ディザスタリカバリもオープンソースで!?

～高可用クラスタソフトウェア Pacemaker 次期バージョンプレビュー～

2012年9月8日

Linux-HA Japan プロジェクト
http://linux-ha.sourceforge.jp/
森 啓介

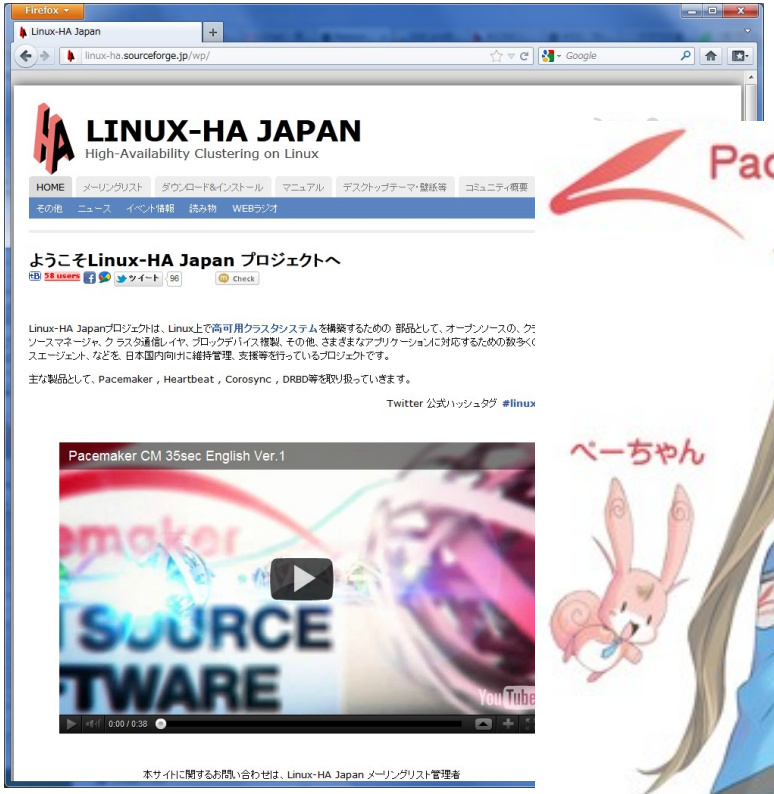
自己紹介

- 名前: 森 啓介 (Keisuke MORI)
  - twitter: @ksk_ha
- Linux-HA Japanプロジェクト関連の活動
  - Pacemaker-1.0系(安定版)のパッチメンテナ
    - 現在Pacemaker-1.0.13リリースに向けて鋭意作業中です!
- 本業
  - 所属会社: NTTデータ先端技術株式会社
    - 所在地: 月島 ー ぜひもんじゃを食べにどうぞ!
  - 普段の業務: NTT OSSセンタ
    - NTTグループ内におけるPacemaker/Heartbeatの導入支援・サポート
    - バグ報告・パッチ作成などによるNTTから開発コミュニティへのフィードバック・貢献
- 賞罰
  - 賞:第7回 日本OSS貢献者賞 授賞(2012年3月)。数多くの方のおかげです！
  - 罰: 中学２年生のとき髪ボサボサのだらしない格好で登校し学年主任に往復ビンタを貰う。

もくじ

- Pacemakerとは
  - Pacemakerの概要
  - 開発バージョンの近況
- Pacemaker次期バージョントピック紹介
  - ディザスタリカバリ対応機能 (booth/ticket)
  - kdump連携機能
  - その他の機能トピック
- Pacemakerの今後
  - クラウドとの関わり

Pacemakerってご存知ですか?



Pacemakerとは…

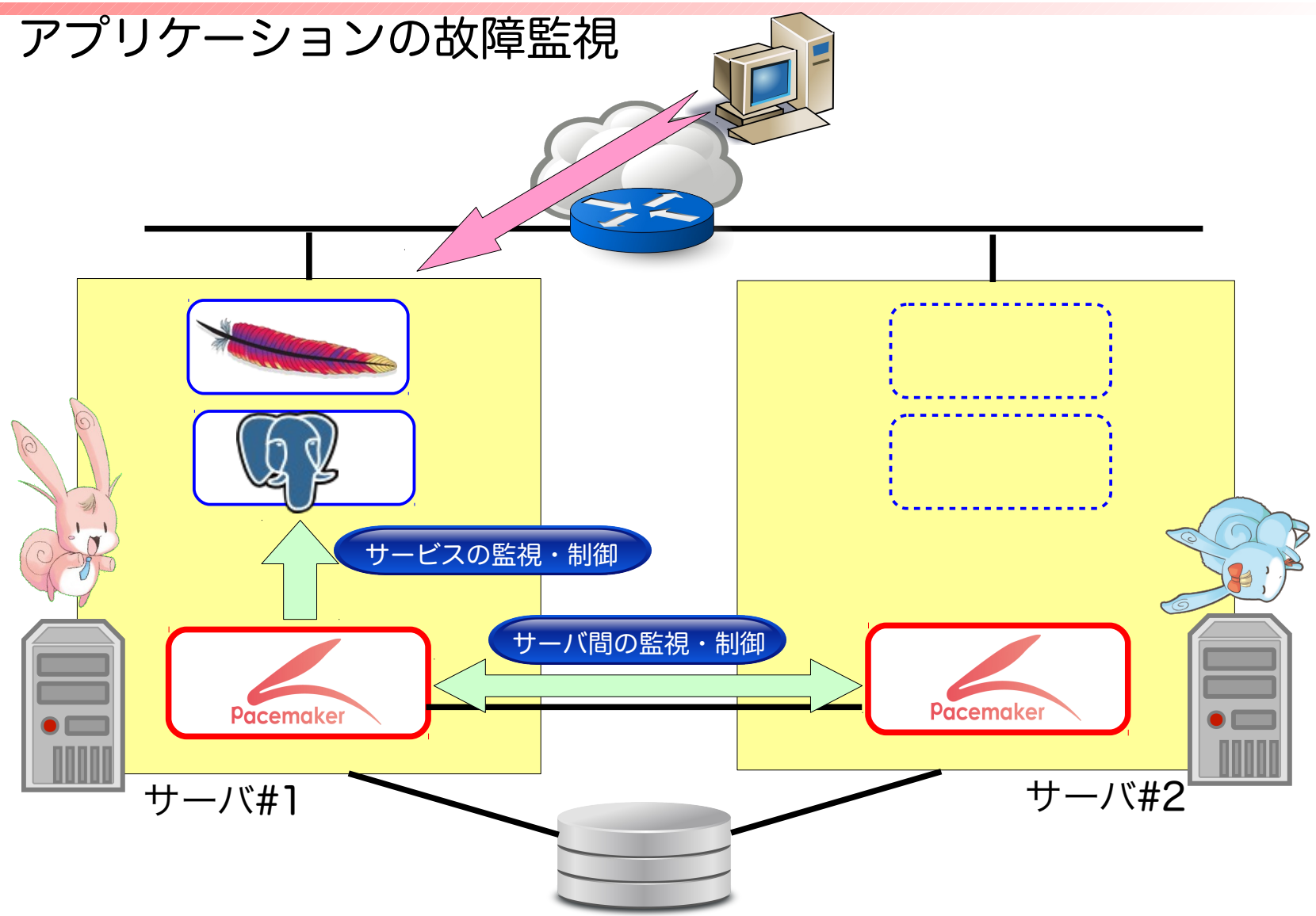
✕ 擬人化ソフトウェアです。

○ 高可用(HA)クラスタソフトウェアです。

Linux-HA Japan プロジェクト 2F 205教室にてデモ展示中！

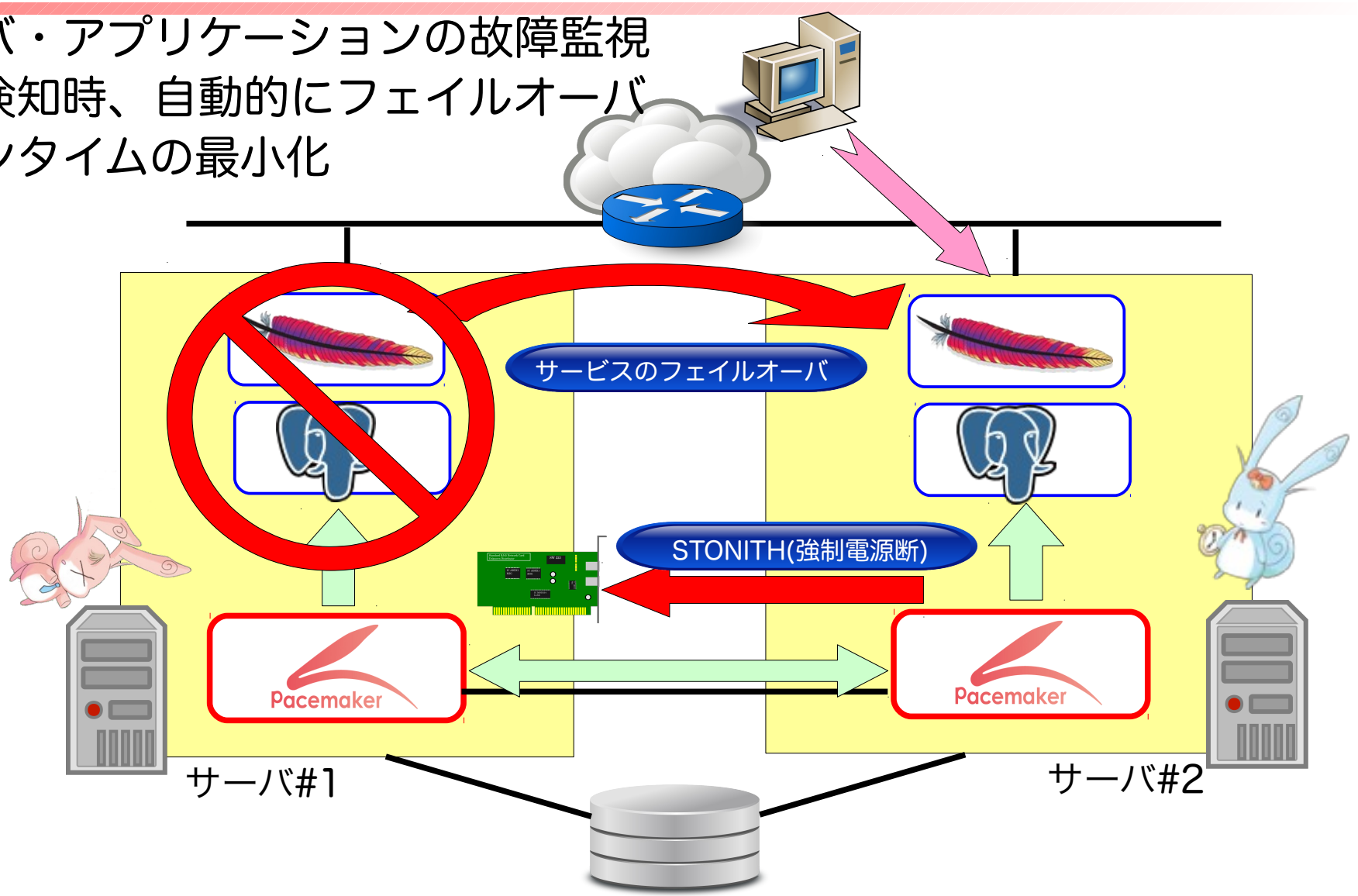
Pacemakerの概要

- サーバ・アプリケーションの故障監視

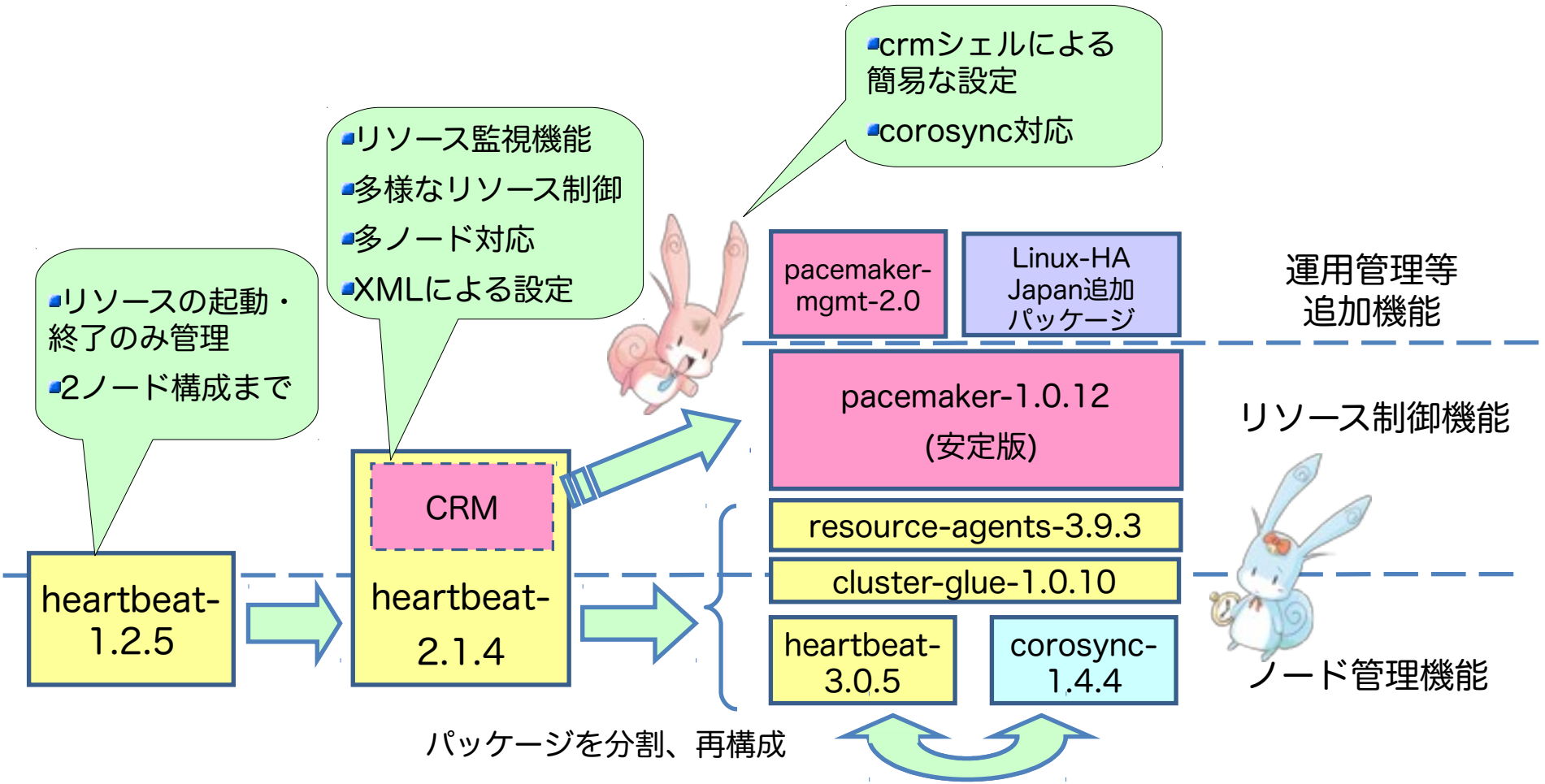


Pacemakerの概要

- サーバ・アプリケーションの故障監視
- 故障検知時、自動的にフェイルオーバ
- ダウンタイムの最小化

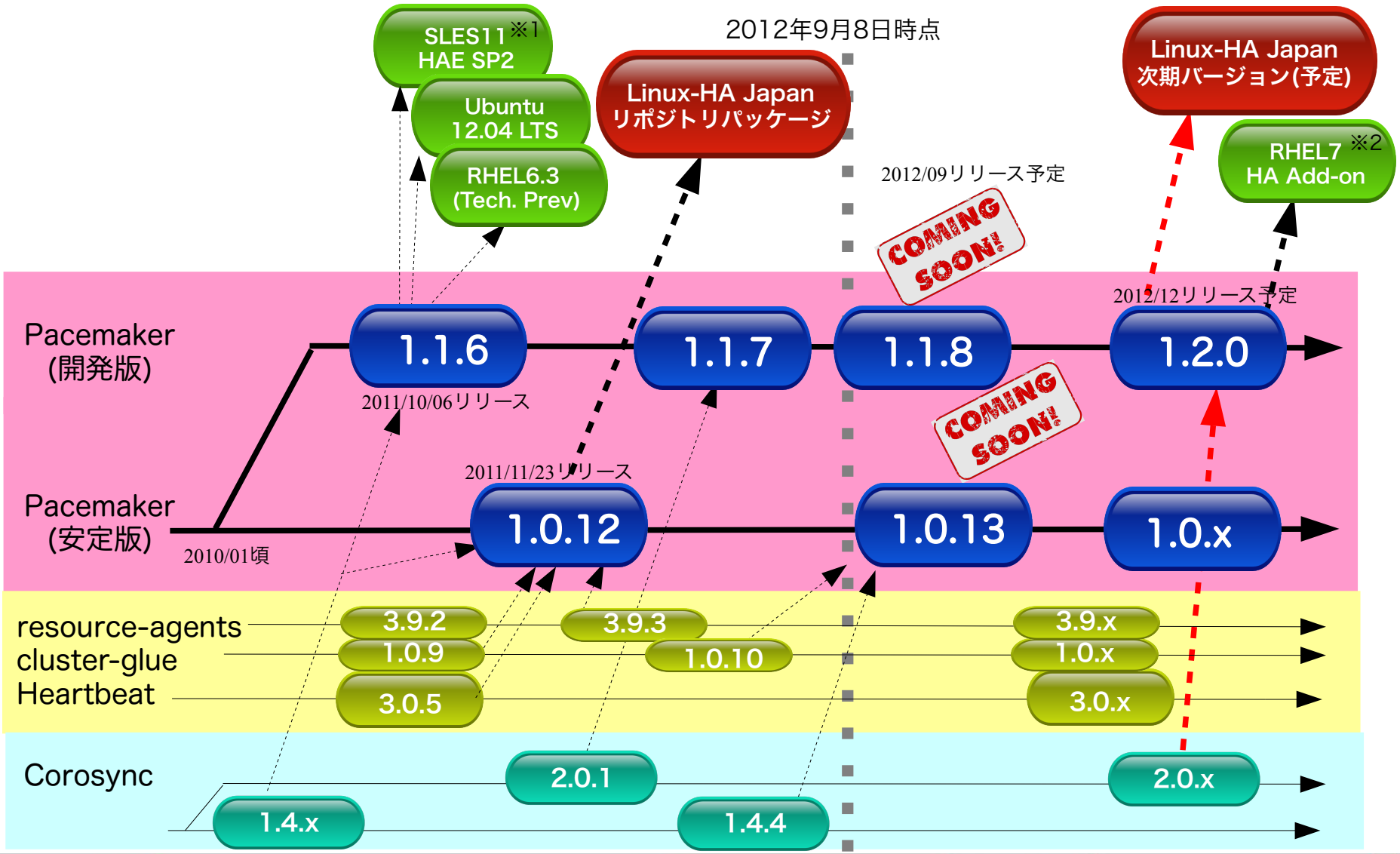


Pacemakerの生い立ち



CRM: Cluster Resource Manager
バージョン番号は2012年9月時点での最新版

Pacemaker リリースタイムライン



※1 SUSE Linux Enterprise Server 11 High Availability Extension
※2 Red Hat Enterprise Linux High Availability Add on

バージョンの依存関係は一部省略しています

Pacemaker開発バージョンの近況まとめ

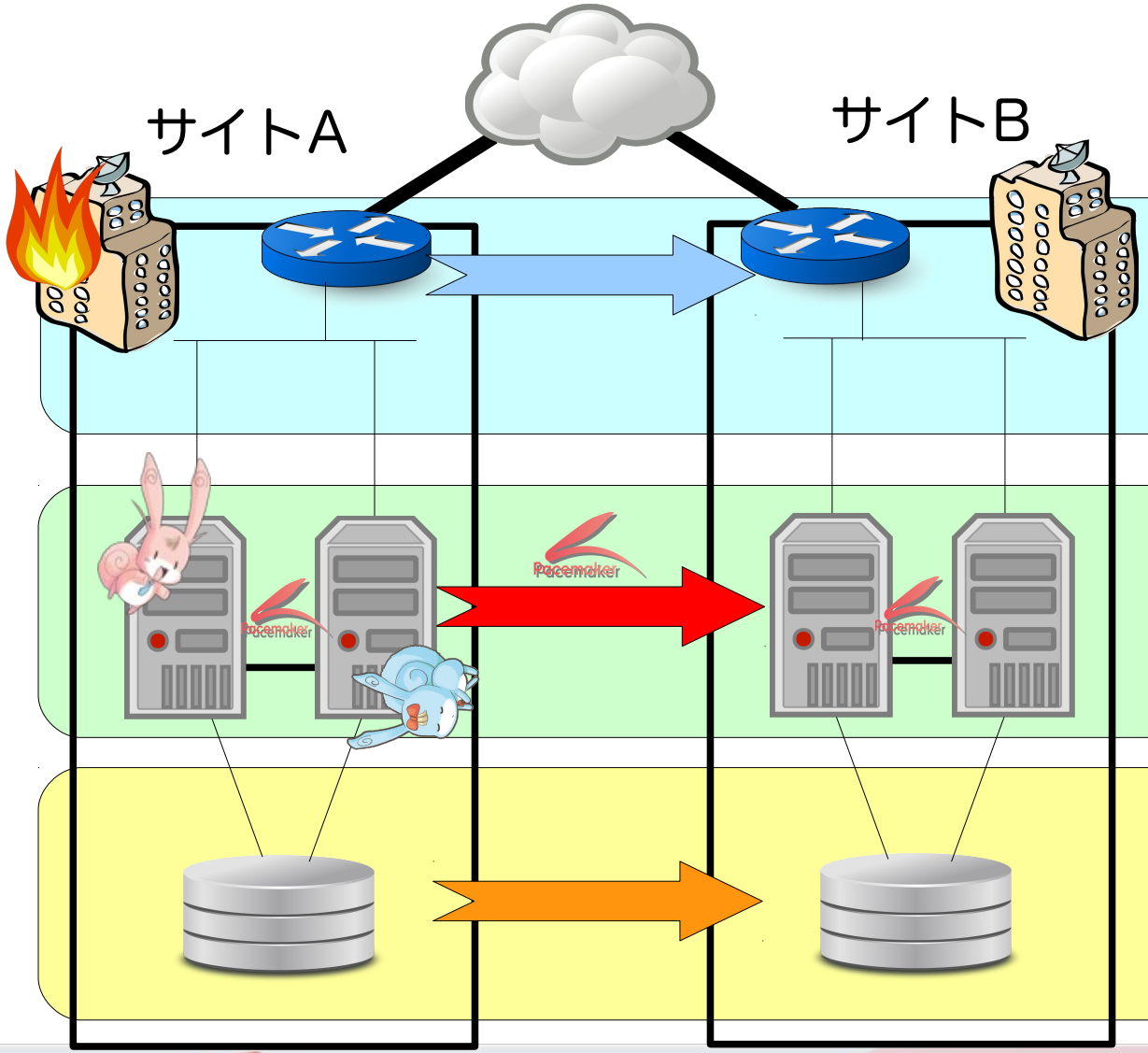
- Pacemaker 安定版
  - Pacemaker-1.0.13 リリース！ (間近)
  - Linux-HA Japan リポジトリパッケージも追随してバージョンアップ予定
  - バグフィックスのみ
- Pacemaker 開発版
  - Pacemaker-1.1.8 リリース！ (間近)
  - Pacemaker-1.1.6 は既にいくつかのディストリビューションで採用実績あり
  - Linux-HA Japan としては…
    - 現状では安定性に難ありと判断
    - 既知の不具合あり(主にSTONITH関連)。コードの修正がまだ安定していない。etc.
- Pacemaker 「次期バージョン」
  - Pacemaker-1.2.0 をLinux-HA Japanの次期バージョンと位置づけます。
  - 本発表では、1.1系以降および1.2.0で新たに利用可能となる機能を紹介します。

Pacemaker次期バージョンにおけるトピック

- ディザスタリカバリ対応機能
  - booth / ticket 機能

- Red Hat HA Add-on 統合に向けた機能
- Corosync-2.0対応
- クラスタ制御機能

ディザスタリカバリにおけるPacemakerの位置づけ



ネットワークの切替

- DNS切替
- NW仮想化(OpenFlow)

サービスの切替

- Pacemaker
- HAクラスタ製品
- 自動切替/手動切替

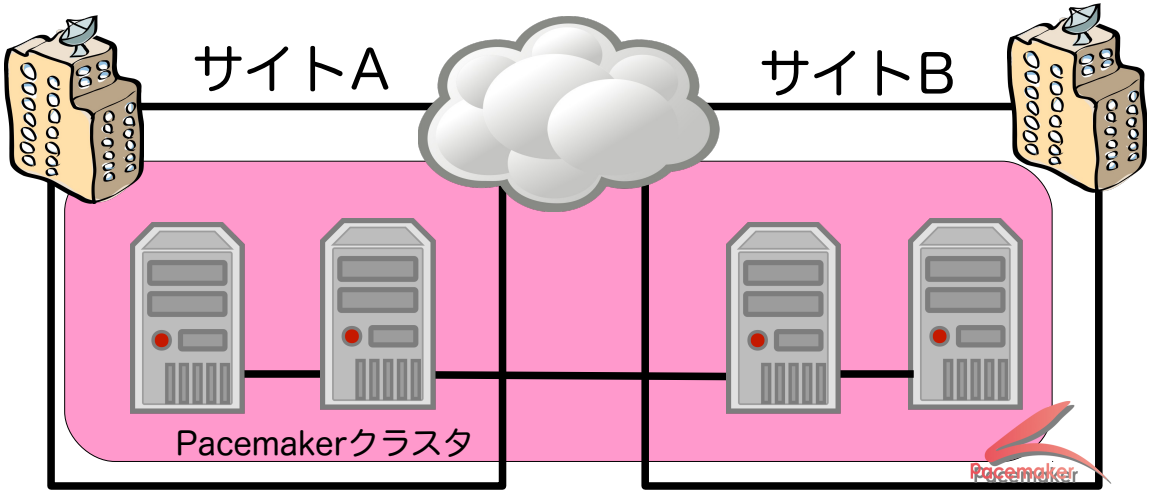
データの複製

- DRBD
- ストレージ製品
- バックアップ製品
- 同期/非同期/オフライン

ストレッチクラスタとスプリットサイト

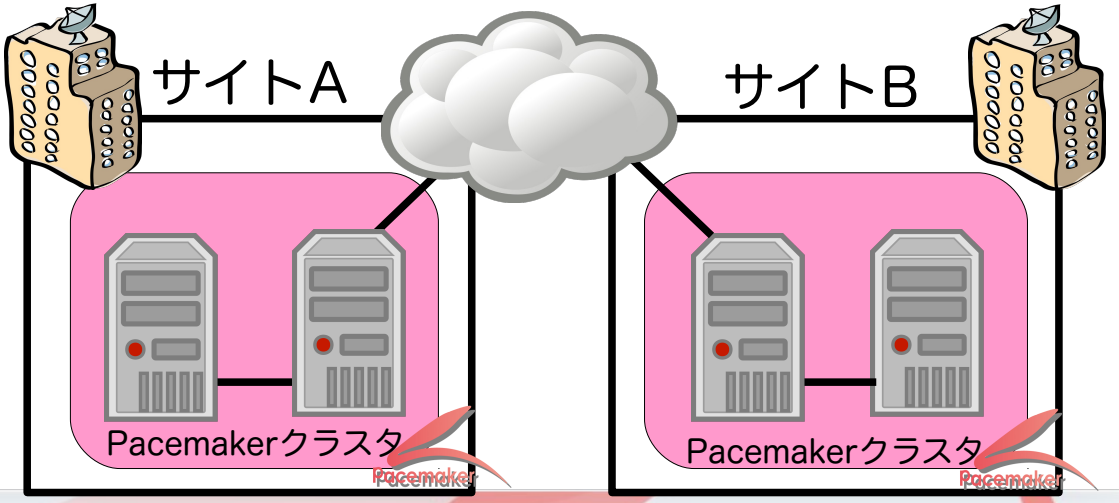
ストレッチクラスタ

- 従来のHAクラスタの単純な延長
- サイト間通信遅延大
  サイト内切替時間へも影響
- 全サイトのノードを意識した設定・運用の複雑化
- サイト間を超えたSTONITHが不可



スプリットサイトクラスタ(マルチサイトクラスタ)

- サイト内でのクラスタとサイト間での運用を分離
  (同一サイト内ではほぼ従来通りの設定・運用が可能)
- サイト間通信遅延の影響を抑える
  (遠隔地間に適した通信)
- サイト間通信切断時はサイト内でSTONITH実行



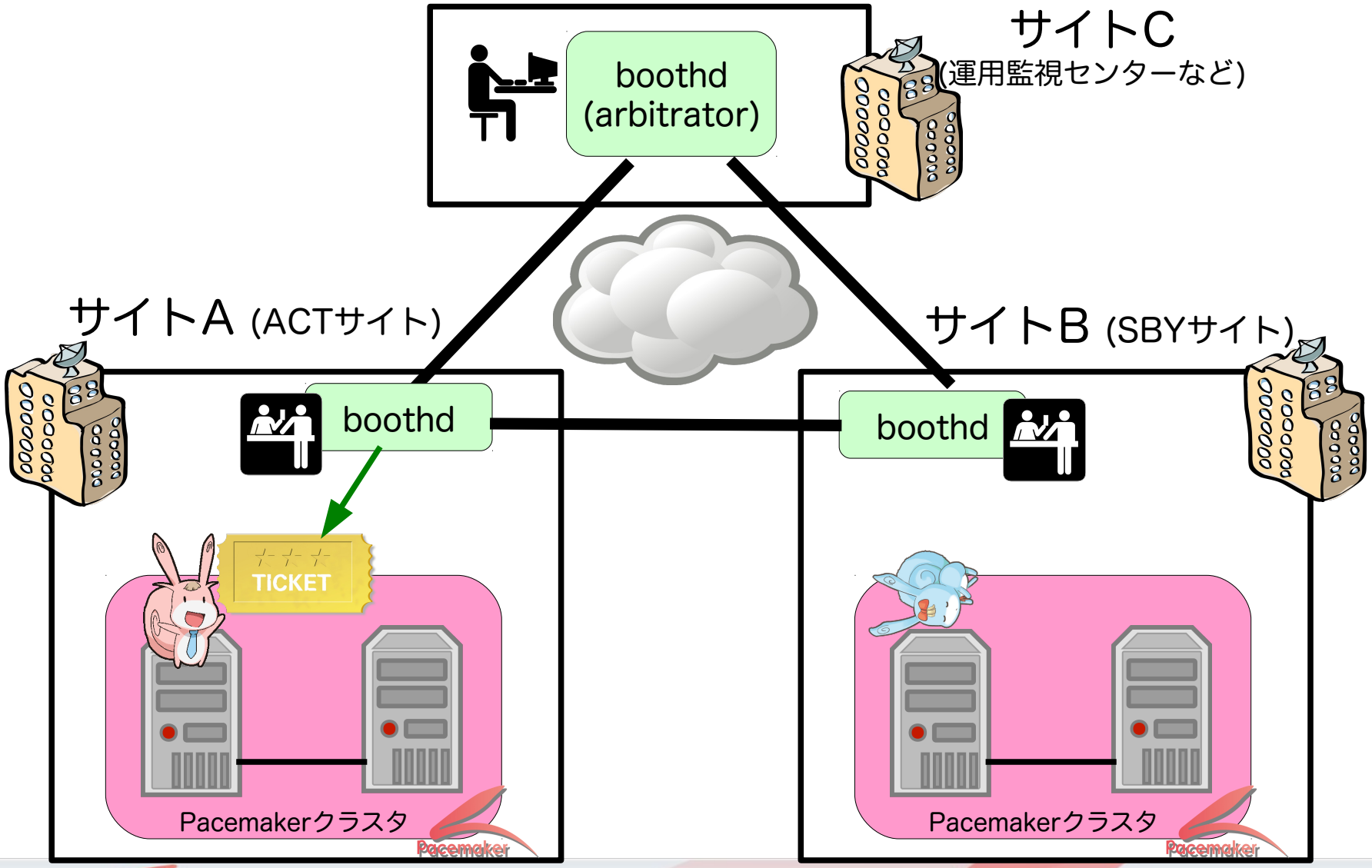
booth 機能 / ticket 機能

- Pacemaker で「スプリットサイトクラスタ」を実現するための追加機能

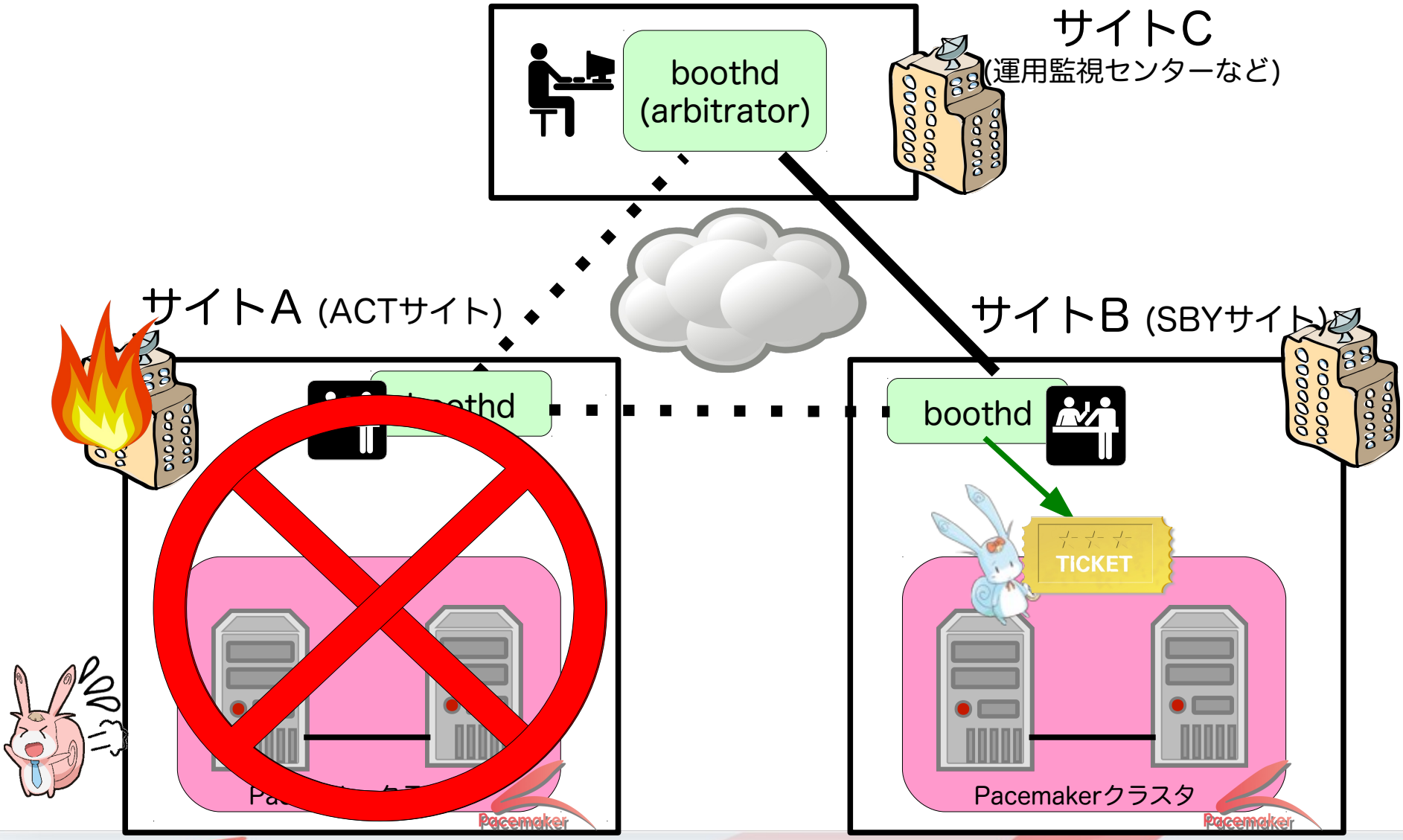
- ticket機能
  - 「チケット」の有無により、クラスタ単位でサービスの有効化を制御する機能。
  - Pacemaker 本体の追加機能 (Pacemaker-1.1.6 以降で利用可能)

- booth機能
  - サイト間で通信を行い、サービスを稼働すべきサイトのPacemakerクラスタへ「チケット」を付与する機能。
  - Pacemakerの外部モジュールとして動作し、別パッケージとして開発・リリースされている。
    - ソースコードは github 上で公開(Pacemakerと同じClusterLabsプロジェクト)。
    - 最新リリース: booth-0.1.0 (2012年5月)

booth / ticket 機能アーキテクチャ



booth / ticket 機能アーキテクチャ



booth / ticket 機能設定例

- 1.boothd設定例: /etc/sysconfig/booth ファイル

```
transport="UDP"
port="6666"
arbitrator="172.16.0.1"
site="172.20.0.2"
site="172.30.0.3"
ticket="ticketA; 1000 "
ticket="ticketB; 1000 "
```

- 2.Arbitrator起動(サイトC)

```
/etc/init.d/booth-arbitrator start
```

- 3.チケットとリソースの依存関係設定

```
crm(live)configure#
rsc_ticket rsc1-req-ticketA ticketA: rsc1 loss-policy="fence"
```

- 4.Boothdリソースグループ設定(サイトA, サイトB)

```
crm(live)configure#
primitive booth-ip ocf:heartbeat:IPaddr2 params ip=”172.20.0.2”
primitive booth ocf:pacemaker:booth-site
group g-booth booth-ip booth
```

booth / ticket 運用コマンド例

Pacemaker

- 1.すべてのサイトのチケットリスト表示

```
booth client list
```

  - 実行結果例1: チケット付与前

```
ticket: ticketA, owner: None, expires: INF
```

  - 実行結果例2: チケット付与後

```
ticket: ticketA, owner: 172.20.0.2, expires: 2012/09/04 20:23:36
```

- 2. サイトへのチケット付与

```
booth client grant -t ticketA -s 172.20.0.2
```

- 3. サイトのチケット取り消し

```
booth client revoke -t ticketA -s 172.20.0.2
```

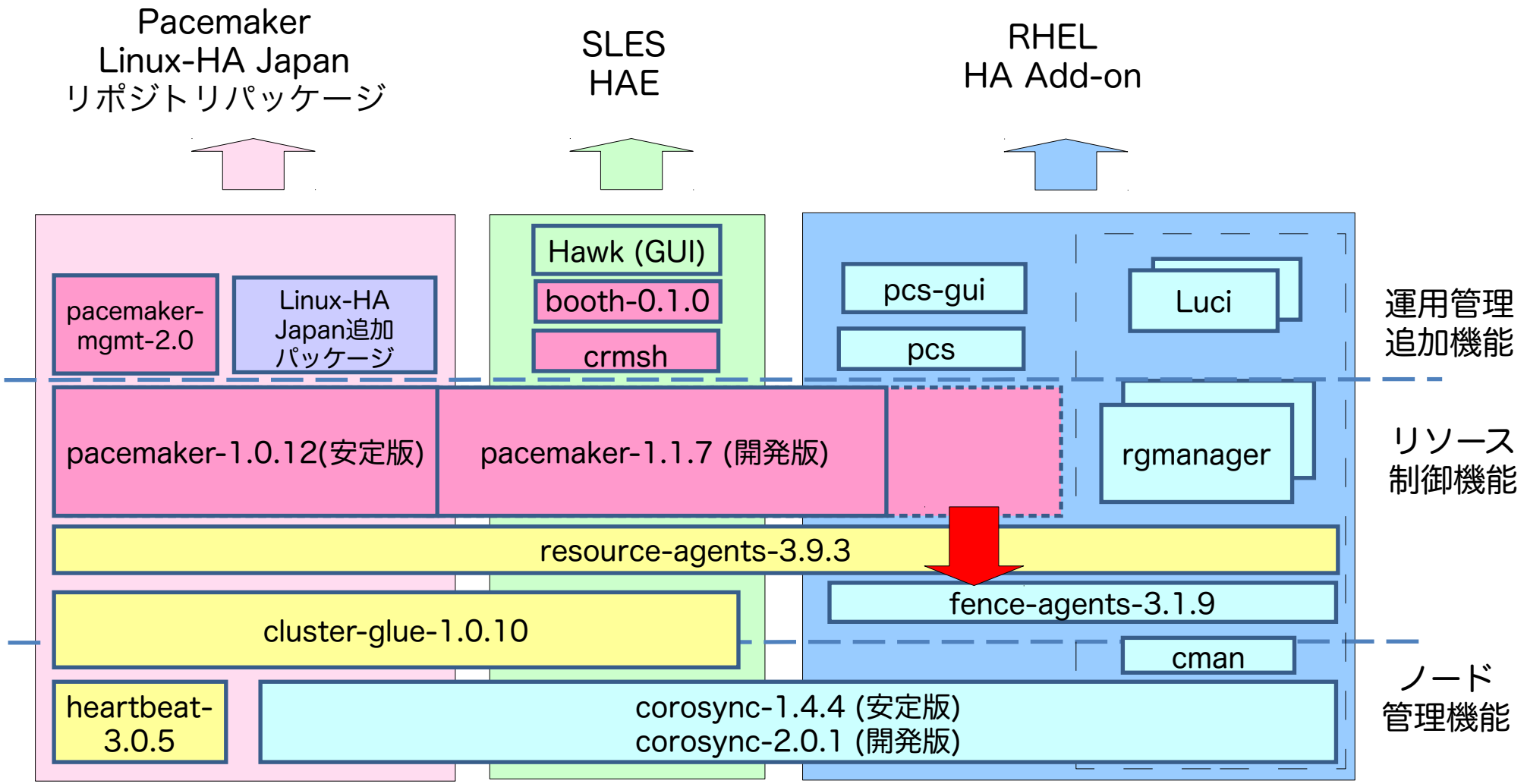
booth / ticket 機能を利用するには…

- 必要なパッケージ、バージョン
  - pacemaker-1.1.6
  - booth-0.1.0
- SUSE Linux Enterprise Server HAE SP2 として利用可能
  - ドキュメントあり
- でも検証してみたらいくつか懸念点が…
  - 主な課題
    - ACT/SBYサイト間の通信断が発生した場合、想定外のサービス再起動もしくはサイト切替が発生する可能性がある。
    - ACTサイト間通信断が発生した場合、ACTサイトサービス停止が完了前にSBYサイトでサービス起動が開始してしまう。
    - ACTサイト内での両系故障の場合、SBYサイトへ自動的に切り替えることができない。
  - 課題については日本からフィードバック済み、現在開発中のバージョンで改善予定
- Pacemaker-1.2のリリースまで待つのがオススメ
  - その時まではboothの次のバージョンもリリースされているはず。

Pacemaker次期バージョンにおけるトピック

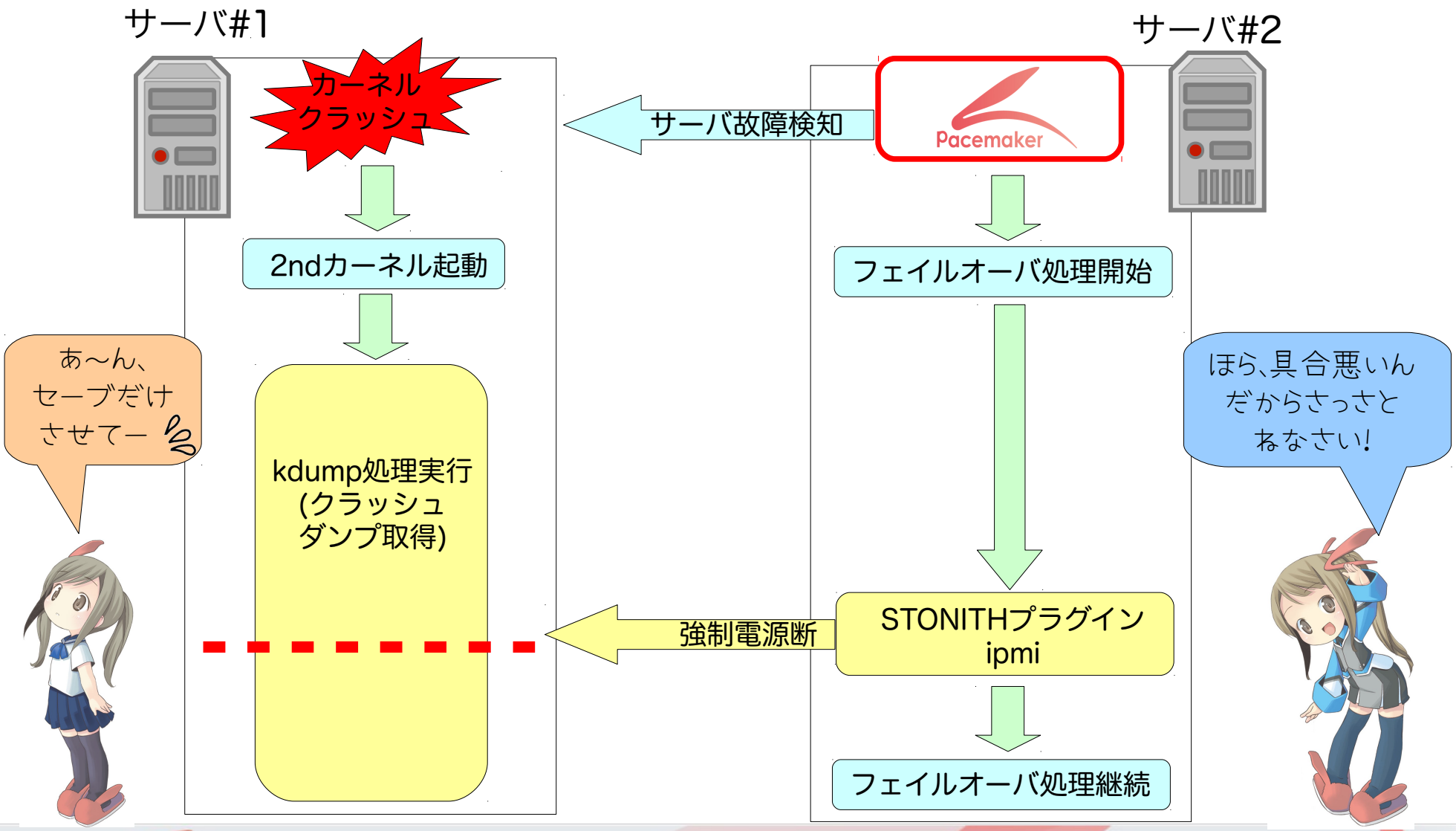
- ディザスタリカバリ対応機能
  - booth / ticket 機能
- Red Hat HA Add-on 統合に向けた機能
  - fence-agents 対応 (STONITHプラグインの共通利用)
    - kdump連携機能
  - lrmd再実装、cman の置き換え (corosync-2.0 quorum機能対応)
  - GUI向け管理ツール: PCS, PCS-GUI (Pacemaker/Corosync configuration system)
    - crmシェルと選択可能に
- Corosync-2.0対応
- クラスタ制御機能

RHEL HA Add-on との統合

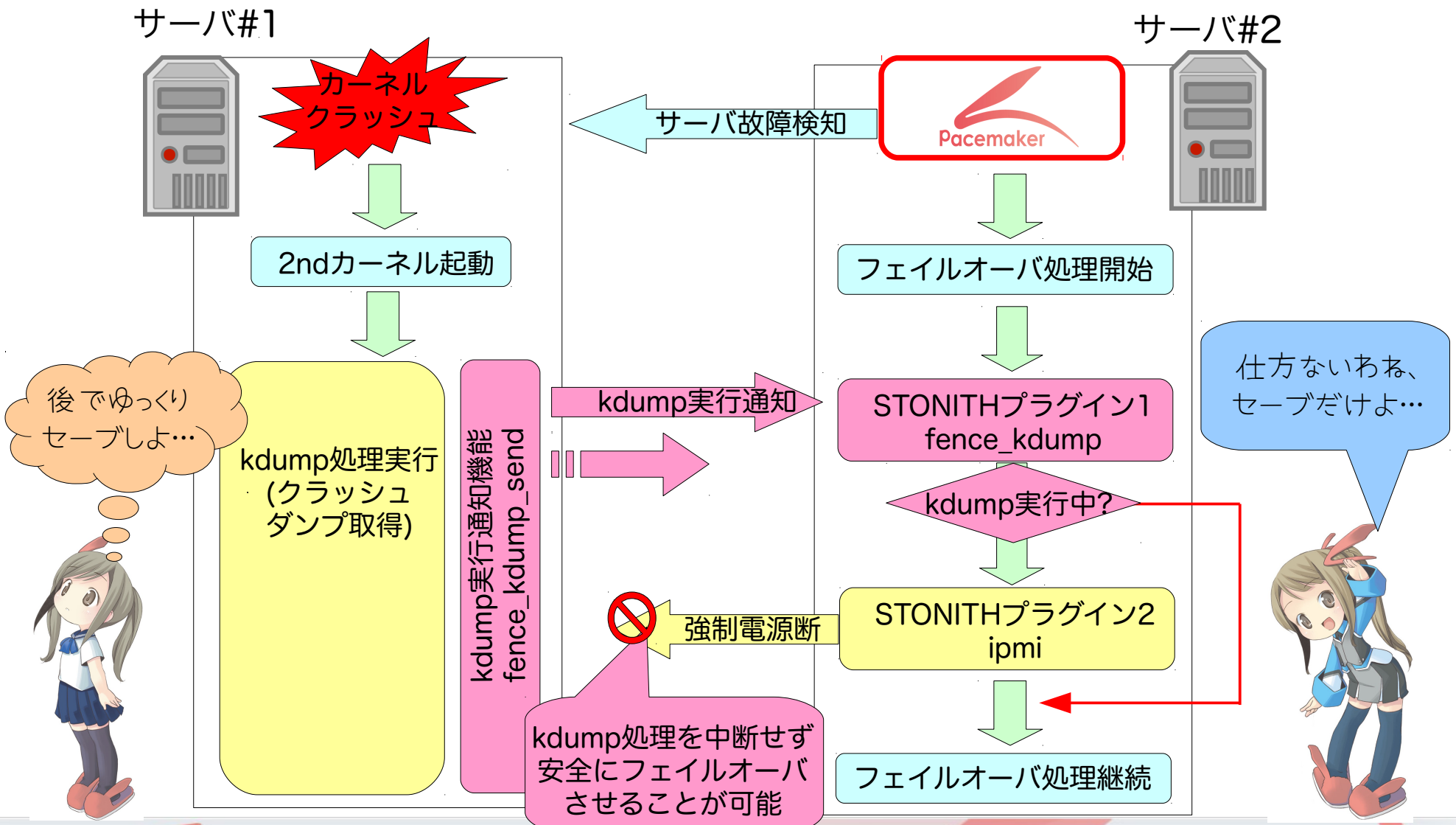


バージョンは2012年9月時点の最新のものを記載しているため、
各製品・パッケージに含まれるバージョンと異なる場合があります。

kdump連携機能 (fence_kdump)



kdump連携機能 (fence_kdump)



kdump連携機能を利用するには…

- 必要なパッケージ・バージョン
  - pacemaker-1.1.8
  - fence-agents-3.1.9 (RHEL HA Add-on パッケージの一部)
  - RHEL6.4 で上記バージョンが含まれる見込み。
- 設定例
  - セカンドカーネルへの組み込み
    - (Red Hatの場合) /etc/cluster/cluster.conf を手動で編集

```
<fencedevice name="kdump" agent="fence_kdump" />
```

  - STONITHリソース例

```
primitive stonith-1 stonith:fence_kdump \
        params pcmk_host_check="static-list" pcmk_host_list="node1" \
        pcmk_reboot_action="off"  pcmk_monitor_action="metadata" \
        nodename=node1  timeout=180
```

- RHEL6 HA Add-on では RHEL6.2 以降で利用可能(のはず)

Pacemaker次期バージョンにおけるその他トピック

- ディザスタリカバリ対応機能
- Red Hat HA Add-on 統合に向けた機能

- Corosync-2.0対応
  - より安定化・切替時間の高速化が可能
    - Heartbeat: 約50秒 → Corosync: 数秒以内という検証例もあり
  - libqb 対応: IPC、ログ出力機能の再実装

- クラスタ制御機能
  - Utilization制御機能: ノードの「容量」に合わせたリソース配置
    - CPU数・メモリ量上限の制御
    - N+1/N+M構成(稼働リソースの相乗りを抑止)
  - 設定変更ACL制御、stonith-topology, or条件のcolocation, order
  - ログ量の改善(かなり減ってる)

Pacemakerの今後

- クラウドとの関係は?

OpenStack コンポーネントの冗長化

- Glance / Keystone の冗長化
  - Pacemakerによる冗長化のため、3つのRAを開発中:
    - glance-api, glance-registry, keystone
- MySQL の冗長化
  - DRBD + Pacemaker もしくは MySQLマルチマスタ機能による冗長化
- RabbitMQ の冗長化
  - DRBD + Pacemaker (ただし deprecated)
  - 今後は RabbitMQ内で Active/Activeキューを実装
- nova-network の冗長化
  - 複数実装案あり、HAクラスタも一つのオプション

出典:

- https://blueprints.launchpad.net/ubuntu/+spec/servercloud-q-openstack-ha
- http://docs.openstack.org/trunk/openstack-compute/admin/content/existing-ha-networking-options.html

クラウド環境におけるゲストのサービス監視

- 過去のPacemaker派生プロジェクト
  - Matahari
    - ホスト側からゲスト内のサービス監視・制御を行うフレームワーク
  - Pacemaker-cloud
    - ゲスト内のサービス監視・再起動・ゲスト再起動へのエスカレーション
    - Matahari と連携
  - しかし、いずれのプロジェクトも既に活動停止
- 今後のクラウド環境に向けた新たなHA関連技術
  - Heat API
    - OpenStack 版 AWS CloudFormation API の実装。
    - ゲスト内のサービス監視・再起動・ゲスト再起動へのエスカレーション
    - Pacemaker-cloud 後継の位置づけ
      - » http://wiki.openstack.org/Heat
  - oVirt
    - ノード故障・VM故障に対するHA機能は実装済み。
    - ovirt-guest-agentによるゲスト内サービス監視が実装可能(Mataharと同様機能)
      - ただし現時点では具体的な実装はなし。今後に期待
        - » https://events.linuxfoundation.org/images/stories/pdf/lcjp2012_azulay.pdf
        - » http://www.ovirt.org/w/images/2/20/Ovirt-guest-agent.pdf

Heat API 利用例

- WordPress_Single_Instance_With_HA.template (抜粋)

```
{
  "AWSTemplateFormatVersion" : "2010-09-09",
  "Description" : "AWS CloudFormation Sample Template WordPress_Multi_Instance:(...).",

(…)
    "WebServerRestartPolicy" : {
      "Type" : "HEAT::HA::Restarter",
      "Properties" : {
        "InstanceId" : { "Ref" : "WikiDatabase" }
      }
    },
    "HttpFailureAlarm": {
     "Type": "AWS::CloudWatch::Alarm",
     "Properties": {
        "AlarmDescription": "Restart the WikiDatabase if httpd fails > 3 times in 10 minutes",
        "MetricName": "ServiceFailure",
        "Namespace": "system/linux",
        "Statistic": "SampleCount",
        "Period": "300",
        "EvaluationPeriods": "1",
        "Threshold": "2",
        "AlarmActions": [ { "Ref": "WebServerRestartPolicy" } ],
        "ComparisonOperator": "GreaterThanThreshold"
      }
    },
(以下略)
```

httpdサービスが故障したら再起動
10分以内に3回以上故障したらサーバ再起動

Pacemakerの今後

- クラウドとの関係は?

- 物理サーバの冗長化は無くならない
  - クラウドインフラストラクチャそのものの冗長化
    - OpenStack コンポーネントの冗長化
  - クラウドが適さないシステム

- ゲスト内のサービス監視
  - クラウド環境に向けた新たなHA技術
    - Heat, oVirt : 今後は要注目、でも現在は発展途上
  - クラウド環境とHAクラスタは補完関係
    - クラウド環境だけでは対応できない故障をHAクラスタで補完

おわり

- 以上です。ありがとうございました！