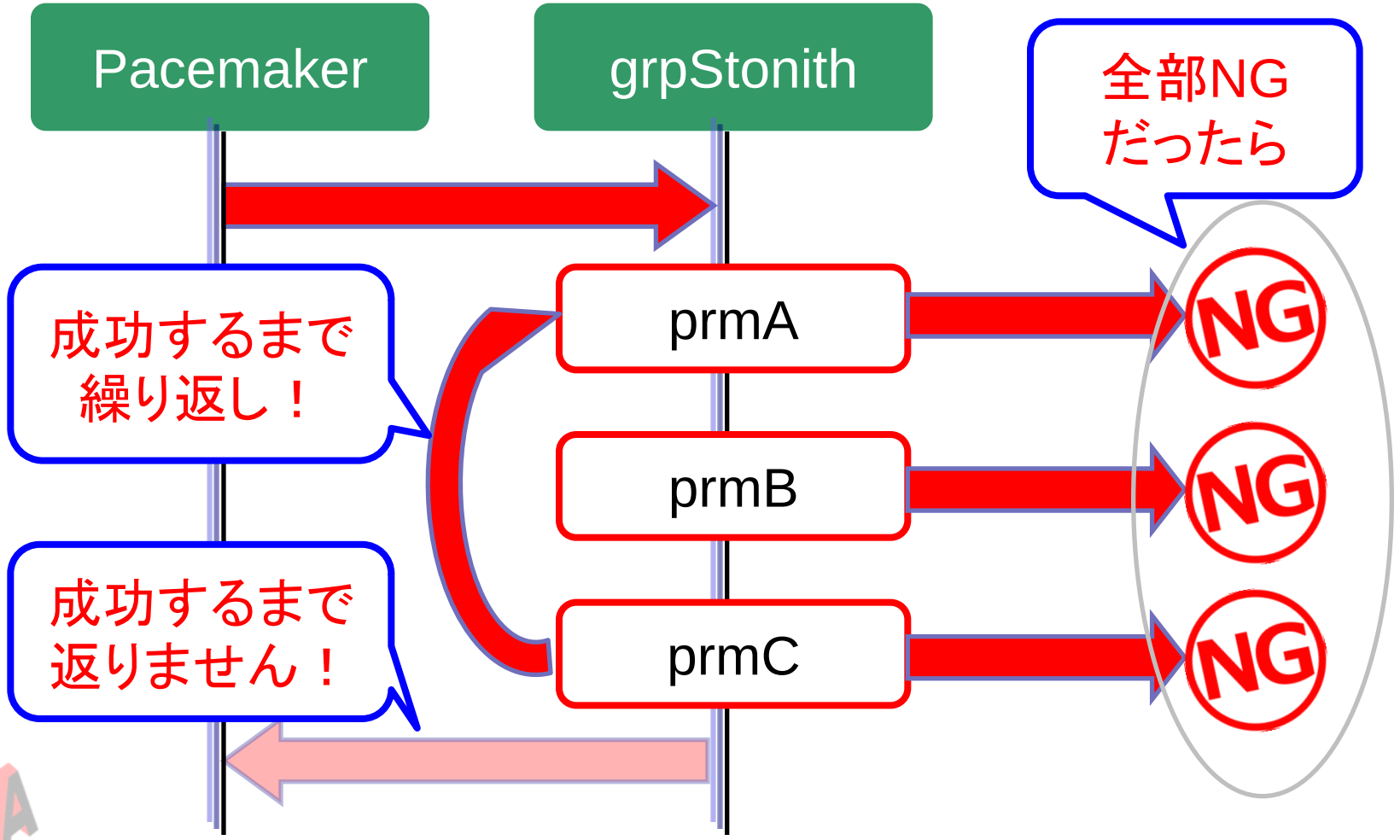
stonith について大まかに触れておきます

stonith をグループにした時の大まかな動作



stonith について大まかに触れておきます

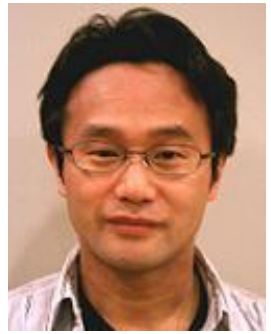
stonith をグループにした時の大まかな動作



たまに聞く質問:

stonith による相撃ちって、
起こりますか？

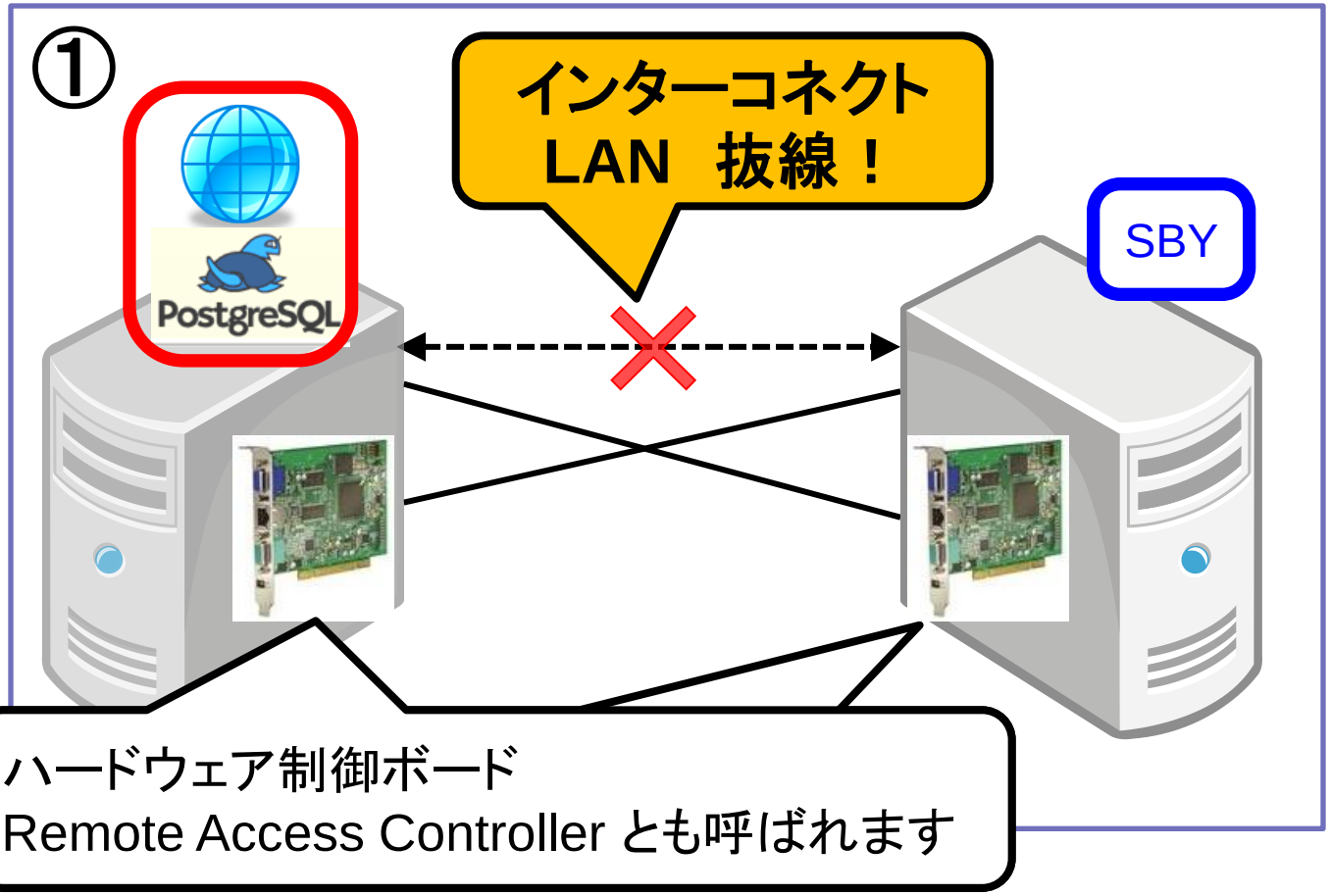
起こります！
待機系が生き残る可能性もあります！



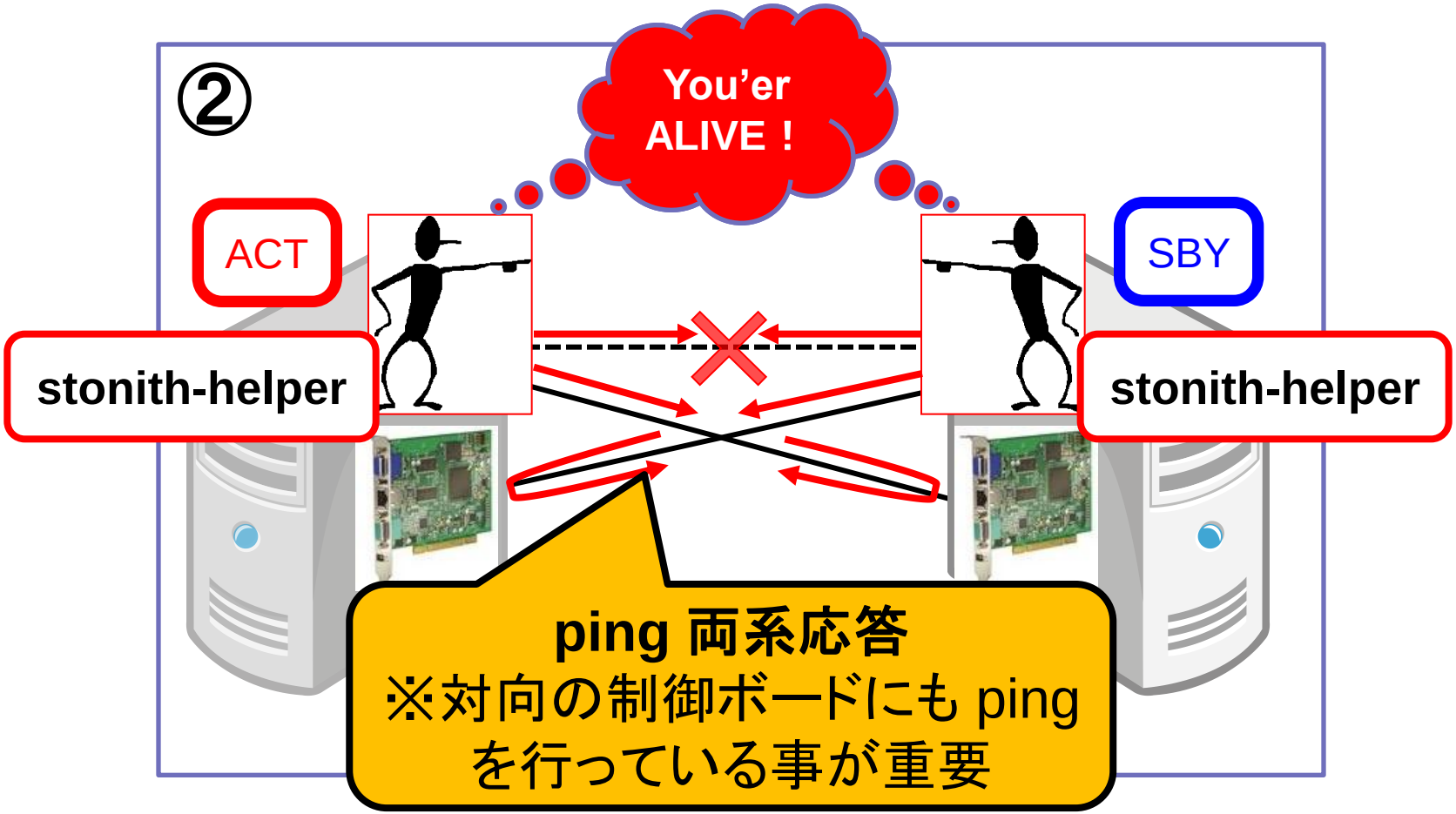
回避しなくちゃいけませんよね？
どうしたらいいんでしょう？

stonith-helper を使います！

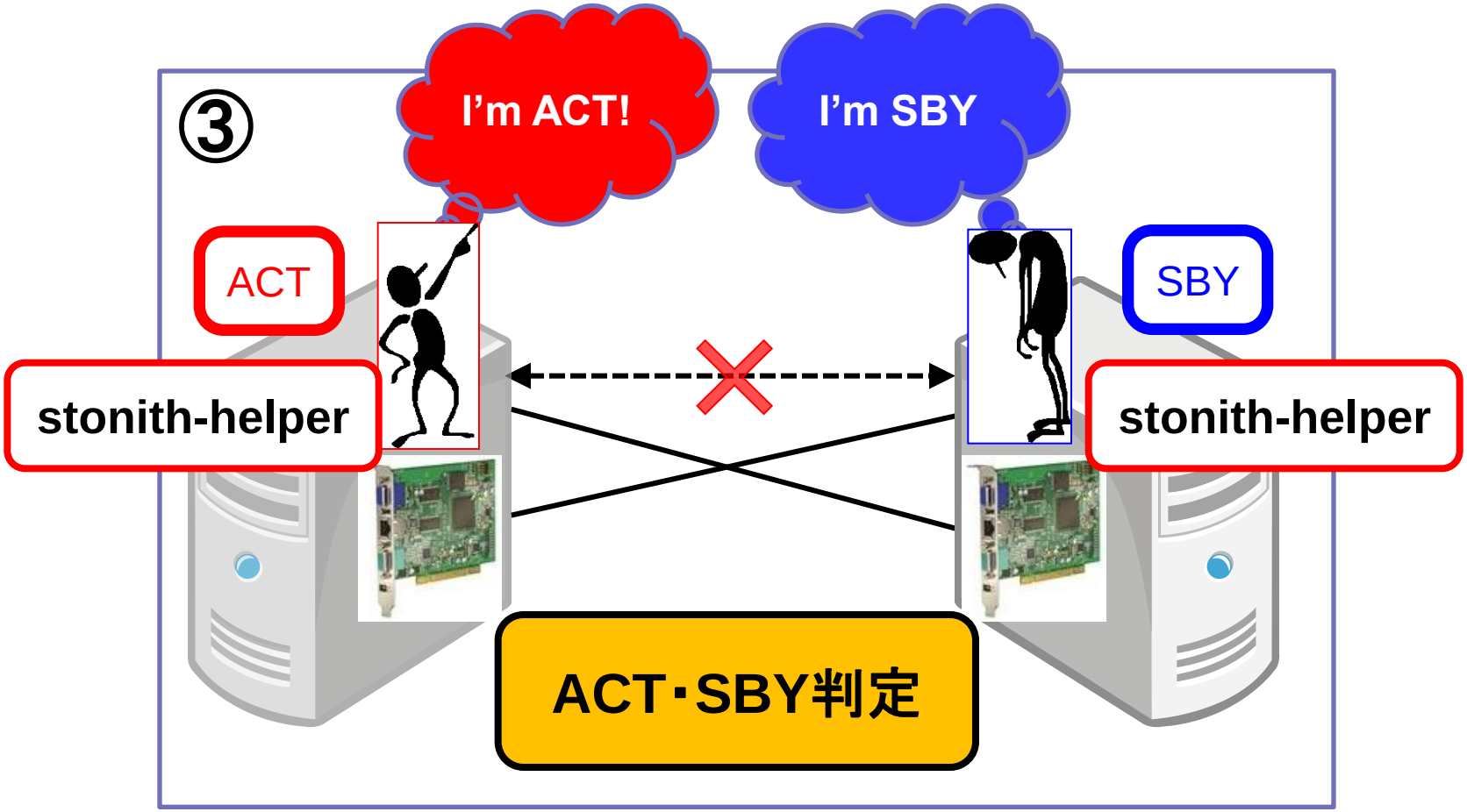
- stonith-helper とは、どちらのサーバを生かすかを判断するリソース！



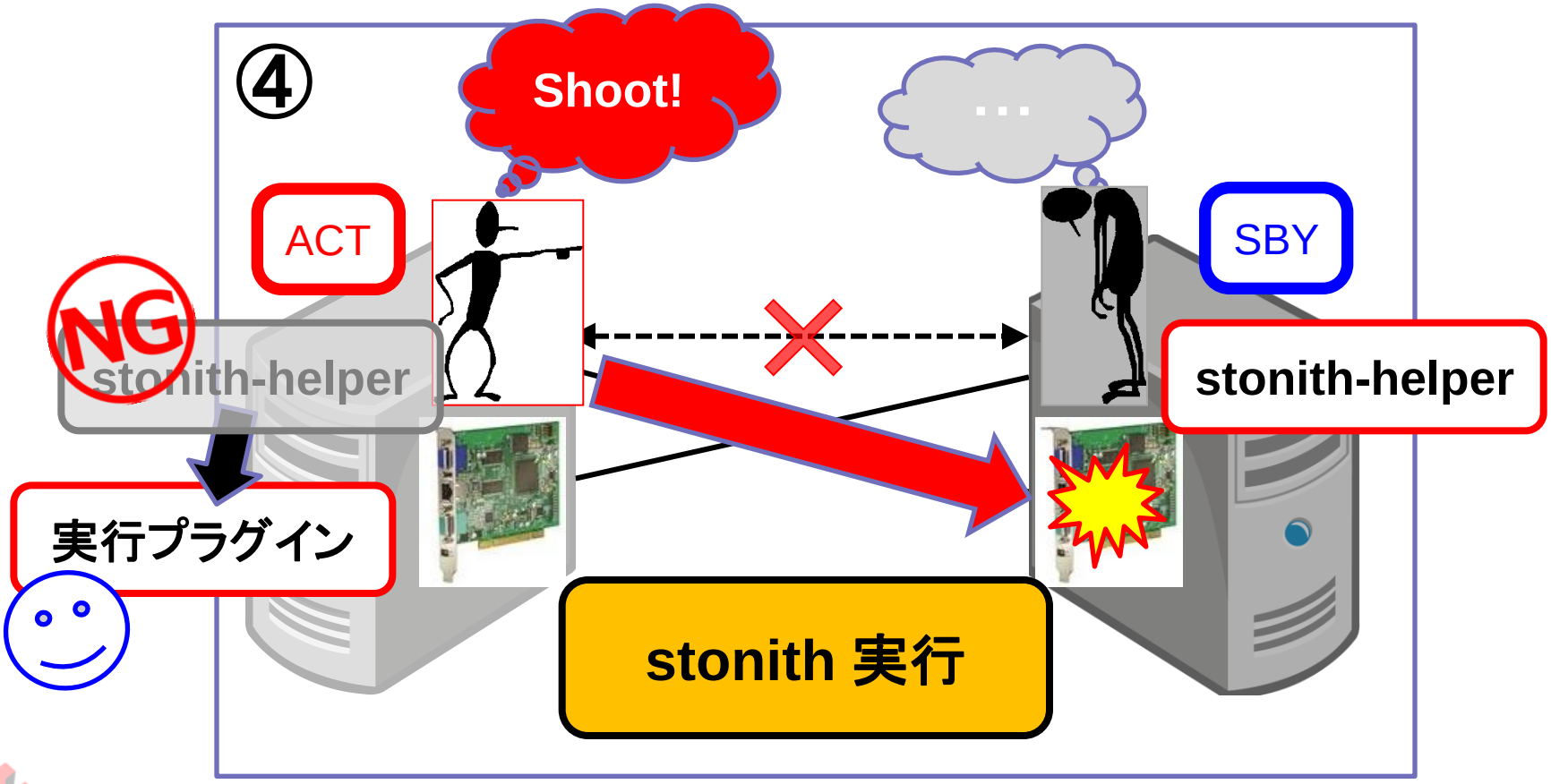
- stonith-helper とは、どちらのサーバを生かすかを判断するリソース！



- stonith-helper とは、どちらのサーバを生かすかを判断するリソース！



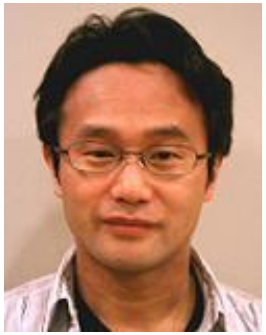
- stonith-helper とは、どちらのサーバを生かすかを判断するリソース！



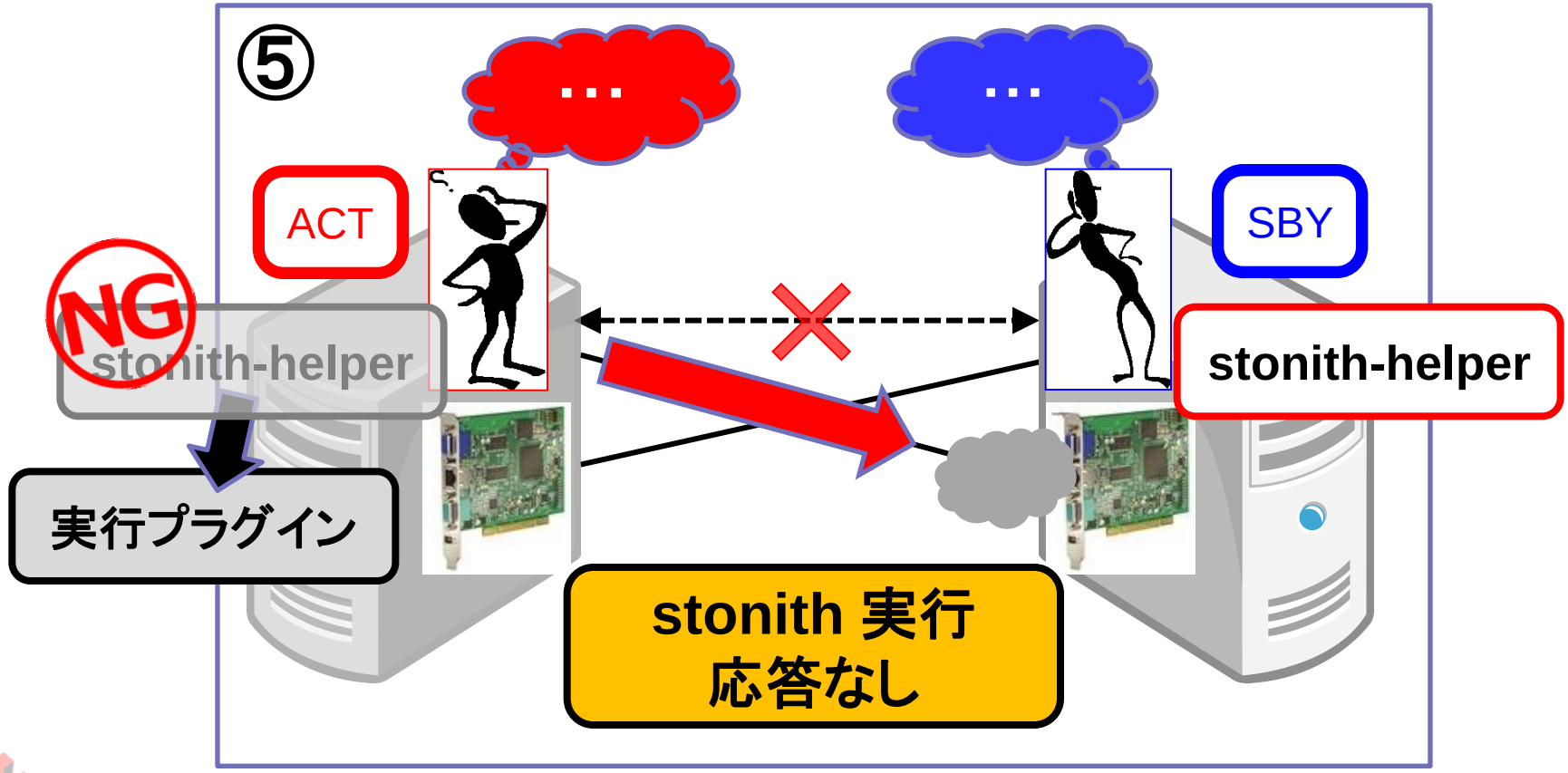
更に聞く質問：

でも 制御ボードが変だった場合って
どうなるんですか？

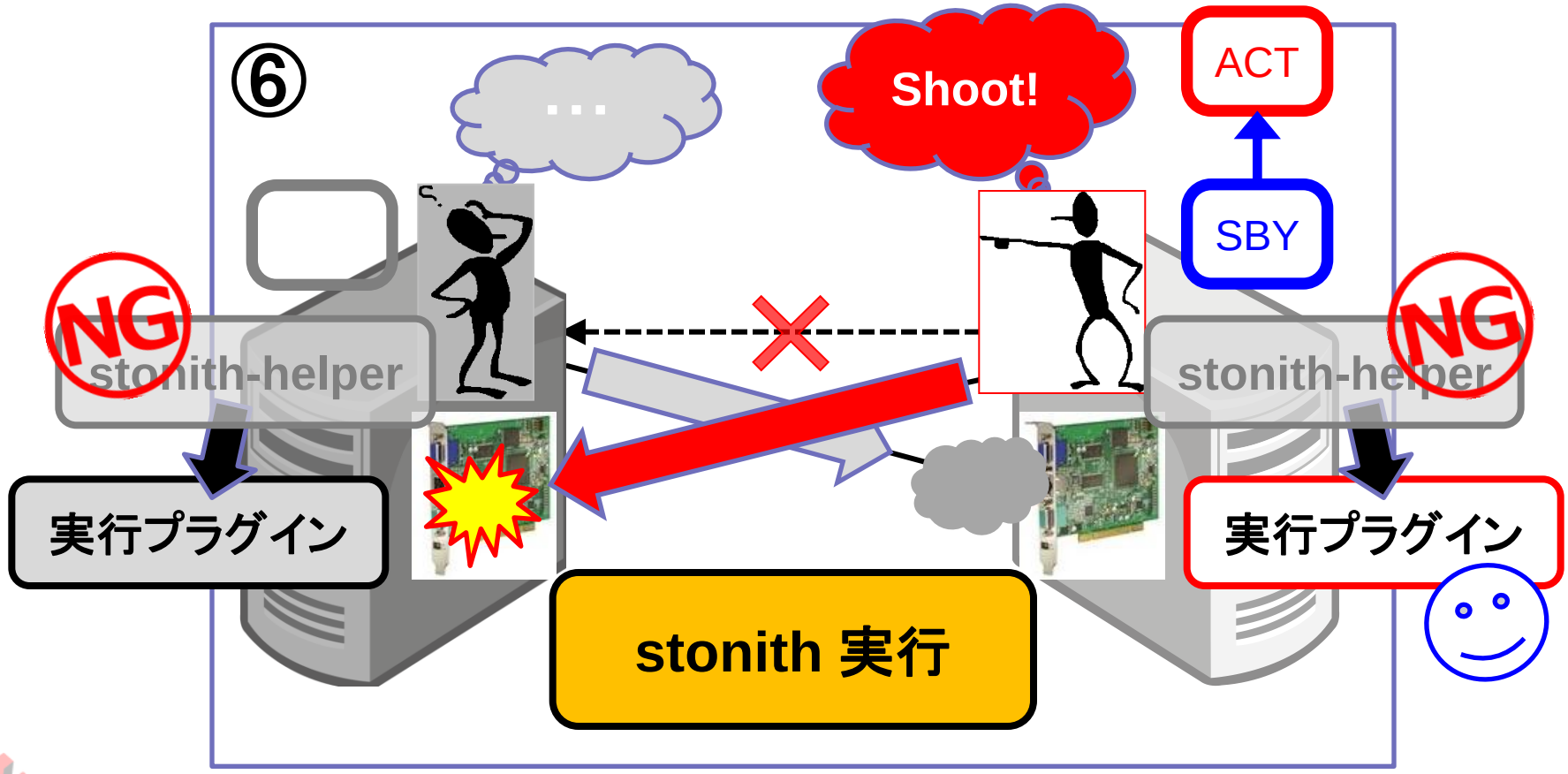
その時は、やっぱり stonith-helper が
判断して、対向サーバが撃ちます！



- stonith-helper とは、どちらのサーバを生かすかを判断するリソース！



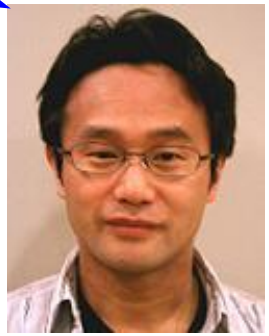
- stonith-helper とは、どちらのサーバを生かすかを判断するリソース！



更に聞く質問：

でも 制御ボードが両系ともに変だった場合ってどうなるんですか？

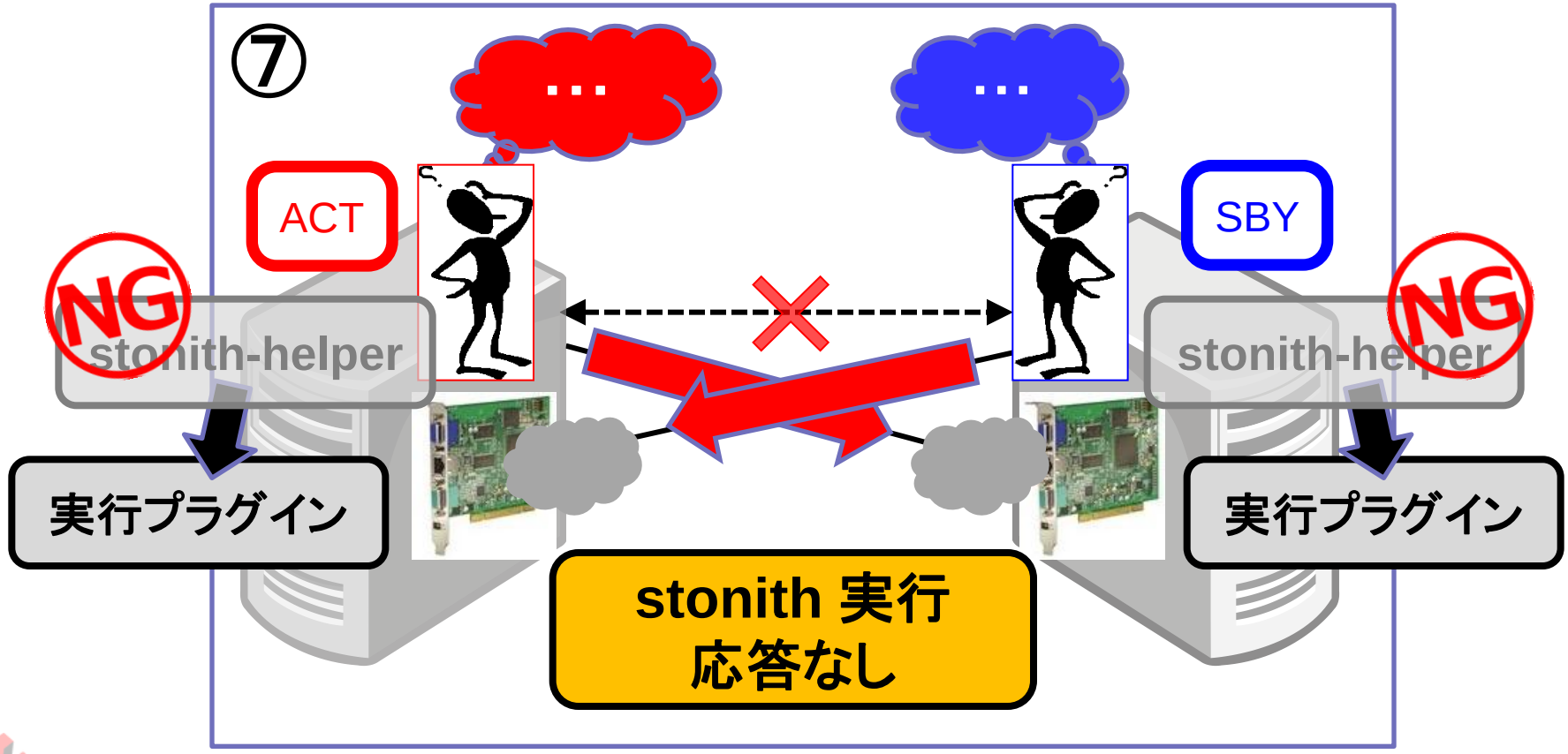
両系とも待つ状態になり、タイムアウトでループし続けることになります。



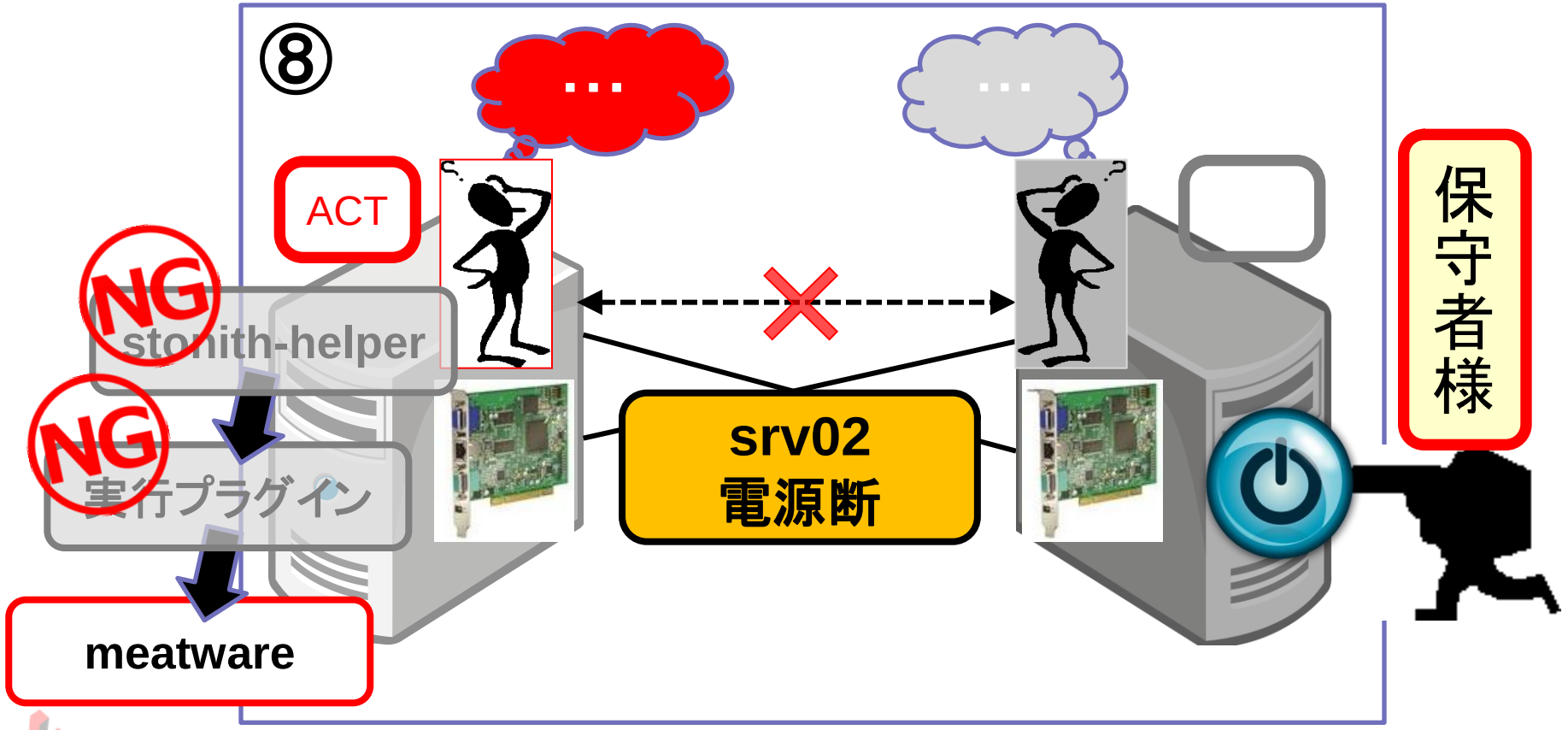
ちょっとまずくないですか？

保守者介在してもらうため meatware を使います！

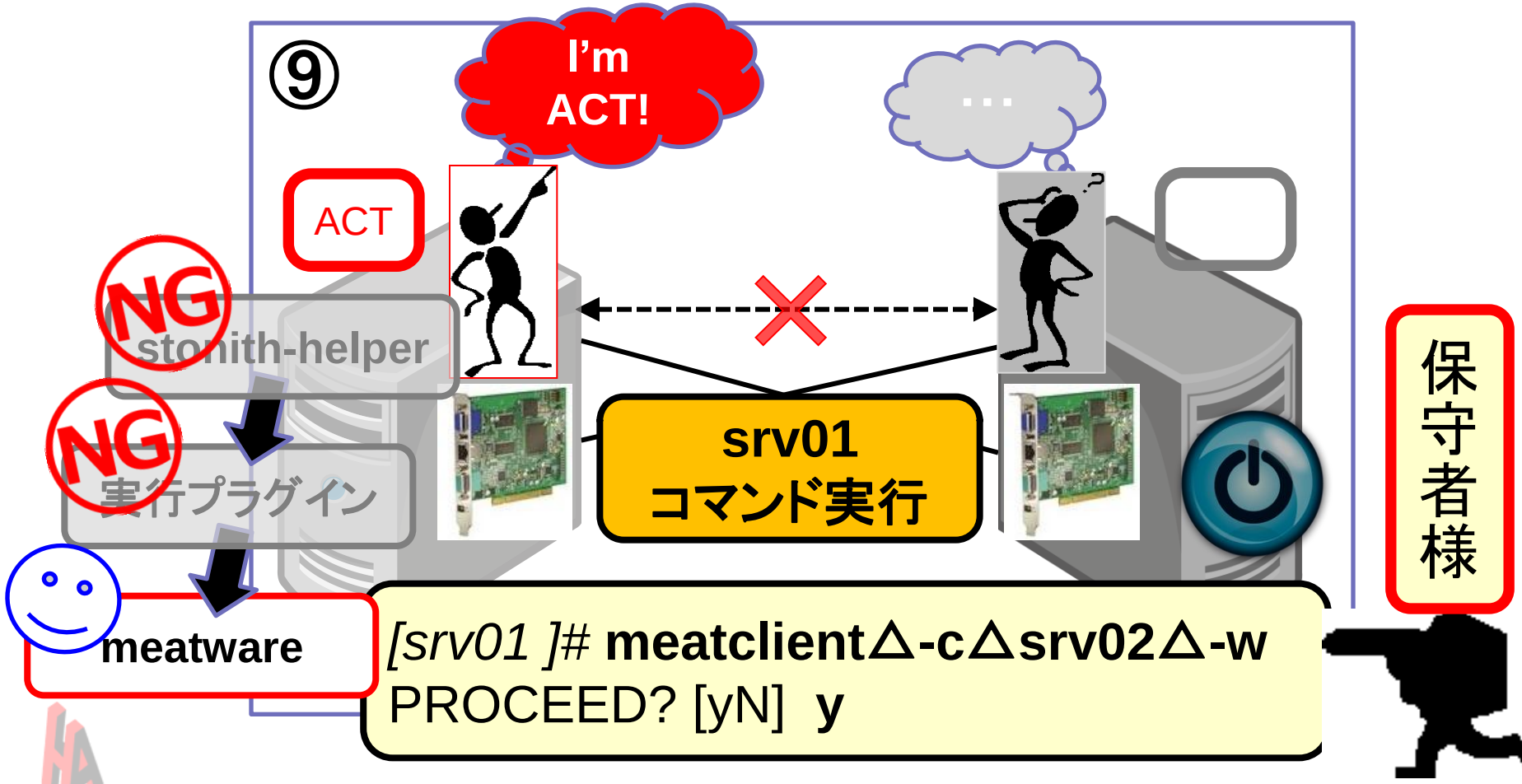
- meatware とは、保守者から Pacemaker へ対向停止を報告する為のインタフェース！



- meatware とは、保守者から Pacemaker へ対向停止を報告する為のインタフェース！



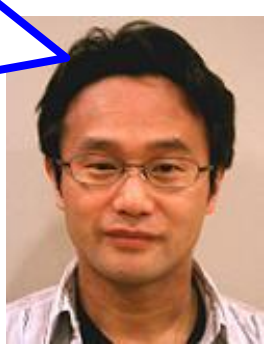
- meatware とは、保守者から Pacemaker へ対向停止を報告する為のインタフェース！



更に聞く質問：

でもこれって 制御ボード向けLANも抜線されてると、まずくないですか？

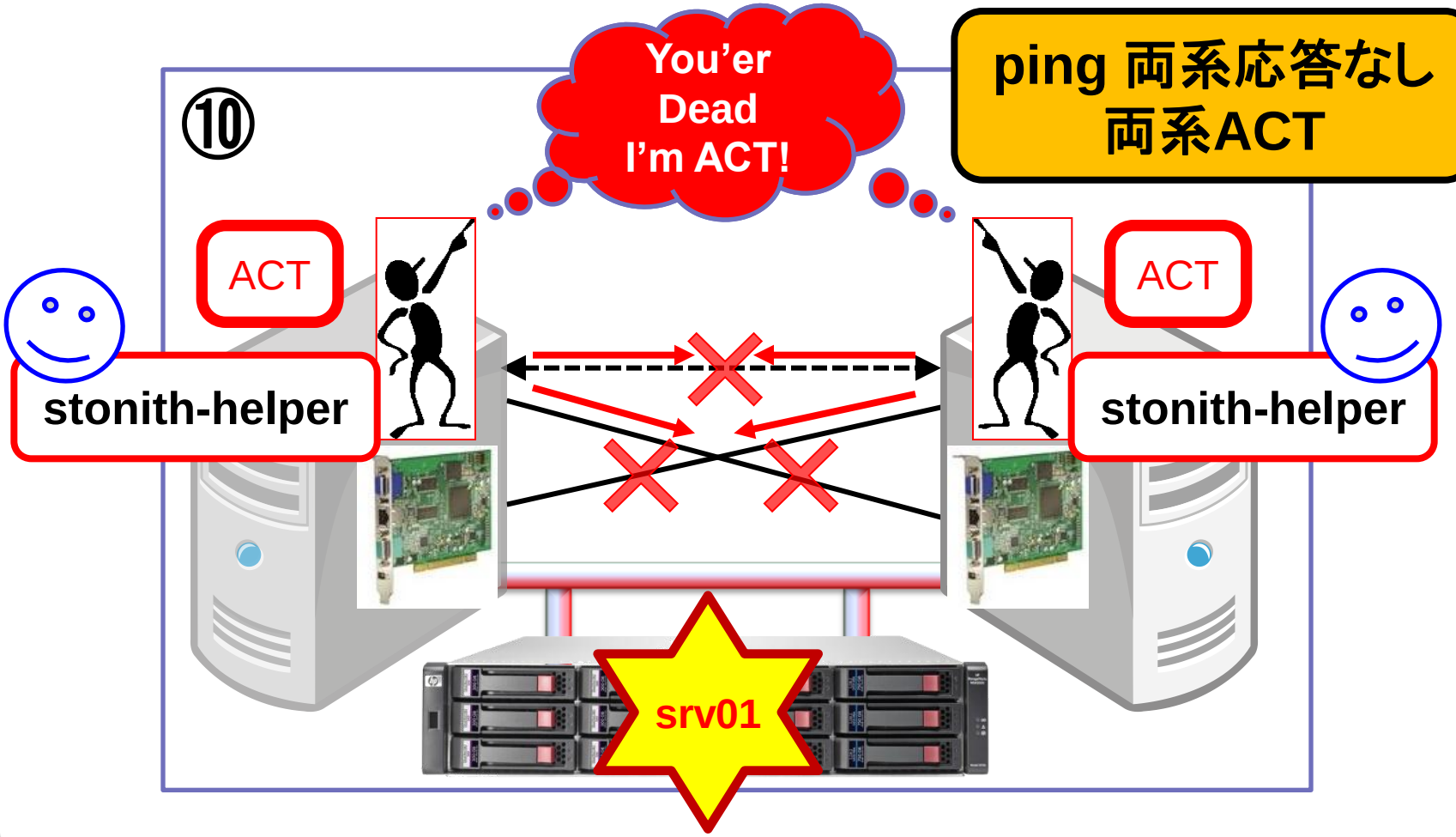
まずいです！！
stonith-helper で両系とも **OK** と判定します！



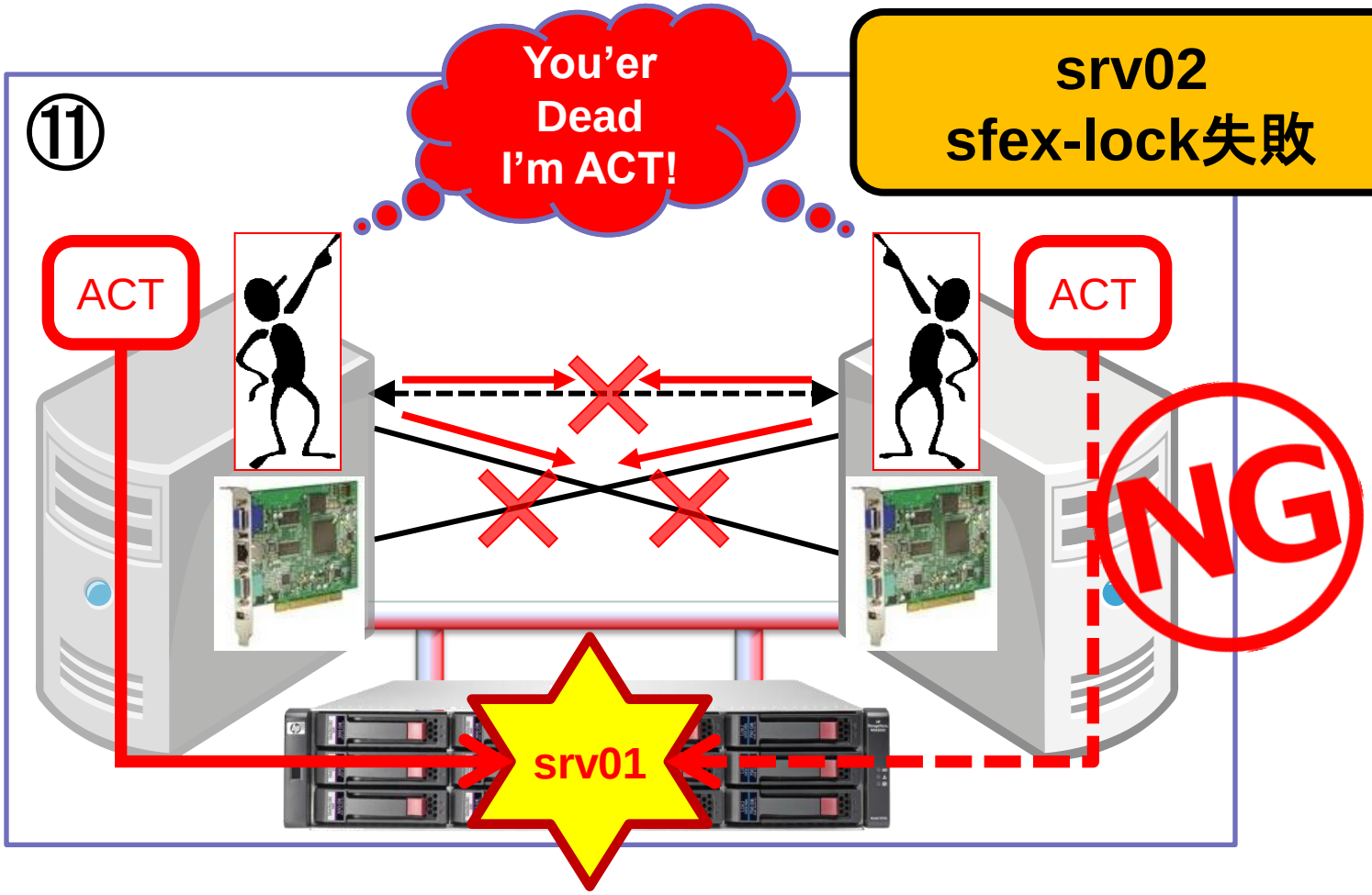
回避しなくちゃいけませんよね？
どうしたらいいんでしょう？

最後の砦 sfex を使用します！

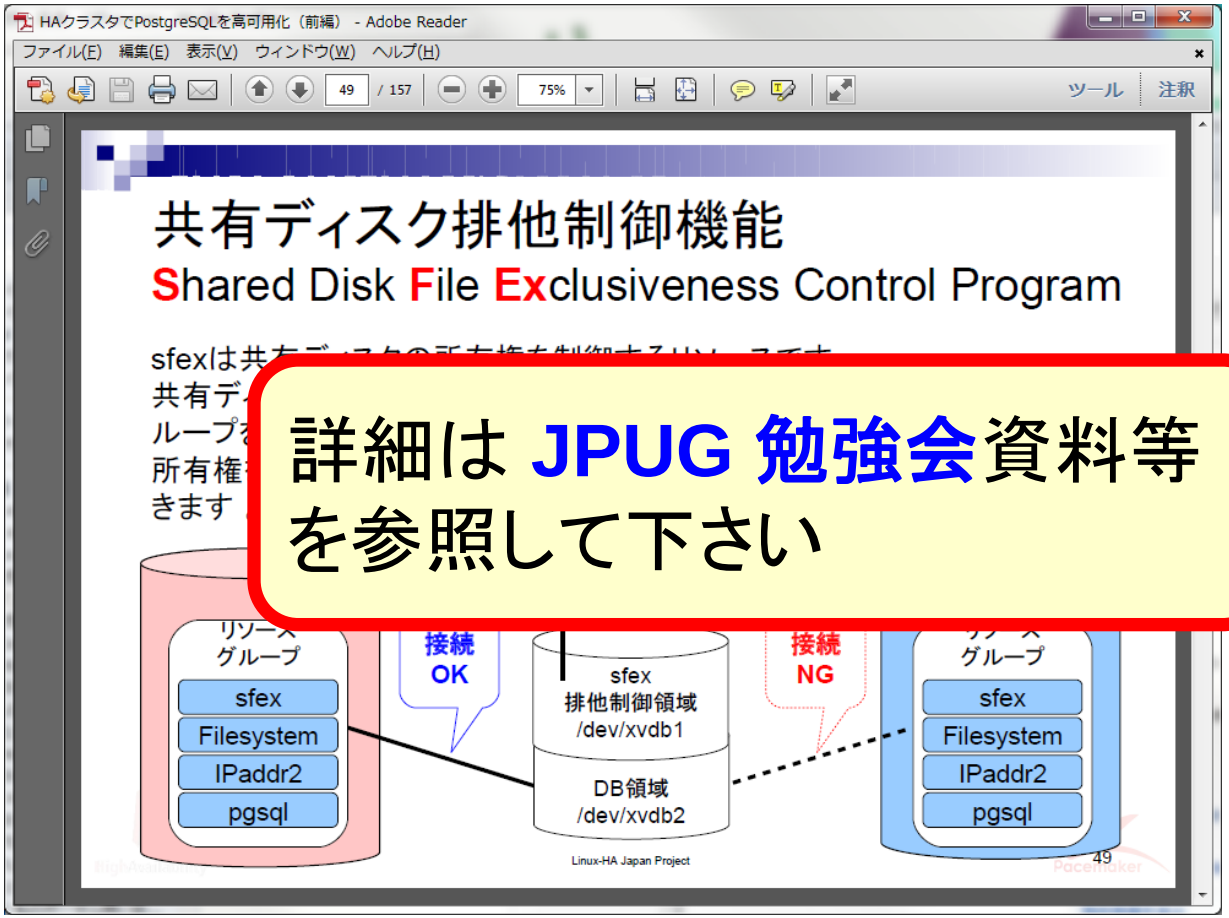
sfexリソースは、最後の砦！



sfexリソースは、最後の砦！



- 尚 sfex リソースの詳細は コミュニティ公開資料を参考にして下さい！



まとめ

- stonith に（最低限）**stonith-helper** 必須！
- meatware が無いと、サーバの電源をコンセントから抜線！
- 下記リソース配置が現状では理想

```
Resource Group: grpStonith1
  prmStonith1-1 (stonith:external/stonith-helper): Started srv01
  prmStonith1-2 (stonith:external/実行プラグイン):  Started srv01
  prmStonith1-3 (stonith:meatware):                 Started srv01
```

- （共有ディスクがあるなら）**sfex** は必須！

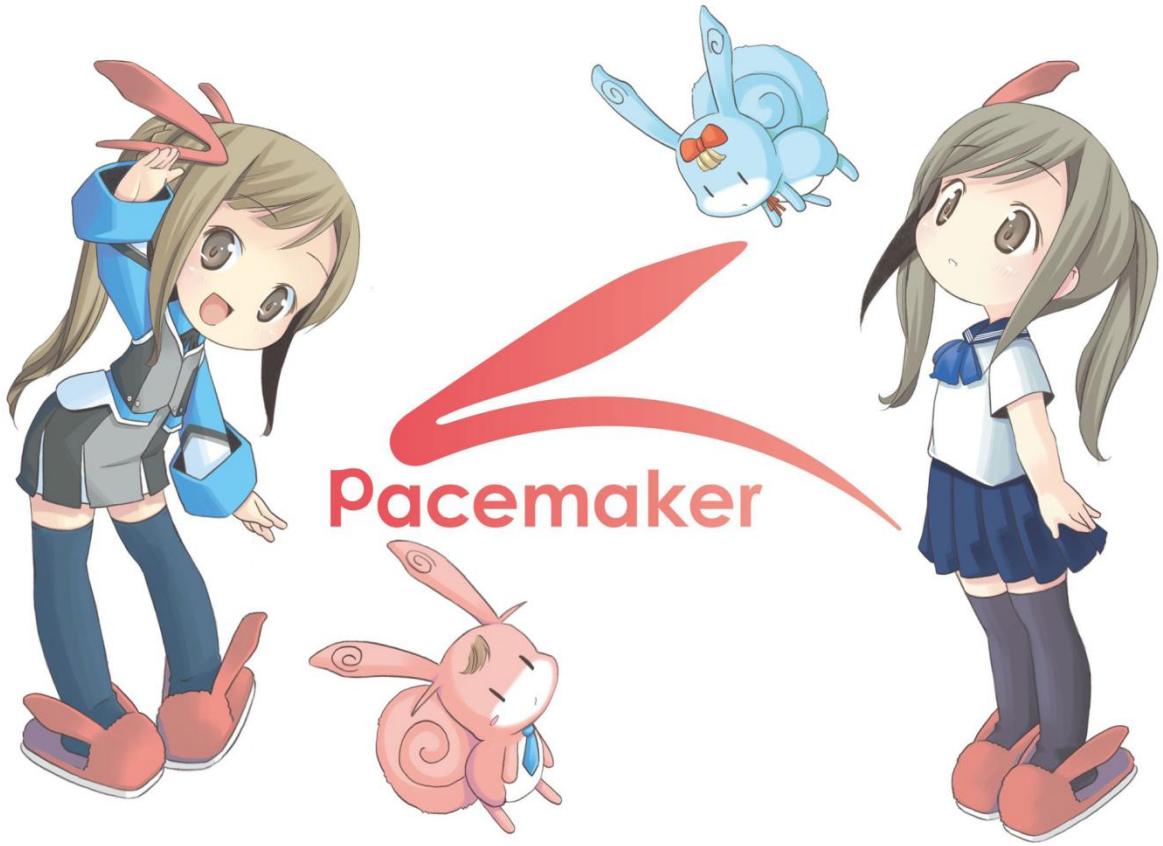
```
Resource Group: grpDB
  prmSFEX (ocf:heartbeat:sfex): Started srv01
  ...
```

ちなみに…実行プラグインについて

- 実行プラグインは、ハードウェア制御ボードを直接制御します
- 2012年8月時点で Linux-HA Japan で検証している
  実行プラグインと制御ボードの組み合わせは下記の通り

実行プラグイン名	ハードウェア制御ボード
ipmi	HP iLO3・DELL DRAC・IBM IMM HP iLO2（ただしファームウェアがバージョン2以上） （HP MicroServer 付属の制御ボードの制御も ipmi です）
riloe	HP iLO1・HP iLO2
ibmrsa-telnet	IBM RSA

以上です！



ご清聴、ありがとうございました

クイズの答え

別冊あんどりゅーくん（第2号）にて
提示されているノウハウです

- 「-R」をつけないと…

```
[srv01 ~]#  crm△resource△move△dummy△srv02△force
[srv01 ~]#  tail△/var/log/messages
Jul DD SS:MM:SS srv01 ¥
    crm_resource: [XXXXX]: info: Invoked: ¥
    crm_resource -M -r dummy –node=srv02 –force
```

- 「-R」をつけると…

```
[srv01 ~]#  crm△-R△resource△move△dummy△srv02△force
.EXT crm_resource -M -r ‘dummy’ –node=’srv02’ –force
[srv01 ~]#  tail△/var/log/messages
Jul DD SS:MM:SS srv01 ¥
    crm_resource: [XXXXX]: info: Invoked: ¥
    crm_resource -M -r dummy –node=srv02 –force
```