

ハイブリッドクラウド対応クラウドコントローラ

Prime Cloud Controller

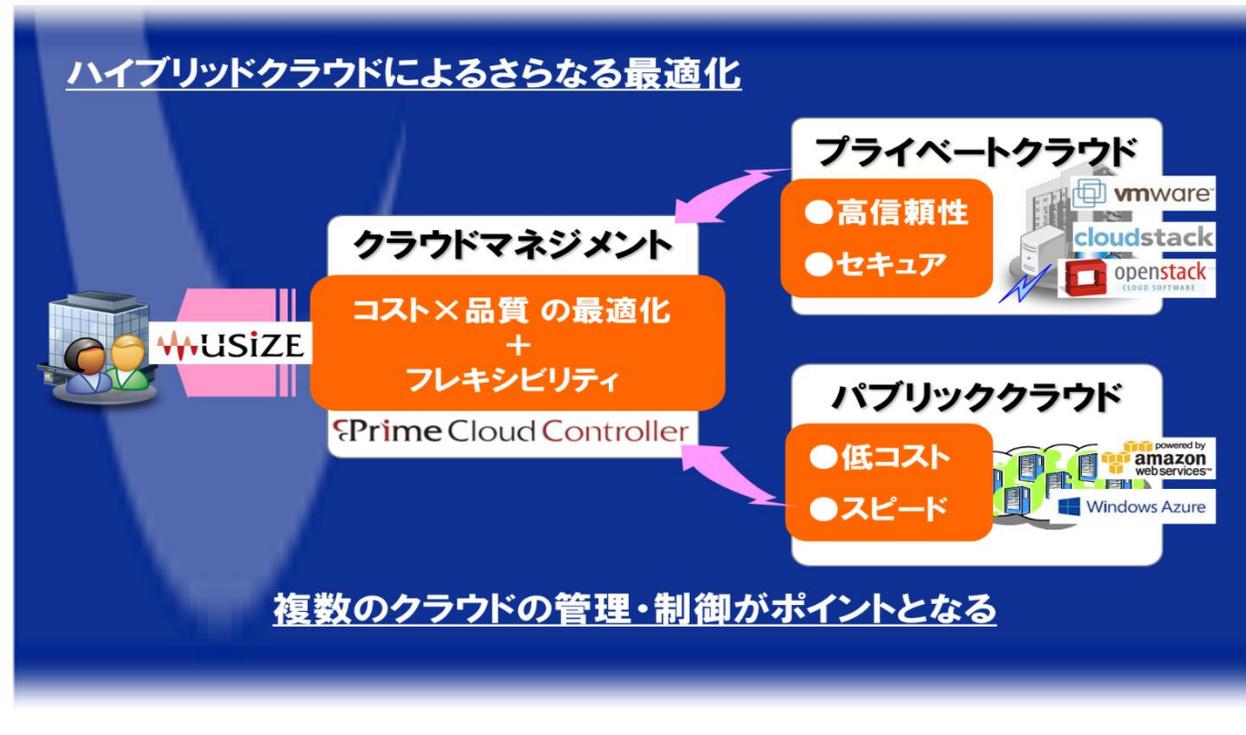
2014/7/17

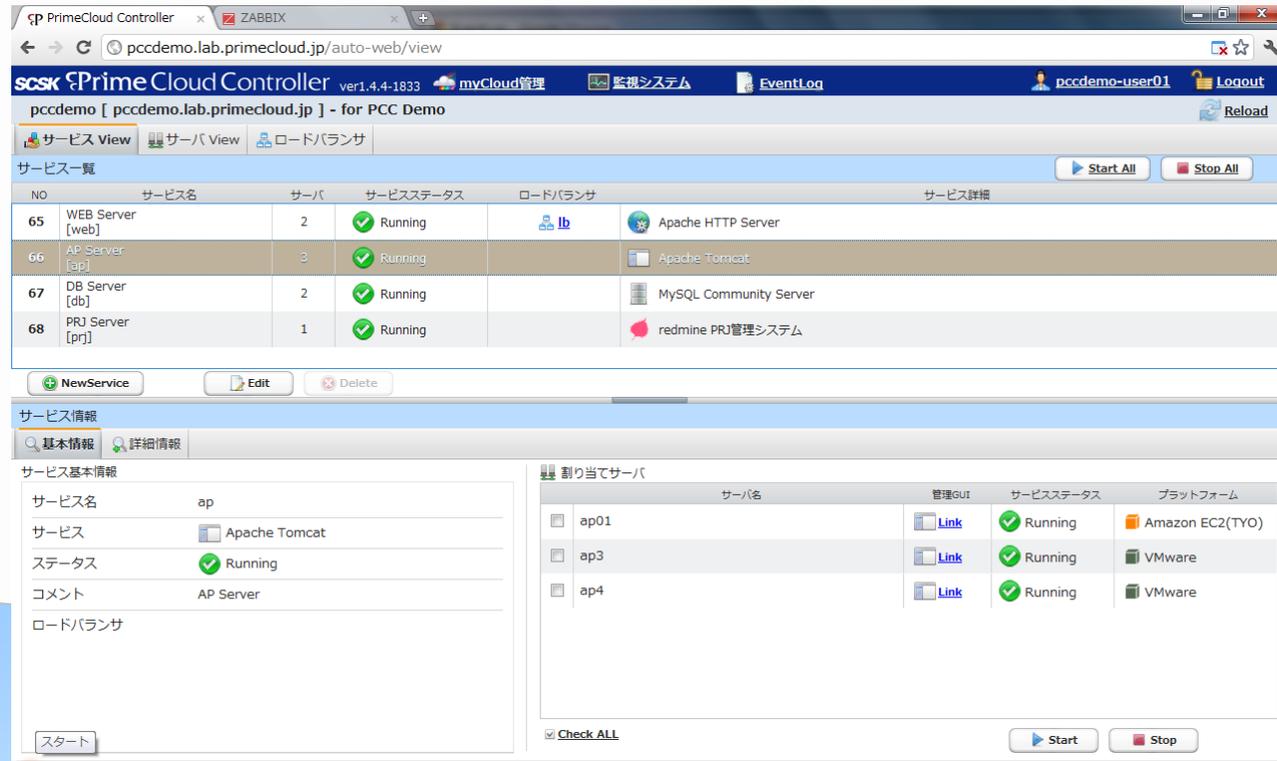
SCSK CORPORATION

1. PrimeCloud Controllerとは？
2. 動作の概要・仕組み
3. 近日リリース予定の新機能
4. インストール方法
5. まとめ

1. PrimeCloud Controllerとは？
2. 動作の概要・仕組み
3. 近日リリース予定の新機能
4. インストール方法
5. まとめ

- クラウドは当たり前前の時代。
- ビジネスを変革させる手段としてとらえられている。
- プライベートクラウドとパブリッククラウドを統合管理するハイブリッドクラウドが着目されている。





さまざまなクラウド環境を
統一的に管理可能

Microsoft Azure
OpenStack



VMWare
・USiZE SharedModel

Amazon Web Services

NIFTY Cloud

(CloudStack)
IDCフロンティア
NTTCom Cloud N
USiZE netXCloud

Prime Cloud Controller

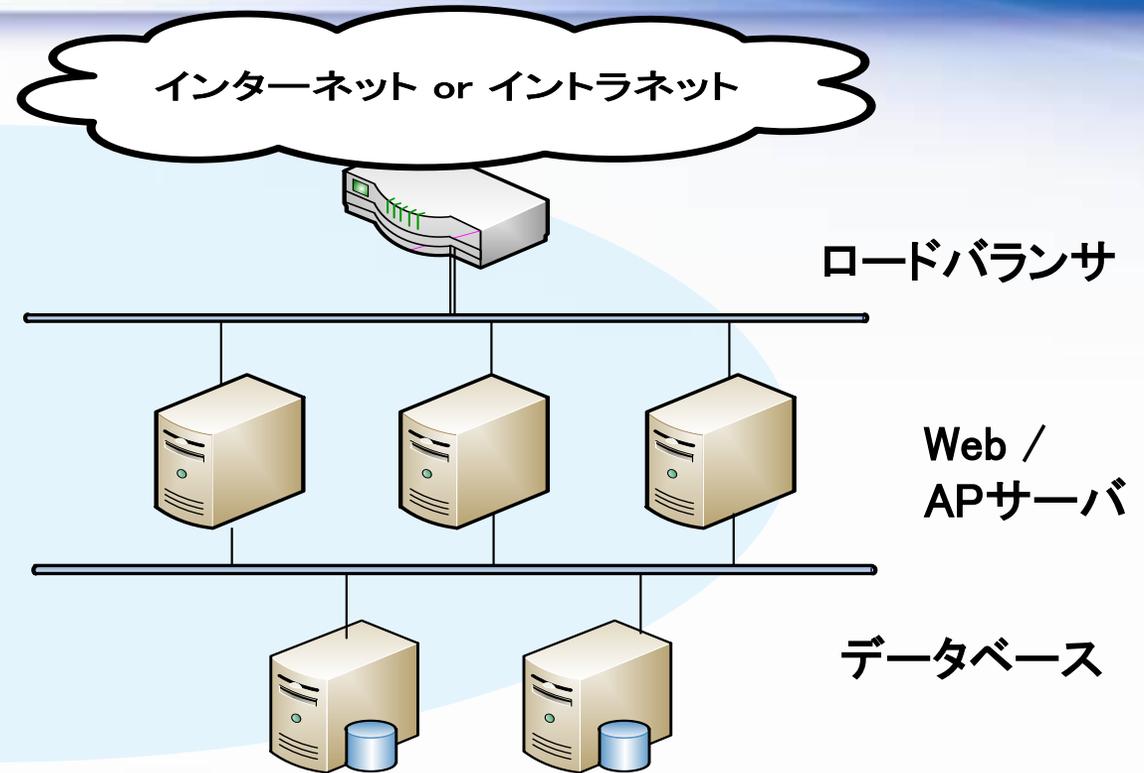
USIZE, Amazon Web Services, NIFTY Cloud, Eucalyptus, vCenter, vCloud, Cloudstack対応

The screenshot displays the Prime Cloud Controller web interface. At the top, there's a navigation bar with 'scsk Prime Cloud Controller ver1.4.4-1833', 'myCloud管理', '監視システム', and 'EventLog'. Below this, a breadcrumb shows 'pccdemo [pccdemo.lab.primecloud.jp] - for PCC Demo'. The main content area is titled 'サービス一覧' (Service List) and contains a table with columns: NO, サービス名 (Service Name), サーバ (Server), サービスステータス (Service Status), ロードバランサ (Load Balancer), and サービス詳細 (Service Details). The table lists several services, including 'WEB Server [web]', 'AP Server', and 'redmine PRJ管理システム', all with a 'Running' status. A modal window titled '+ 新規サーバをサービスに追加' (Add new server to service) is open, showing a form with 'サーバ名 (Prefix)*' set to 'ap' and 'サーバ台数*' (Number of servers) set to '1'. A 'Delete' button is visible at the bottom of the modal.

NO	サービス名	サーバ	サービスステータス	ロードバランサ	サービス詳細
65	WEB Server [web]	2	Running	lb	Apache HTTP Server
66	AP Server	2	Running		Apache Tomcat
			Running		MySQL Community Server
			Running		redmine PRJ管理システム

- 簡単なGUI操作で、プライベート/パブリッククラウドを統一的に制御することが可能。
- サーバ起動/停止だけでなく、複雑な設定なども自動化。
→ 仮想化によるサーバ統合だけでなく、運用の効率化が可能

ミドルウェアのインストールや設定、動作確認、監視など、システム全体の自動化を実現。



- ・ PrimeCloud Controllerを活用した場合 = **20分**
ミドルウェアの設定や動作確認、クラウド間のVPN接続、操作ログの記録
監視サーバへのノード追加も自動で対応

1. ハイブリッドクラウド対応

プライベートクラウド : VMWare, Eucalyptus, CloudStack(Xen, ESX, KVM)

パブリッククラウド :

AWS, NIFTY Cloud, IDCフロンティア, NTTCom Cloud n, USiZE

Microsoft Azure, OpenStack(Havana)

2. Web, AP, DBなどミドルウェアなども自動設定

3. さまざまなクラウドで統一的な操作

4. 他のクラウドもまとめてZabbixで一元管理

- 監視ノード/監視対象サービスの自動追加と削除

5. 他のクラウドに必要なに応じて自動でVPN接続

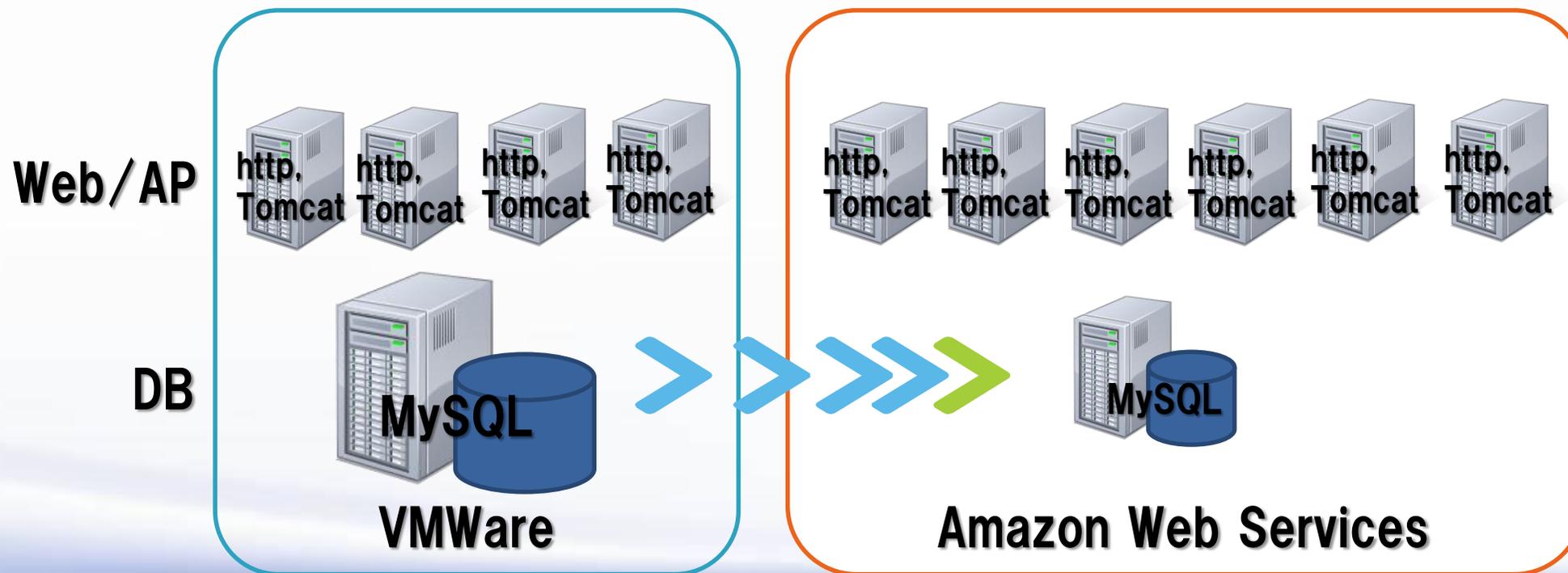
1. ハイブリッドクラウド対応

プライベートクラウド : VMWare, Eucalyptus, CloudStack(Xen, ESX, KVM)

パブリッククラウド :

AWS, NIFTY Cloud, IDCフロンティア, NTTCom Cloud n, USiZE

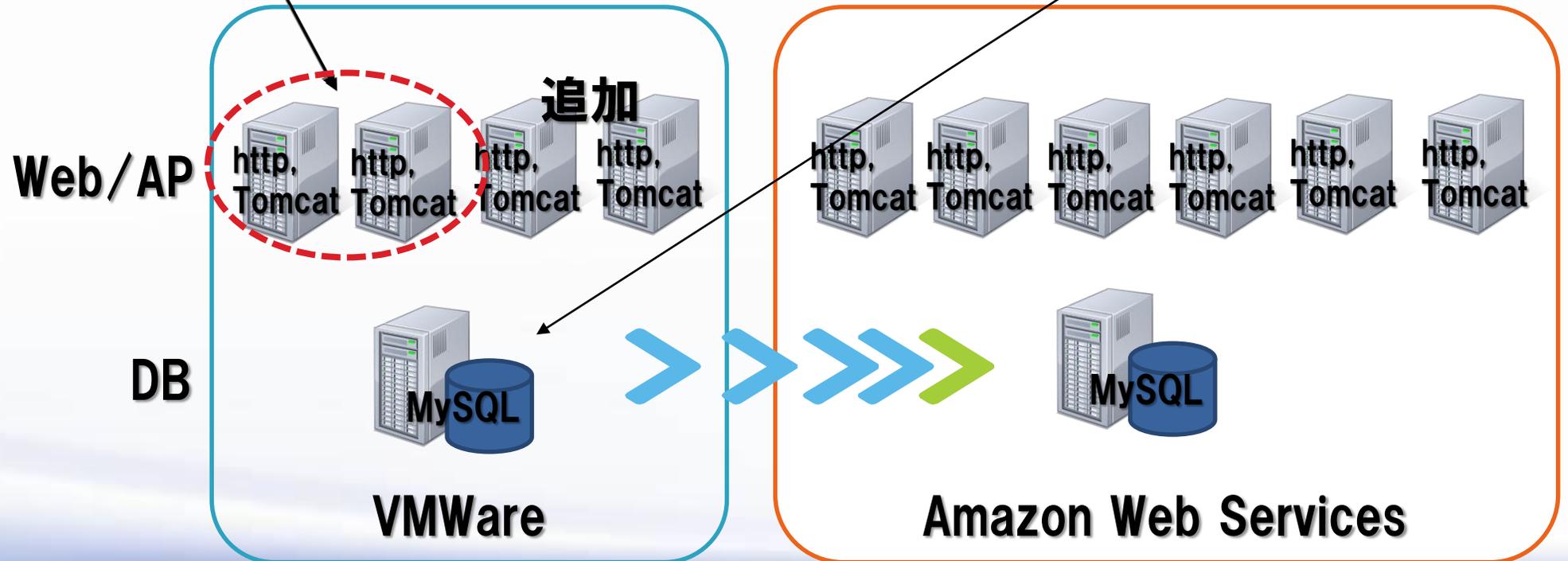
Microsoft Azure, OpenStack(Havana)



2. Web, AP, DBなどミドルウェアなども自動設定

既存のサーバも必要に応じて、
自動で設定変更

影響を受けるサーバも自動変更



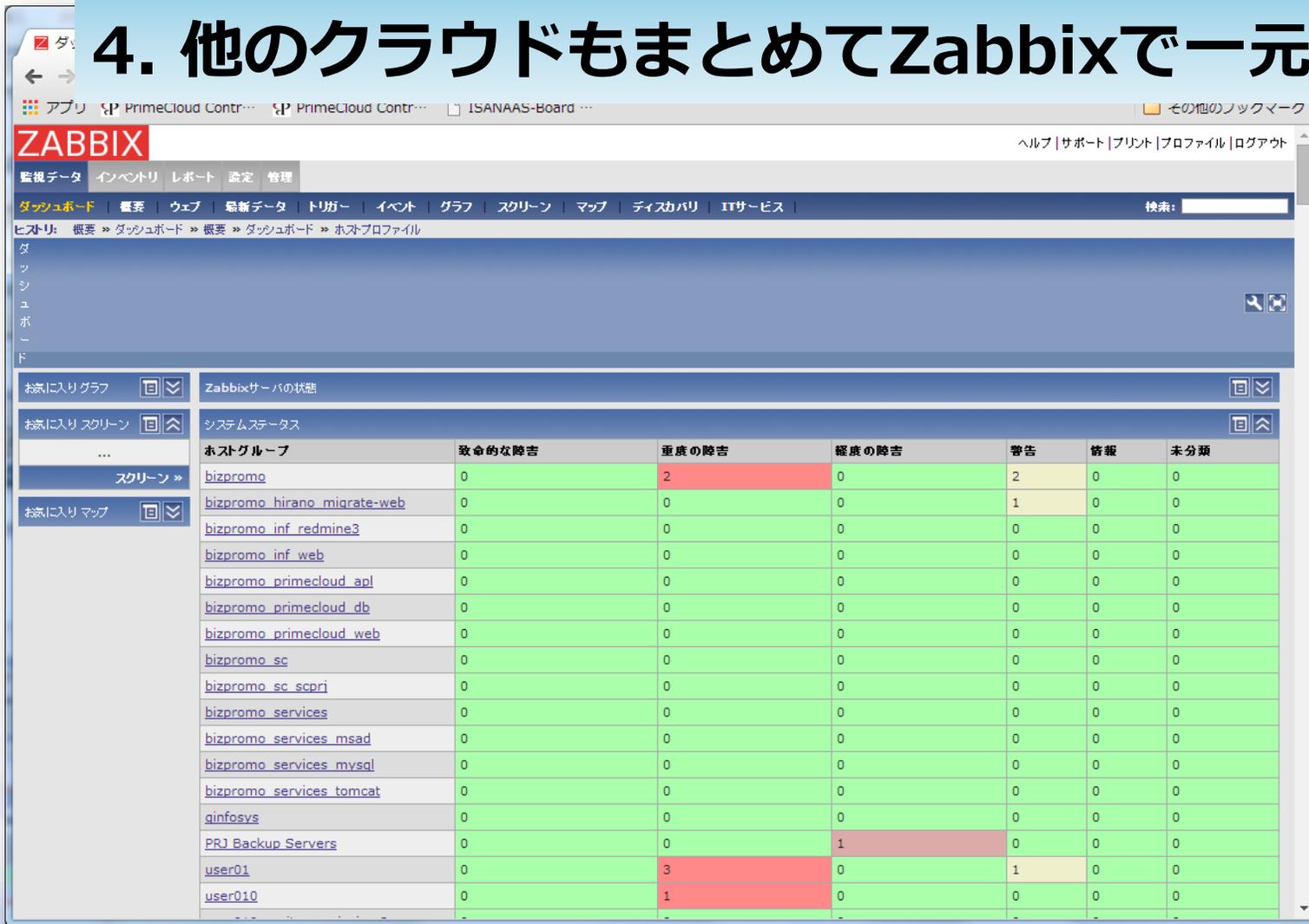
3. さまざまなクラウドで統一的操作

The screenshot displays the PrimeCloud Controller web interface. The main table lists services with columns for NO, Service Name, Servers, Service Status, Load Balancer, and Application. A 'サービス追加' (Add Service) dialog is open, showing a form for 'サービス名' (Service Name) set to 'apl', 'コメント' (Comment) set to 'アプリケーションサーバ', and a list of services including MySQL Community Server, Apache Tomcat, Apache HTTP Server, and a development project server. A '新規サーバをサービスに追加' (Add New Server to Service) dialog is also open, showing a list of platforms like Eucalyptus Private Cloud, Amazon EC2, Amazon EC2 + VPC, VMWare vSphere 4, and ニフティクラウド (Nifty Cloud), with 'サーバ台数' (Number of Servers) set to 1.

NO	サービス名	サーバ	サービスステータス	ロードバランサ	アプリケーション
65	WEB Server [web]	2	Running	lb	Apache HTTP
66	AP Server [ap]	3	Running		Apache Tomcat
67	DB Server [db]	2	Running		MySQL Community Server
68	PRJ Server [prj]	1	Running		redmine PRJ

操作ログも取れるので、内部統制対応・監査対応も容易

4. 他のクラウドもまとめてZabbixで一元管理



The screenshot shows the Zabbix web interface with a table titled 'Zabbixサーバの状態' (Zabbix Server Status). The table has columns for '致命的な障害' (Critical Issues), '重度の障害' (Severe Issues), '軽度の障害' (Minor Issues), '警告' (Warnings), '情報' (Information), and '未分類' (Unclassified). The table lists various host groups and their corresponding issue counts.

ホストグループ	致命的な障害	重度の障害	軽度の障害	警告	情報	未分類
bizpromo	0	2	0	2	0	0
bizpromo_hirano_migrate-web	0	0	0	1	0	0
bizpromo_inf_redmine3	0	0	0	0	0	0
bizpromo_inf_web	0	0	0	0	0	0
bizpromo_primecloud_apl	0	0	0	0	0	0
bizpromo_primecloud_db	0	0	0	0	0	0
bizpromo_primecloud_web	0	0	0	0	0	0
bizpromo_sc	0	0	0	0	0	0
bizpromo_sc_scprj	0	0	0	0	0	0
bizpromo_services	0	0	0	0	0	0
bizpromo_services_msad	0	0	0	0	0	0
bizpromo_services_mysql	0	0	0	0	0	0
bizpromo_services_tomcat	0	0	0	0	0	0
qinfosys	0	0	0	0	0	0
PRJ_Backup_Servers	0	0	1	0	0	0
user01	0	3	0	1	0	0
user010	0	1	0	0	0	0

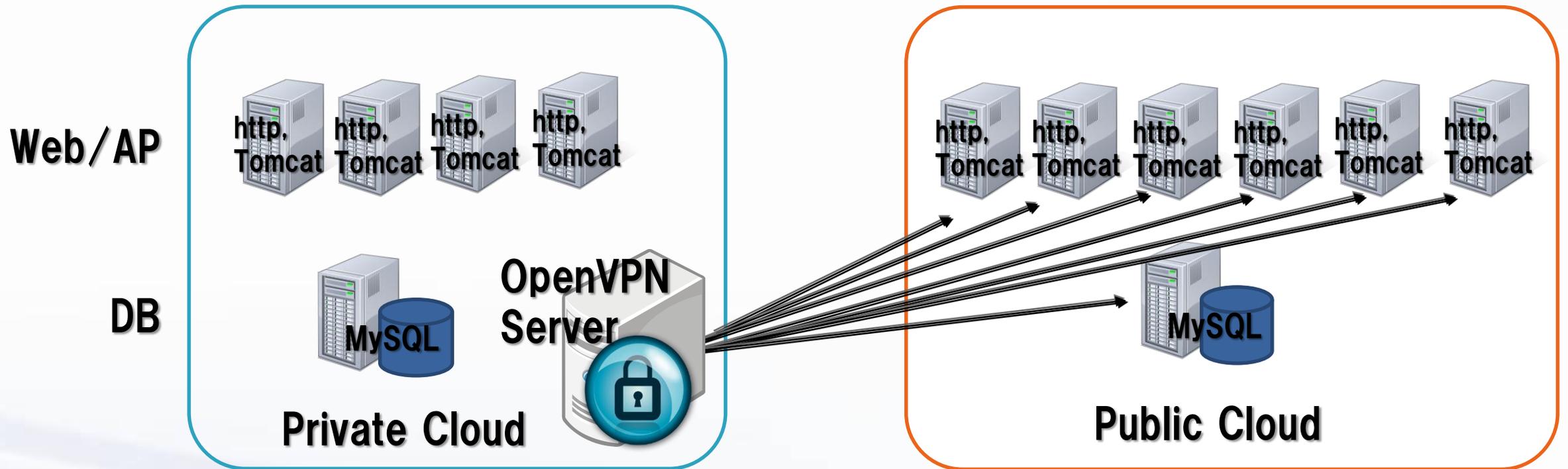
選択したテンプレートに応じて、監視項目を変化させている。

たとえば)

- Webサーバ

Apacheプロセス,
ログ,
CPU, Mem, Disk
ping, 応答速度

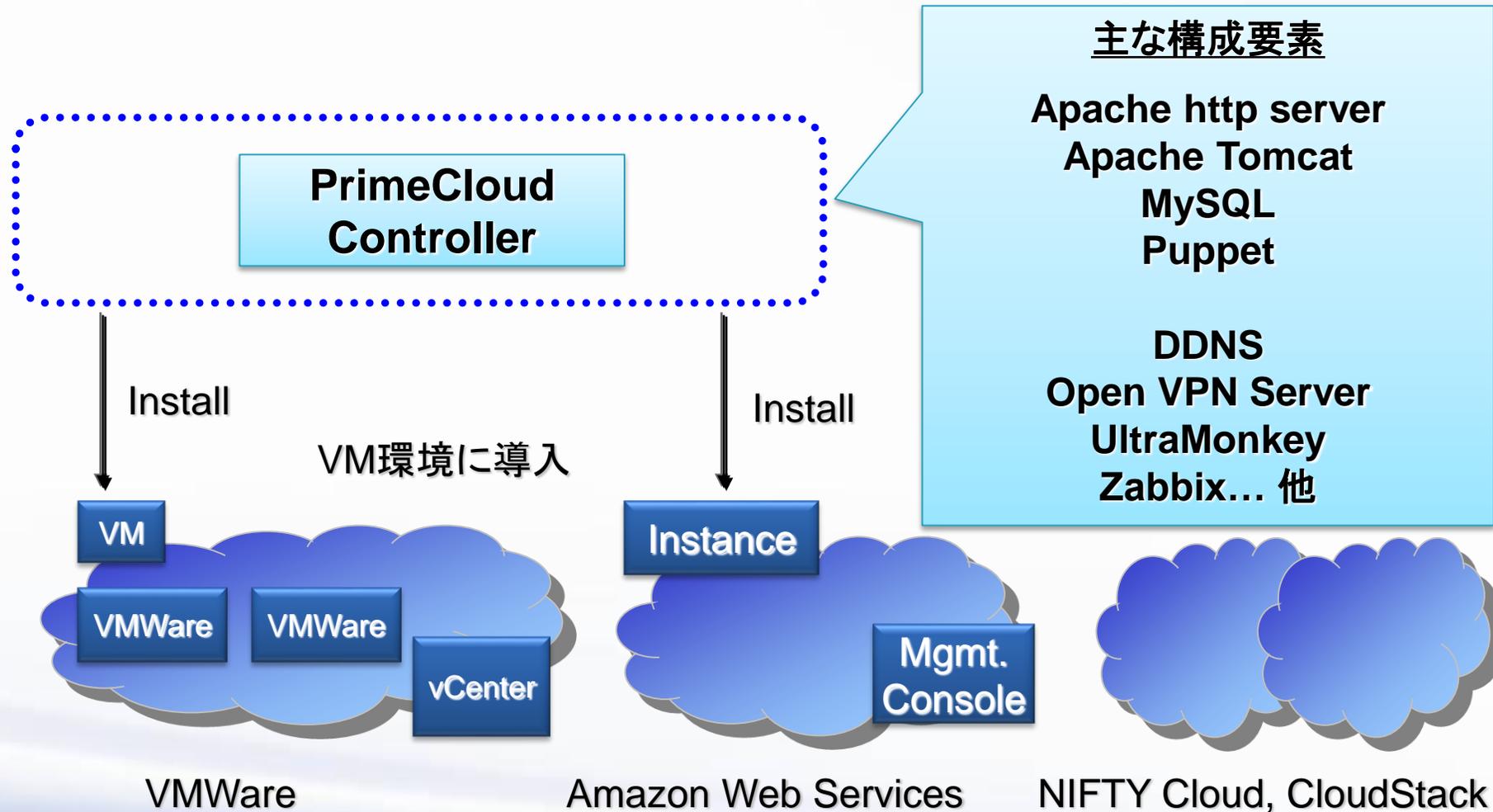
5. 他のクラウドに必要なに応じて自動でVPN接続

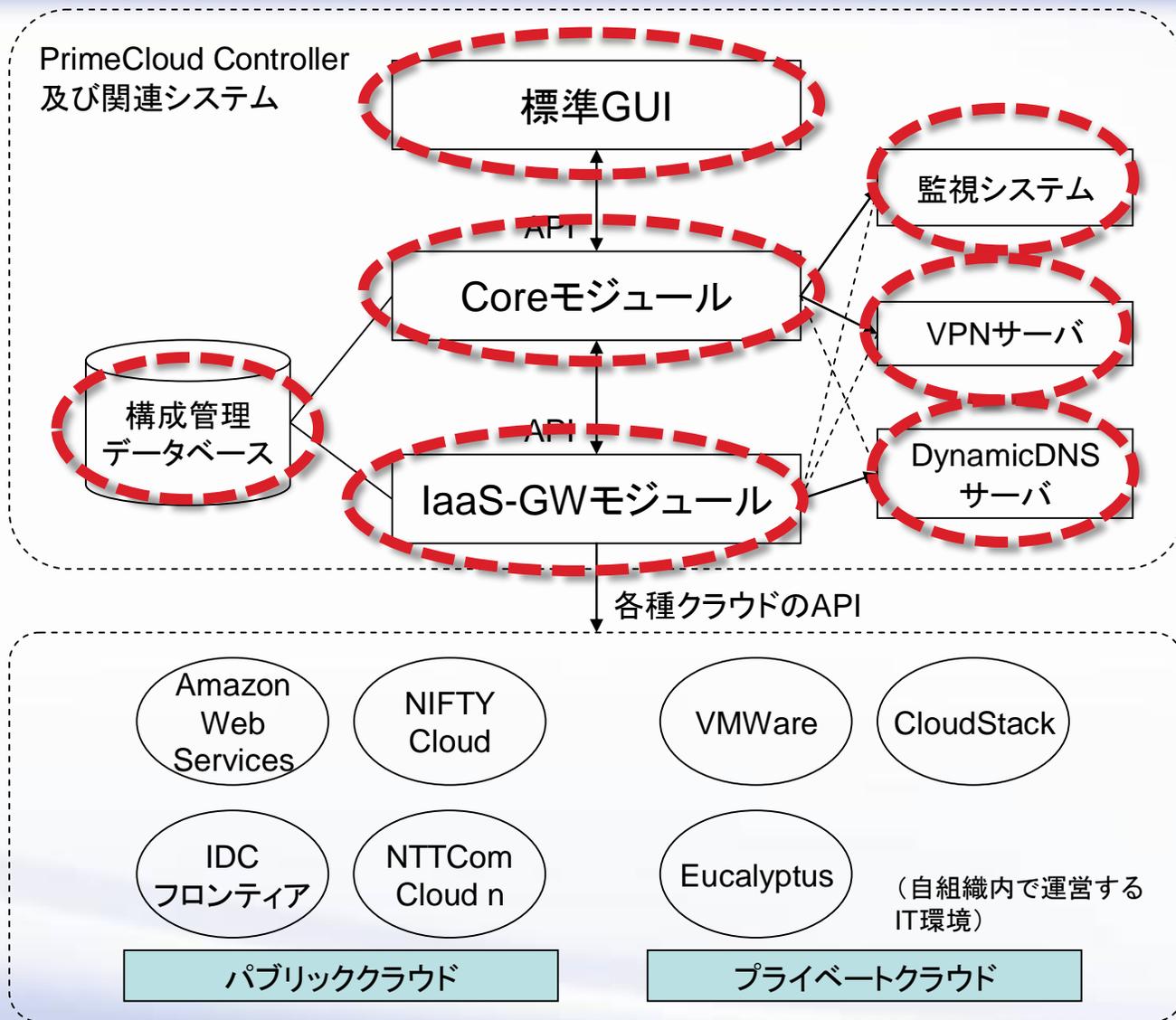


不要になれば、VPN設定も削除

1. PrimeCloud Controllerとは？
- 2. 動作の概要・仕組み**
3. 近日リリース予定の新機能
4. インストール方法
5. まとめ

PrimeCloud ControllerのVMイメージを自所有のクラウド環境に導入





GUIフレームワーク | Vaadin

自動化制御 | puppet

IaaS制御 (一部) | libcloud

構成管理データベース | MySQL

監視システム | Zabbix

VPNサーバ | OpenVPN

Dynamic DNS | bind, dnsmask

- ① GUIモジュール（ユーザインターフェイス部）
 - WEBアプリによるGUIモジュール
 - 言語はJava、VaadinというGWTをWrapするライブラリを使用して記述

- ② PCC-Coreモジュール（自動化制御部）
 - PCC制御用モジュール
 - 画面からの操作を受け、IaaS-GWモジュールに起動指示を出す

- ③ IaaS-GWモジュール（ハイブリッドクラウド制御部）
 - インスタンスを制御するためのモジュール
 - ハイブリッド制御のためにLibcloudを利用
 - Libcloudに移行しつつあるが、一部Javaで記述されている

- クラウドインフラの制御
 - ✓ 仮想サーバの作成、仮想ディスクの作成など
- 仮想サーバ内のOSやミドルウェアの設定
 - ✓ ホスト名やネットワーク設定
 - ✓ サーバ間やミドルウェア間の連携設定
- DNSサーバへの登録
- 監視サーバへの登録
- syslogの転送設定
- VPN接続

インスタンスの主な起動処理の流れ

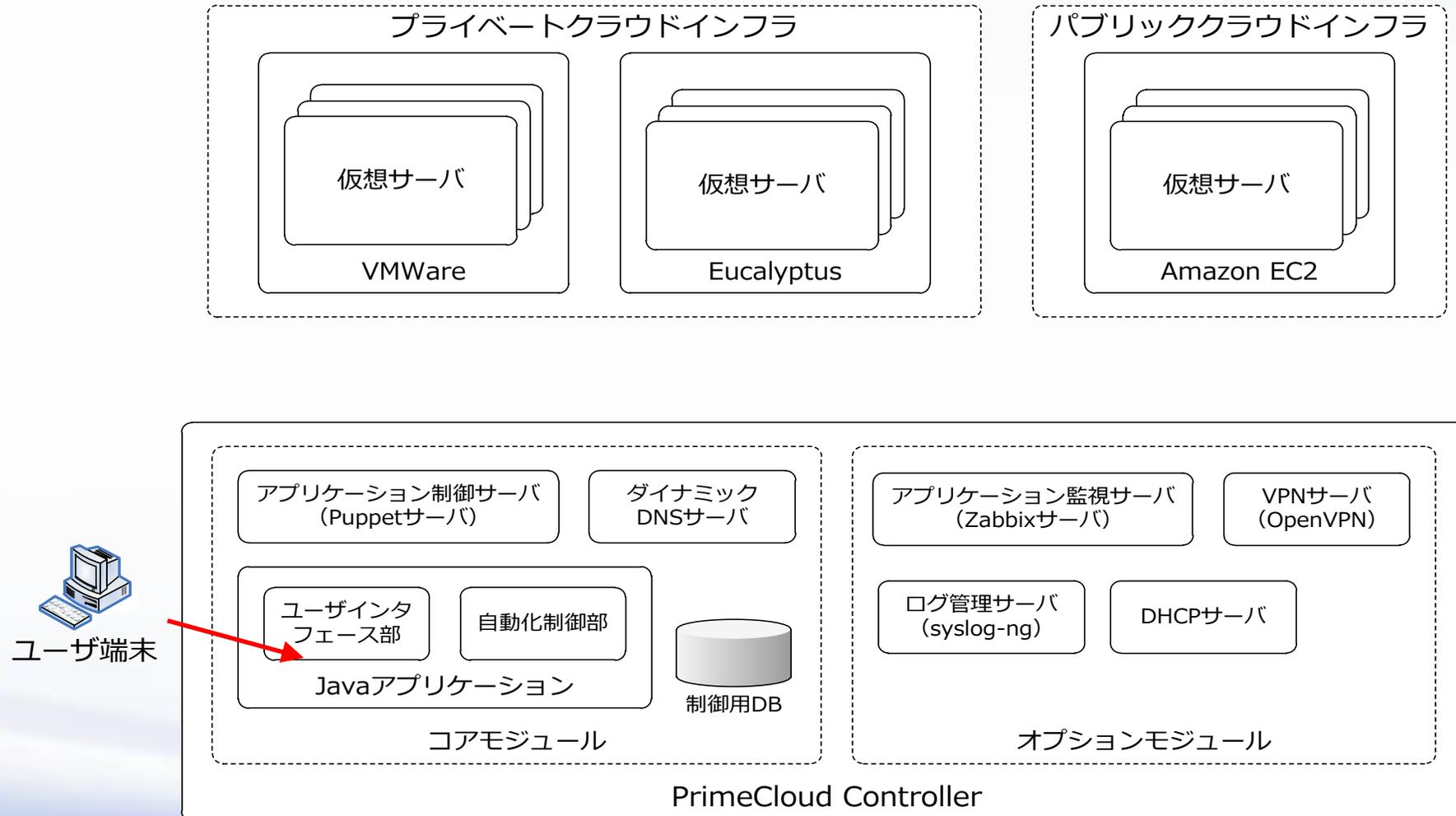
1. ユーザによって仮想サーバの起動操作が画面上で行われる
2. 自動化制御部からAPIを操作し、仮想サーバを起動（クローン、ハードウェア構成変更、パワーオン）する
 - パブリッククラウドの場合はVPN接続を行う、VPCやVMwareの場合はVPN接続をしない
 - コアモジュールからIaaS-GWモジュールを呼び出すことで起動する
 - DHCPにより仮想マシンにIPアドレスが割り振られる
 - アプリケーション制御用のエージェント（Puppet）が起動される
3. 割り振られたIPアドレスをインスタンスから取得し、DNSサーバに登録する
4. アプリケーション制御サーバを通して、仮想サーバの基本的な設定を行う
 - 仮想サーバからログ管理サーバへsyslogの転送設定がされる
 - アプリケーション監視用のエージェント（Zabbix）が起動される
5. アプリケーション監視サーバに、仮想サーバを監視対象として登録する
 - 仮想サーバの標準的なOSの監視が行われる
6. ユーザは仮想サーバを利用可能になる

サービス起動処理の流れ（例：MySQL）

1. ユーザによってMySQLサービスの起動操作が画面上で行われる
2. 仮想ディスクをサーバに追加する
3. MySQLサービスの起動状態を定義するマニフェストを生成する
4. Puppetクライアントがマニフェストを取得し、サーバ内で適用する
 - 仮想ディスクのパーティション作成、フォーマット
 - MySQLのデータベースファイルの作成
 - my.cnfの作成
 - MySQLの管理ユーザ、レプリケーション用ユーザなどの作成
5. MySQLサービス用の監視設定を追加する
6. ユーザはMySQLサービスを利用可能になる

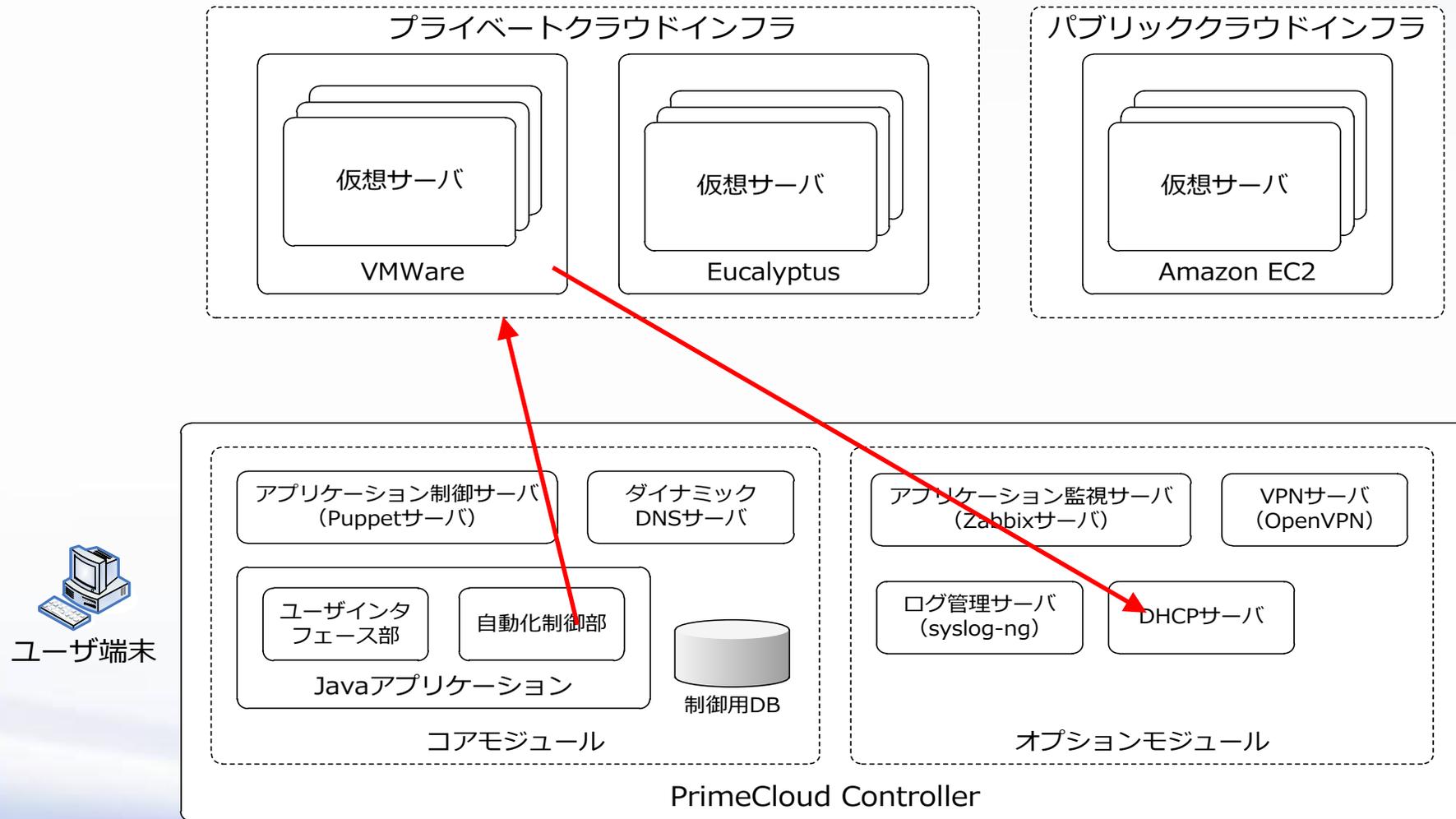
新規仮想サーバの起動処理の流れ

1. ユーザによる仮想サーバの起動操作



新規仮想サーバの起動処理の流れ

2. VMwareを操作し、仮想サーバを作成・起動する

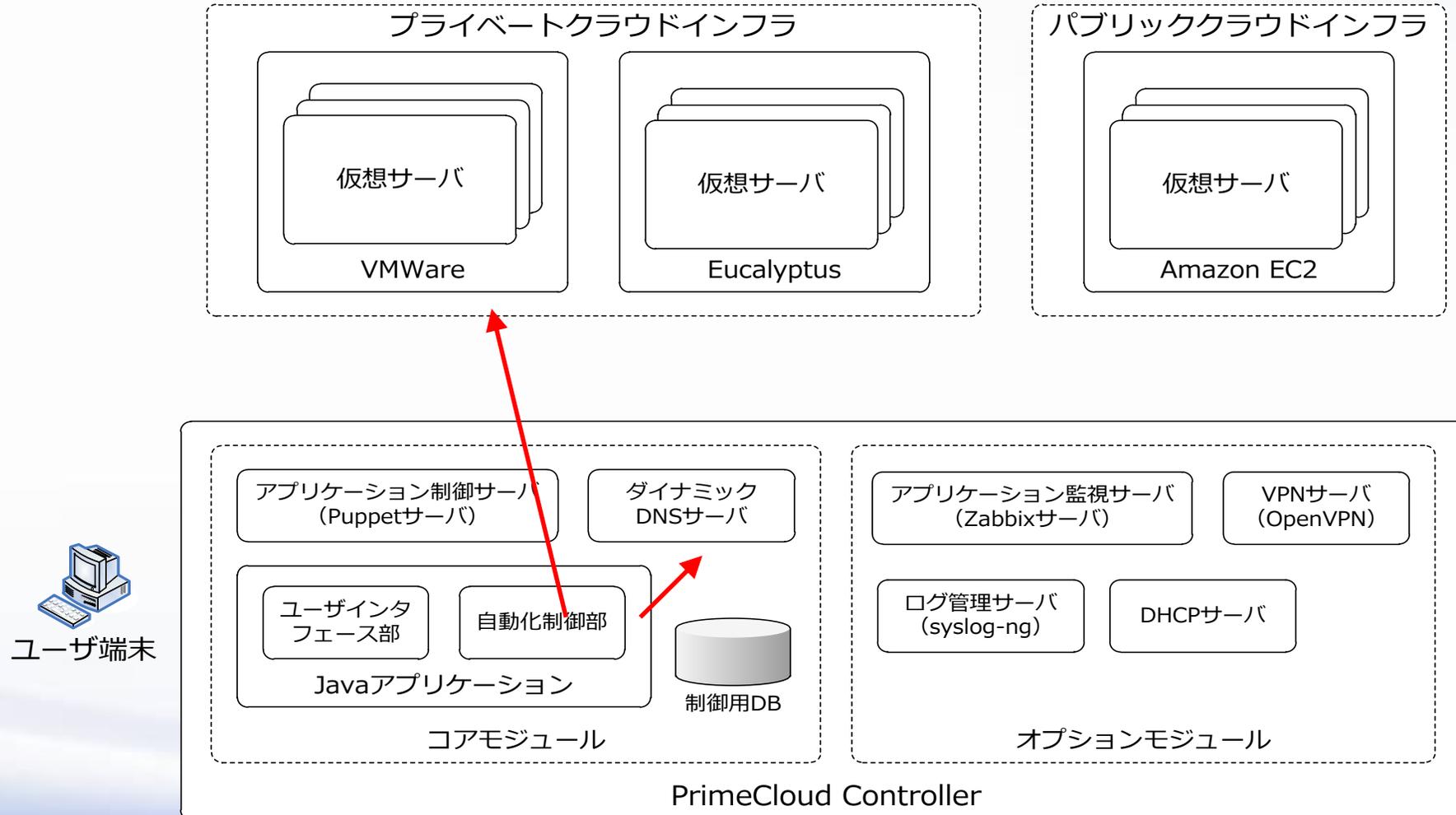


インスタンスを起動すると自動的に起動用のスクリプトが実行される

イメージ埋め込んだスクリプトと動的にWgetで取得するスクリプトがある。動的なスクリプトはクラウドのタイプによって異なる。

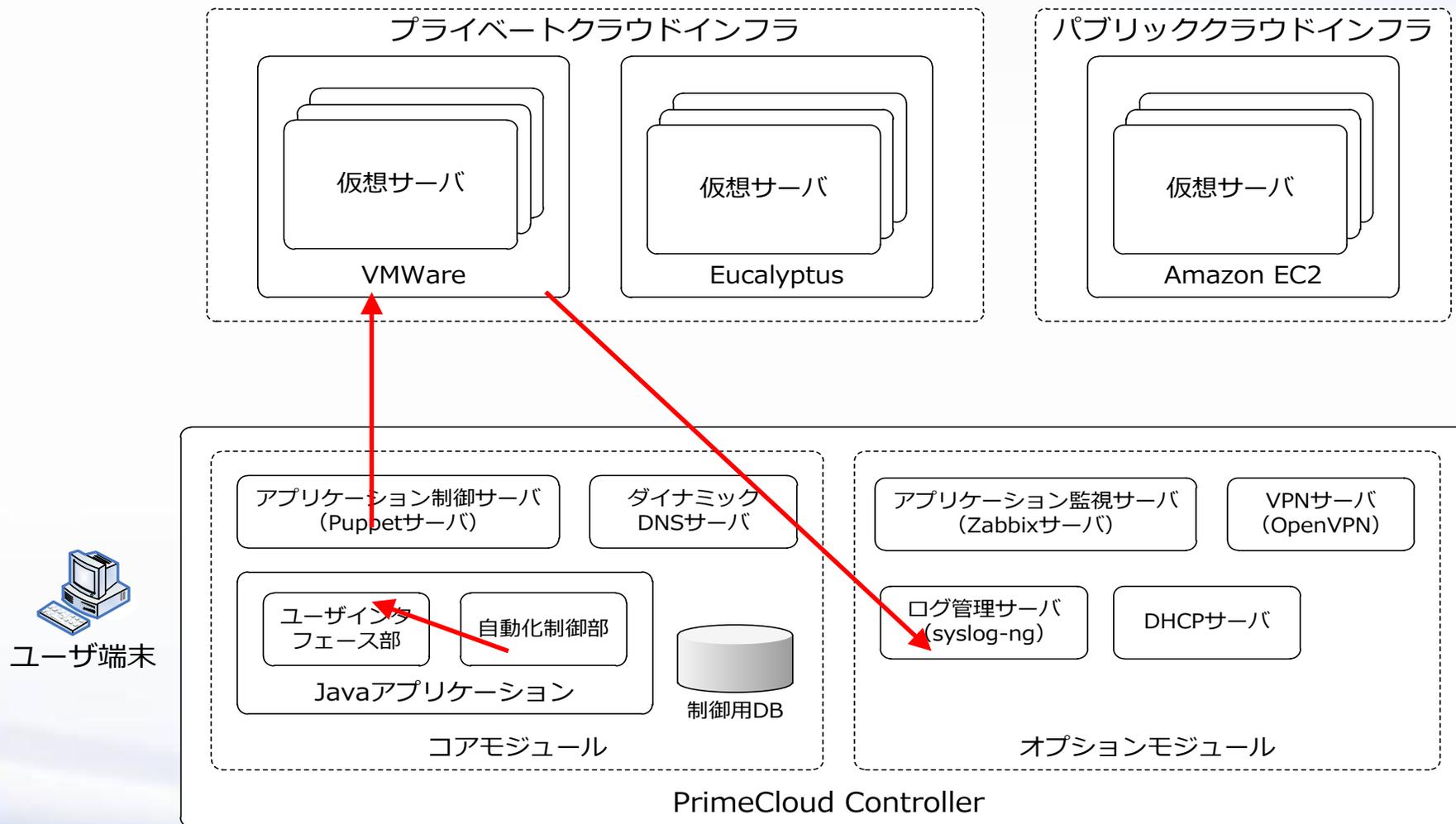
- イメージに静的に埋め込んだスクリプト
 - OpenVPN接続（パブリッククラウドの場合のみ）
 - 動的な初期設定スクリプトを取得
 - 動的な初期設定スクリプトを実行
- 動的に取得したスクリプト
 - ホスト名を設定
 - Puppetクライアントを設定およびパッチ適用
 - DNSリゾルバを設定
 - dnsmasqを設定および起動
 - VPNのIPアドレスをDNSサーバにAレコードとして登録（パブリッククラウドの場合のみ）
 - rootユーザでのログイン用の公開鍵を設定（VMwareの場合のみ）
 - DNSのAレコード削除用スクリプトを設定
 - スワップファイルを作成（Amazon EC2の場合のみ）
 - デフォルト言語を日本語（ja_JP.UTF-8）に設定

3. 仮想サーバのIPアドレスをDNSサーバに登録する



新規仮想サーバの起動処理の流れ

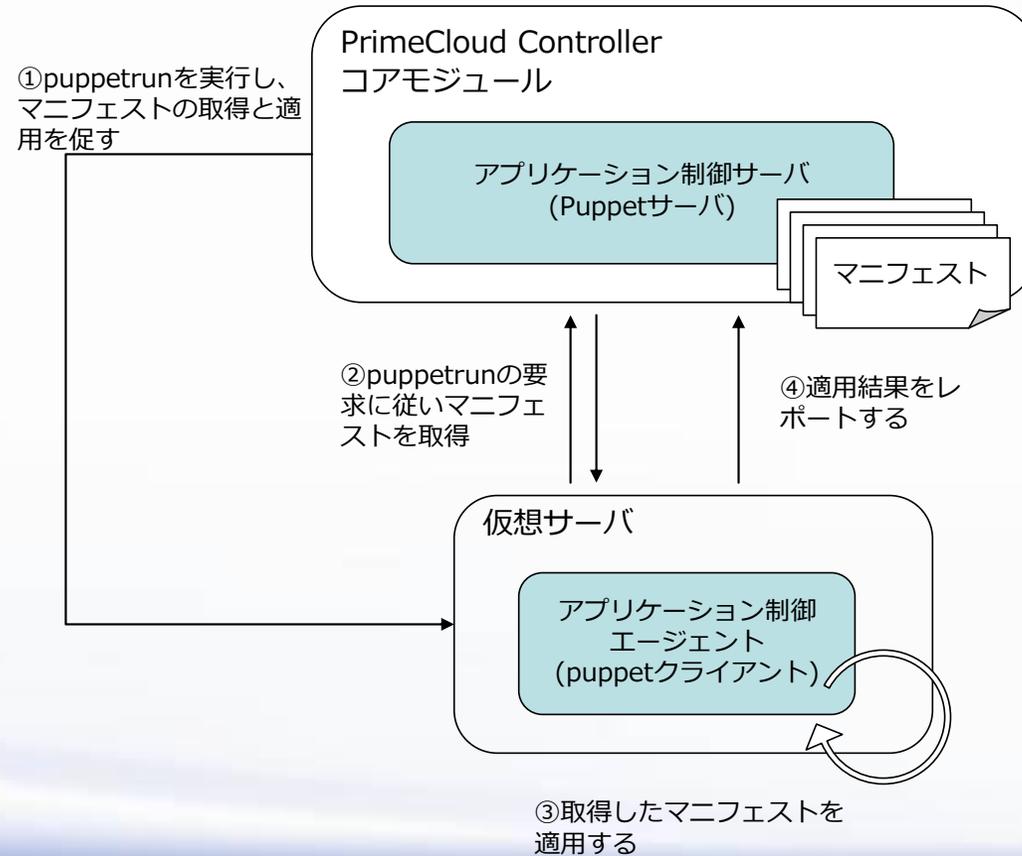
4. 仮想サーバの基本的な設定を行う



- OSのユーザ作成
- ディスクのパーティション作成、フォーマット
- MySQLの設定、ユーザ作成
- Tomcatの設定、アプリケーション展開
- Apache の設定、コンテンツ展開
- UltraMonkeyのロードバランス設定

補足) Puppet制御の概要

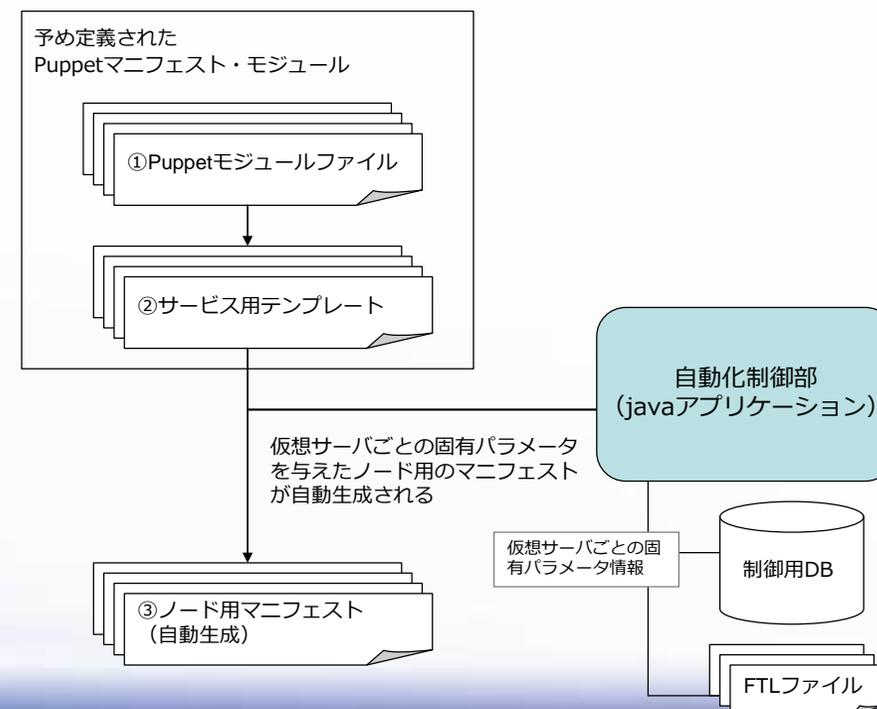
- 自動化制御部（Javaアプリケーション）は、仮想サーバごとにノードのマニフェストを生成した後、puppetrunコマンドを用いてマニフェストを仮想サーバへ適用する



補足) マニフェストファイル構成

マニフェストファイル種別	概要
①Puppetモジュールファイル (/etc/puppet/module/)	ディスク設定, Tomcat, MySQLなどのミドルウェアやパッケージ単位に用意したモジュール。
②サービス用テンプレート (/etc/puppet/manifest/template/)	PCCのサービスを構成するマニフェスト。 (サービスごとにPuppetモジュールを集約したもの)
③仮想サーバごとに作成されるノード用マニフェスト (/etc/puppet/manifest/auto/)	PCCが自動作成。サービス用テンプレートに、仮想サーバごとの固有設定値を与えたノードのマニフェストを生成する。 このマニフェストが仮想サーバへ適用される。

- ① Puppetモジュールファイルと
- ② サービス用テンプレートのマニフェストは、サーバに依らず共通のもので、あらかじめ作成して定義しておく。
- ③ ノード用のマニフェストは仮想サーバごとに異なるもので、自動化制御部 (Javaアプリケーション) から動的に作成される。

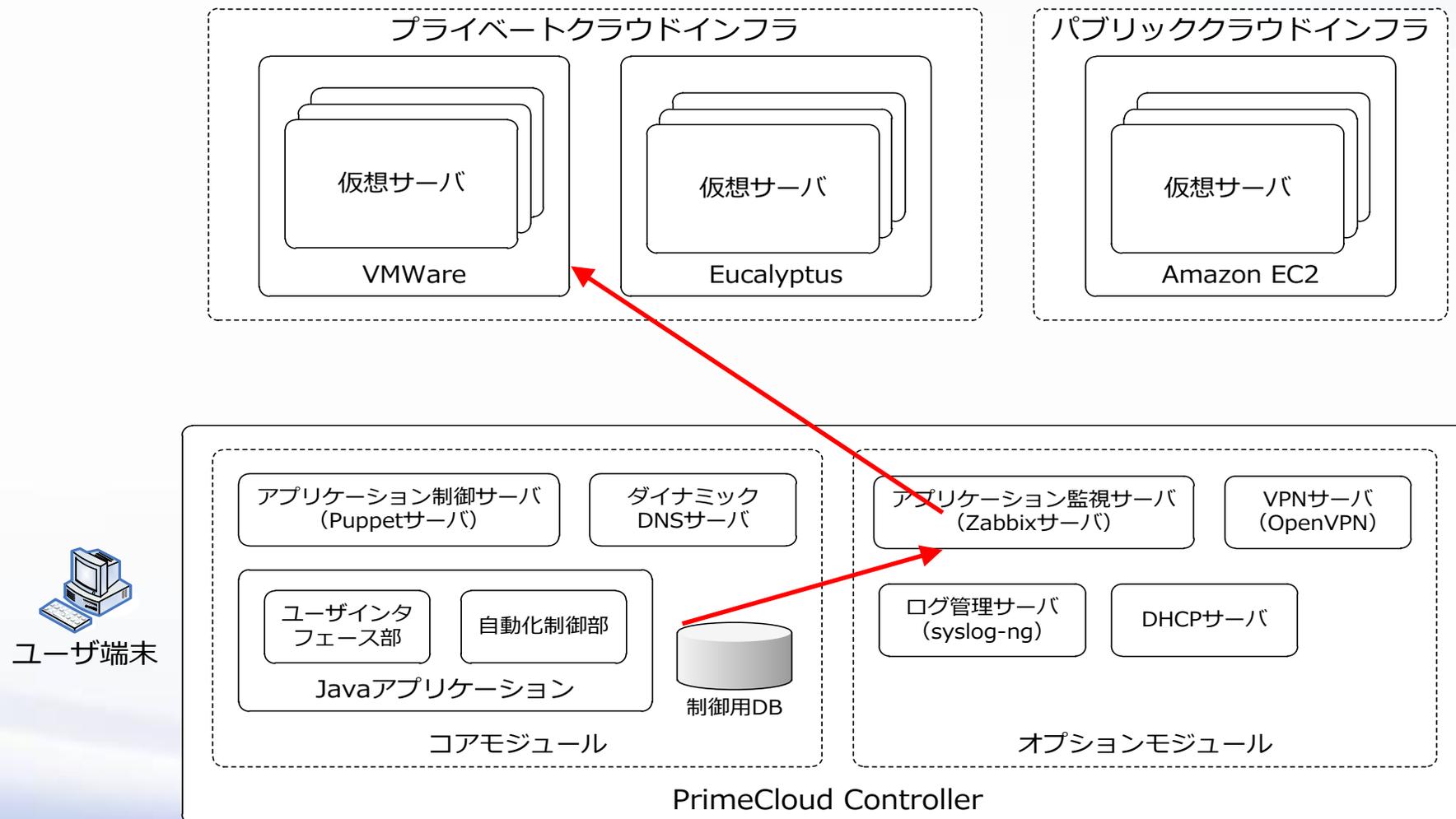


- サーバごとのマニフェストはJavaから自動生成している
 - FreeMarkerというテンプレートライブラリを使い、Puppetのマニフェストを生成

ノードの定義用マニフェスト	ノードごとに1つ作成される。 ファイル名: [仮想サーバFQDN].pp 例: server1.puppetdev.dev.csk-cc.com.pp
ノードの基本設定用マニフェスト	ノードごとに1つ作成される。 ファイル名: [仮想サーバFQDN].base.pp 例: server1.puppetdev.dev.csk-cc.com.base.pp
ノードの協調設定用マニフェスト	ノードごとに1つ作成される。 ファイル名: [仮想サーバFQDN].base_coordinate.pp 例: server1.puppetdev.dev.csk-cc.com.base_coordinate.pp
ノードのサービス設定用マニフェスト	各ノードのサービスごとに1つ作成される。 ファイル名: [仮想サーバFQDN].[サービス名].pp 例 : server1.puppetdev.dev.csk-cc.com.mysql.pp server1.puppetdev.dev.csk-cc.com.tomcat.pp server1.puppetdev.dev.csk-cc.com.apache.pp

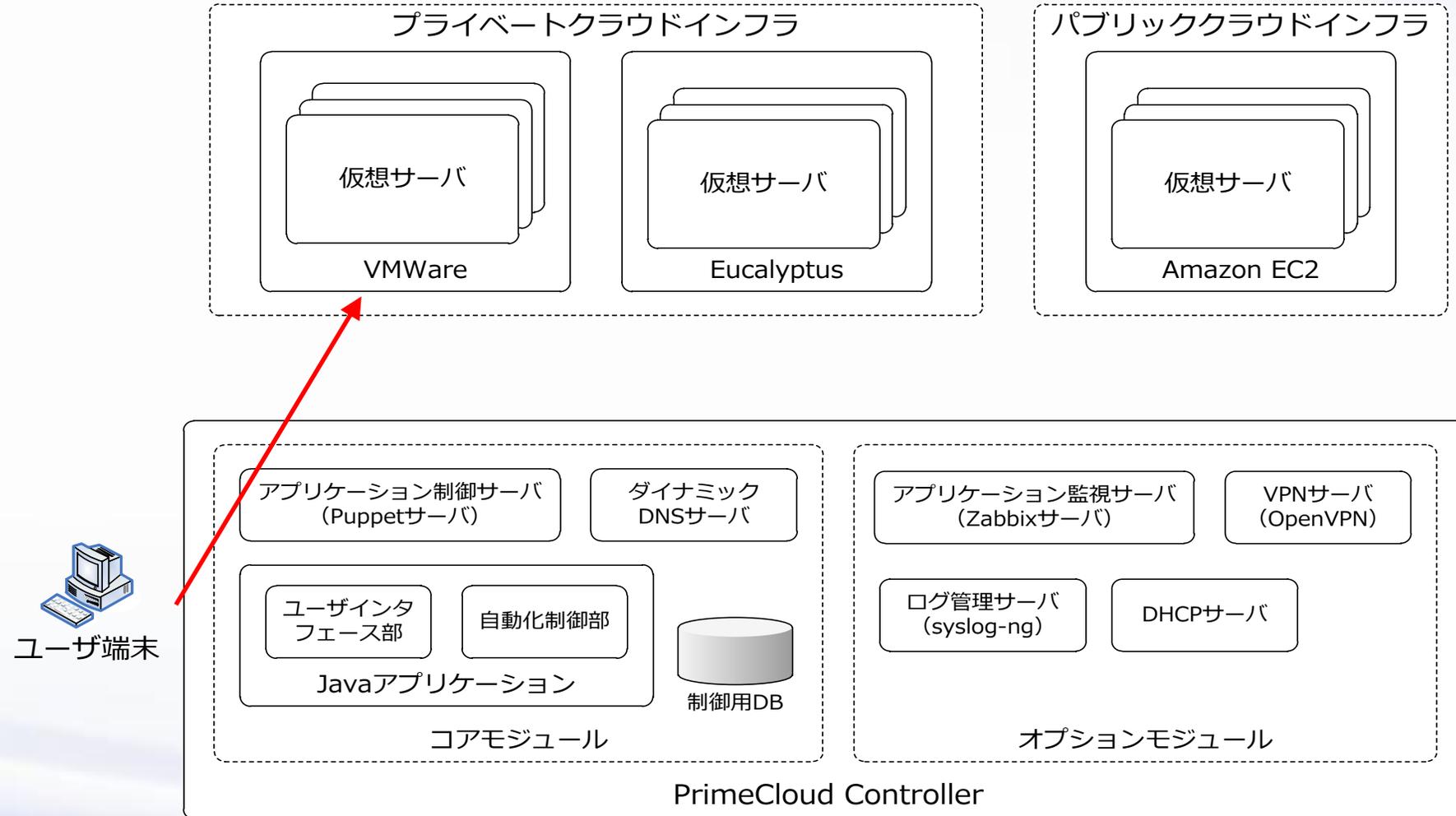
新規仮想サーバの起動処理の流れ

5. 監視サーバに仮想サーバを登録する



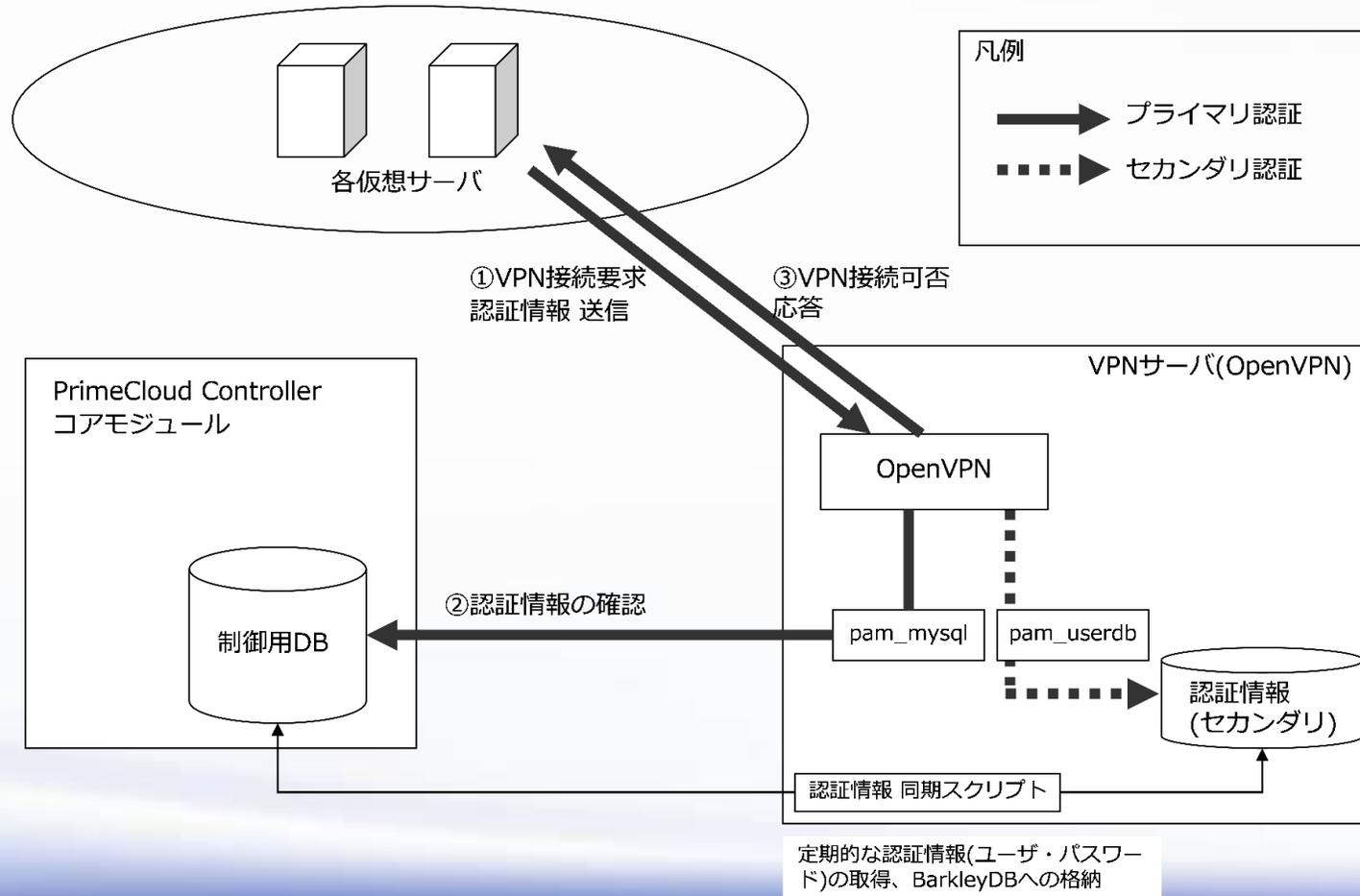
新規仮想サーバの起動処理の流れ

6. 仮想サーバが利用可能になる



補足) VPN認証の流れ

- パブリッククラウド上の仮想サーバが社内ネットワークやプライベートクラウド上の仮想サーバに対しアクセスするためのVPNサーバ
- OpenVPN接続時の認証情報は制御用DBのデータにアクセスし、認証を行う



3件の特許を取得済み

① サービス制御システム（特許番号4852734）

- サーバ/アプリ等の設定ファイルを自動生成し、管理する仕組み。

② サーバ復旧システム（特許番号4743726）

- 仮想サーバの起動の制御を行う仕組み。

③ VPN接続システム（特許番号4721082）

- VPN接続設定を自動的に行う仕組み。

3件の特許を取得済み

① サービス制御システム（特許番号4852734）

- サービス制御システムを、サービス提供側から管理する仕組み。

オープンソース版においては、
特許の使用を許諾。

② サービス

- 仮

すなわち、**無償**

③ VPN接続システム（特許番号4721082）

- VPN接続設定を自動的に行う仕組み。

1. PrimeCloud Controllerとは？
2. 動作の概要・仕組み
- 3. 近日リリース予定の新機能**
4. インストール方法
5. まとめ

PrimeCloud Controller / x

www.primecloud-controller.org

Apps PrimeCloud Contr... PrimeCloud Contr... ISANAAS-Board ... Other bookmarks

オープンソース版PrimeCloud Controller - ハイブリッドクラウド対応クラウドコントローラ
(AWS, Azure, IDCフロンティア, NIFTY Cloud, Eucalyptus, VMWare, CloudStack, USIZE対応)

SCSK

Prime Cloud Controller

ホーム PCCとは? ダウンロード ドキュメント フォーラム FAQ お問い合わせ

SCSK PrimeCloud Controller

25分でスタートアップ可能な
オープンソース版PrimeCloud Controllerの
構築手順を公開 (AMI編)
(2014/5/16 update)

・PrimeCloud Controllerのセミナー資料を公開しました。
「2014/5/9 クラウドオーケストレーションセミナー講演資料」

ツイート
いいね! シェア 213人がいいね!しています。

トピックス

2014/05/13 **更新情報** 「2014/5/9 クラウドオーケストレーションセミナー」で講演した資料を公開しました。

2014/04/18 **ニュース** PrimeCloud Controllerのセミナーを開催します。「2014/5/9 クラウドオーケストレーションセミナー (場所:品川)」

www.primecloud-controller.org/documents.html

漏洩の脆弱性に関する対応方法を記載しました。

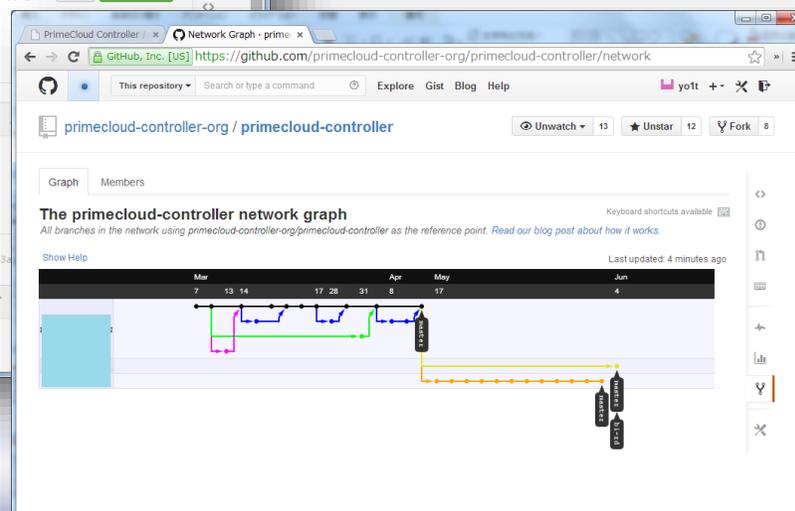
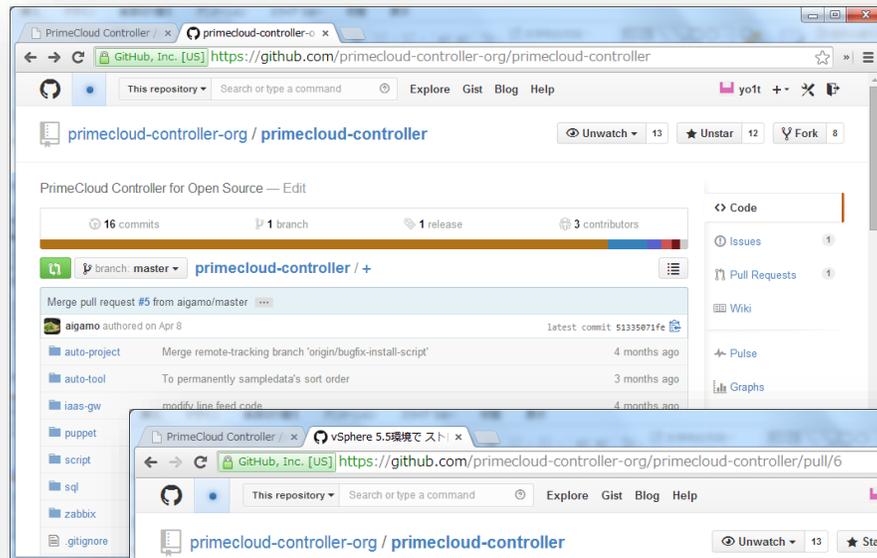
2014/3/31より、
オープンソース版も提供

<http://www.primecloud-controller.org/>

商用サポートサービスも提供します。
(運用保守サポート、開発サポート)

GitHubでのソース公開

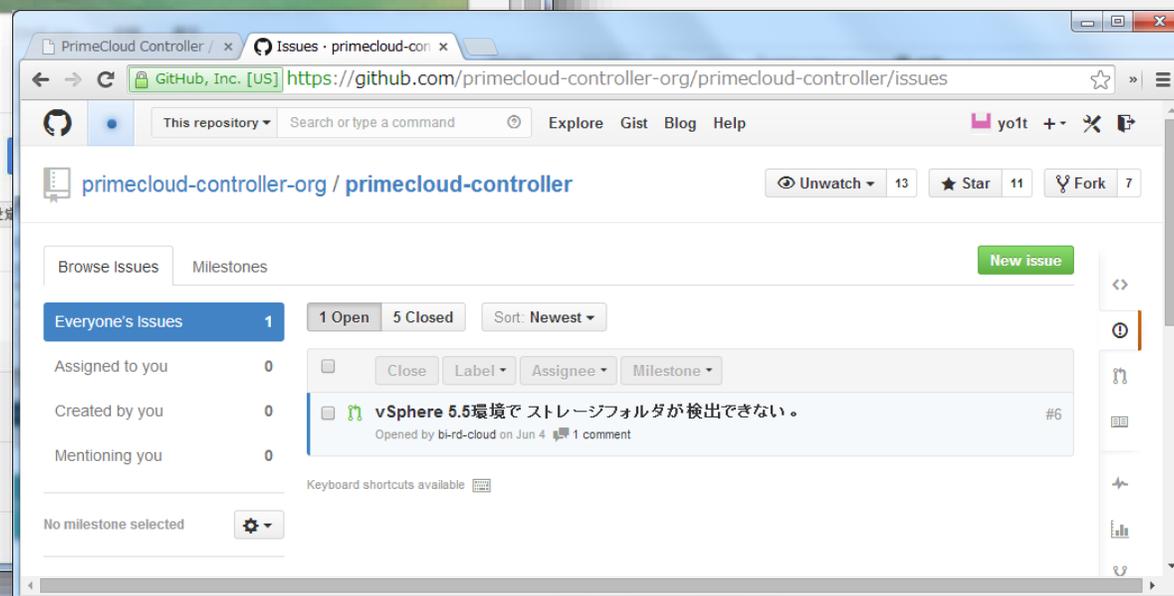
どなたでも、PrimeCloud Controllerの開発に参加することが可能です。
機能追加も自由に行うことができます。





フォーラム/GitHub-issue

どなたでも、不明な点や改善点を投稿可能。回答や意見についても、どなたでも返信可能。



OSS化して、何をしようとしているか？

- クラウドをもっと便利に。
- ハイブリッドクラウド環境をより簡単に。
- 自動化の促進。
- 効果的なITの活用。

- ITでより便利な世の中を実現。

PCC/オープンソース化による方向性

Phase2: ソフトウェア配信サービス

Phase1: オープンソース版提供

・ハイブリッドクラウド市場の醸成を行う。



PrimeCloud Controllerで利用可能なサービステンプレート（自動化パッケージ）を配信。ハイブリッドクラウド環境管理の自動化を促進。

ソフトウェアパッケージ

- PrimeCloud Controller設定情報
- RPM情報(RPM名、バージョン)
- puppetマニフェスト
- 設定ファイル/スクリプトなどをパッキング

パブリック
or
プライベート
(自組織内)

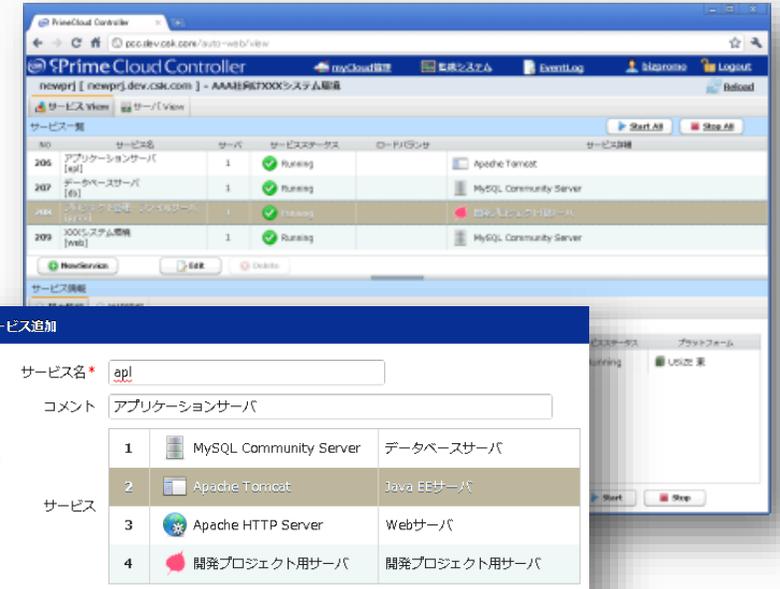


PrimeCloud Controller 管理者用コマンド

① PCC list
(パッケージ一覧表示)

② PCC install xxx
(パッケージ取り込み)

PCC info xxx
(パッケージ情報表示)



③ サービス一覧に表示

④ デプロイ



- PCCテンプレートとしてファイルサーバを追加したい場合
(PCCのサービス一覧に、「ファイルサーバ」を追加)

サービス	サービス名	説明
1	MySQL Community Server	データベースサーバ
2	Apache Tomcat	Java EEサーバ
3	Apache HTTP Server	Webサーバ
4	Windows Server	Windows Server

選択可能サーバ: web06, web7

選択済みサーバ:

Buttons: Add New Server, Add, Cancel

pcc list コマンド実行

```
[root@pccdemo:~]# pcc list
Loaded plugins: fastestmirror, presto
Cleaning repos: base epel extras puppetlabs-deps puppetlabs-products updates
                : zabbix-jp zabbix-jp-relatedpkgs
Cleaning up Everything
Cleaning up list of fastest mirrors
delta-package files removed, by presto
pcc-apache.x86_64                0.01-0                @pcc
pcc-apache_content.x86_64       0.01-0                @pcc
pcc-apache_module.x86_64       0.01-0                @pcc
pcc-fileserver.x86_64          0.01-0                @pcc
pcc-icons.x86_64              0.01-0                @pcc
pcc-lvm_module.x86_64          0.01-0                @pcc
pcc-webserver.x86_64           0.01-0                @pcc
pcc-apserver.x86_64            0.01-0                pcc
pcc-base.x86_64                0.01-0                pcc
pcc-base_coordinate.x86_64     0.01-0                pcc
pcc-basenode.x86_64            0.01-0                pcc
pcc-dbserver.x86_64            0.01-0                pcc
pcc-geronimo.x86_64            0.01-0                pcc
```

pcc install pcc-fileserverコマンド実行

```
[root@pccdemo:~]# pcc info pcc-fileserver
Loaded plugins: fastestmirror, presto
Loading mirror speeds from cached hostfile
Installed Packages
Name       : pcc-fileserver
Arch      : x86_64
Version   : 0.01
Release   : 0
Size      : 6.1 k
Repo      : installed
From repo : pcc
Summary   : PCC service define manifest (File Server)
License   : SCSK
Description: File Server用の puppetサービス定義
          : このパッケージは以下のパッケージ
          :   pcc-lvm_module

[root@pccdemo:~]# pcc install pcc-fileserver
Loaded plugins: fastestmirror, presto
Loading mirror speeds from cached hostfile
Setting up Install Process
Resolving Dependencies
--> Running transaction check
--> Package pcc-fileserver.x86_64 0:0.01-0 will be installed
--> Finished Dependency Resolution

Dependencies Resolved

=====
Package             Arch      Version      Repository    Size
=====
Installing:
pcc-fileserver      x86_64    0.01-0       pcc            4.3 k
=====

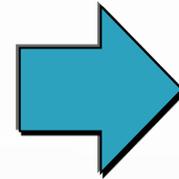
Transaction Summary
=====
Install      1 Package(s)

Total download size: 4.3 k
Installed size: 6.1 k
Is this ok [y/N]: y
Downloading Packages:
Setting up and reading Presto delta metadata
Processing delta metadata
Package(s) data still to download: 4.3 k
pcc-fileserver-0.01-0.x86_64.rpm      | 4.3 kB    00:00
Running rpm_check_debug
Running Transaction Test
Transaction Test Succeeded
Running Transaction
  Installing : pcc-fileserver-0.01-0.x86_64      1/1
  Verifying  : pcc-fileserver-0.01-0.x86_64      1/1

Installed:
pcc-fileserver.x86_64 0:0.01-0

Complete!
[root@pccdemo:~]#
```

PCCのサービス一覧に、
「ファイルサーバ」が追加！



サービス追加

サービス名 * samba01

コメント

サービス	サービス名	説明
2	Apache Tomcat	Java EEサーバ
3	Apache HTTP Server	Webサーバ
4	Windows Server	Windows Server
5	samba	ファイルサーバ

ディスクサイズ(GB) * 2

選択可能サーバ

web7

選択済みサーバ

web06

サーバ選択

Add New Server : このサービスに新規サーバを追加

Add Cancel

Phase2:ソフトウェア配信サービスの進め方

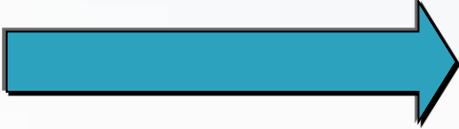
コミュニティでの
対応も検討

OSS

・SCSK OSS RaderScope ※
150ソフトウェアを対象とする。
※SCSKがOSSの格付けを実施している



・上位評価XX程度を選別
(最終的には全部を対象)



商用プロダクト

対象候補リストアップ

対象商材リストを作成し、
選出

対象選別

配信できるかどうか
を検証



- 【条件 (or)】
- 1.ソフトウェアであること
 - 2.サービスであること
(接続用のクライアントソフトを配信)
 - 3.製品デモ/評価版



契約条件・整備

版元とサービス型
契約を整備する。

- 1.体制構築
- 2.既存契約で包含
できるか?
- 3.既存ビジネスへ
の影響検討



クラウド標準化ラボ(テンプレート化・テスト)



ソフトウェア配信サーバ登録・展開

ログ - Google Chrome

ppcdev09.lab.primecloud.jp/auto-web/view/2/

ユーザ名:dev09-user01

表示日時
 現在時刻から 1日以内
 日時指定 From: To:

myCloud名 sysytem001 ログレベル
 サービス名 サーバ名

自動更新

リセット 検索

最終更新時刻:2014/06/18 17:20:19

日時	ログレベル	MYCLOUD名	サービス名	サーバ名	メッセージ
2014/06/18 13:24:15	DEBUG	sysytem001		ap1	Zabbixテンプレート適用(ホスト=ap1.sysytem001.dev9.lab.primecloud.jp, テンプレート=template_1)
2014/06/18 13:24:14	DEBUG	sysytem001		ap1	Zabbix監視登録(ホスト=ap1.sysytem001.dev9.lab.primecloud.jp, ID=10393)
2014/06/18 13:24:14	DEBUG	sysytem001		ap1	サーバ基本設定完了
2014/06/18 13:24:14	DEBUG	sysytem001		ap1	Puppetマニフェスト適用完了(ホスト=ap1.sysytem001.dev9.lab.primecloud.jp, タイプ=base)
2014/06/18 13:23:43	DEBUG	sysytem001		ap1	Puppetマニフェスト適用開始(ホスト=ap1.sysytem001.dev9.lab.primecloud.jp, タイプ=base)
2014/06/18 13:23:32	DEBUG	sysytem001		db1	仮想マシン起動開始(プラットフォーム=vmware, 仮想マシン=sysytem001_db1)
2014/06/18 13:23:28	DEBUG	sysytem001		db1	仮想マシン作成完了(プラットフォーム=vmware, 仮想マシン=sysytem001_db1)
2014/06/18 13:21:06	DEBUG	sysytem001		ap1	サーバ基本設定開始
2014/06/18 13:21:06	DEBUG	sysytem001		ap1	DNSサーバ登録(ホスト=ap1.sysytem001.dev9.lab.primecloud.jp, IPアドレス=172.20.2.245)
2014/06/18 13:21:04	DEBUG	sysytem001		ap1	インスタンス作成完了(プラットフォーム=EC2, インスタンス=i-8f6f5888)
2014/06/18 13:20:46	DEBUG	sysytem001		ap1	インスタンス作成開始(プラットフォーム=EC2)
2014/06/18 13:20:45	DEBUG	sysytem001		db1	仮想マシン作成開始(プラットフォーム=vmware, 仮想マシン=sysytem001_db1)
2014/06/18 13:20:45	INFO	sysytem001		db1	サーバ起動開始
2014/06/18 13:20:45	INFO	sysytem001		ap1	サーバ起動開始
2014/06/18 13:20:15	DEBUG	sysytem001	web		サーバ割り当て変更(サーバ=[ap1])
2014/06/18 13:20:15	INFO	sysytem001	web		サービス新規作成(サービス=apache)
2014/06/18 13:19:53	DEBUG	sysytem001	apl		サーバ割り当て変更(サーバ=[ap1])
2014/06/18 13:19:53	INFO	sysytem001	apl		サービス新規作成(サービス=tomcat)
2014/06/18 13:19:25	INFO	sysytem001		ap1	サーバ新規作成(プラットフォーム=ec2_vpc_tokyo)
2014/06/18 13:18:53	DEBUG	sysytem001	db		サーバ割り当て変更(サーバ=[db1])
2014/06/18 13:18:53	INFO	sysytem001	db		サービス新規作成(サービス=mysql)
2014/06/18 13:18:50	INFO	sysytem001		db1	サーバ新規作成(プラットフォーム=vmware)
2014/06/18 13:17:18	INFO	sysytem001			myCloud新規作成

クラウドの利用状況と
 コストを集計・分析し、
 最適解を提示する。

1. PrimeCloud Controllerとは？
2. 動作の概要・仕組み
3. 近日リリース予定の新機能
- 4. インストール方法**
5. まとめ

PrimeCloud Controller / x
www.primecloud-controller.org/documents.html

オープンソース版PrimeCloud Controller - ハイブリッドクラウド対応クラウドコントローラ
(AWS, Azure, IDCフロンティア, NIFTY Cloud, Eucalyptus, VMWare, CloudStack, USIZE対応)

Prime Cloud Controller

ドキュメント

インストールチュートリアル

ダウンロードした方は、再度ダウンロードしてください。

TIPS

- 各種ログの見方
- 各種ファイルの解説

アーキテクチャ

- アーキテクチャガイド

事前準備 x
www.primecloud-controller.org/documentation/tutorial/step0.html

オープンソース版PrimeCloud Controller - ハイブリッドクラウド対応クラウドコントローラ
(AWS, Azure, IDCフロンティア, NIFTY Cloud, Eucalyptus, VMWare, CloudStack, USIZE対応)

Prime Cloud Controller

インストール環境の概要

PrimeCloud Controller (以降PCC)で使用するサーバを用意します。今回はAmazon Web Services(以降AWS)上にインストールすることを想定します。ハイブリッドクラウド構成に関しては、次回のドキュメント公開のタイミングにて記載する予定です。

構成イメージ

本チュートリアルでの解説範囲

- US East (N.Virginia)
EC2 Instance
- PrimeCloud Controller 管理サーバ
- Asia Pacific (Tokyo)

1. 事前準備

- インストール環境の概要
- 構成イメージ
- PCC全体構成
- アプリケーションサーバ
- 設定パラメーター一覧
- インストーラのソースツリー

2. インストール環境の作成

3. OpenVPNサーバのインストール

4. DNSサーバのインストール

5. インストールスクリプトの実行

6. インスタンスイメージの登録

各ドキュメントのページビュー数

1. 事前準備

- インストール環境の概要
- 構成イメージ
- PCC全体構成
- アプリケーションサーバ
- 設定パラメータ一覧
- インストーラのソースツリー

documents.htmlのPV:1921

(2014/3/31~7/14)

→ 774

ここが脱落ポイント

2. インストール環境の作成

→ 308

3. OpenVPNサーバのインストール

→ 341

4. DNSサーバのインストール

→ 144

5. 1

→ 203

6. 1

→ 193

7. Z

→ 206

8. PCC利用ユーザの作成

→ 160

おおむね、インストール完了まで
2時間はかかります。

AMIの提供：約**25分**で環境構築可能に。

他の環境向けにも、順次、
簡単なインストール環境を
提供していきます。



The screenshot shows a web browser window displaying the documentation for Prime Cloud Controller (PCC) on Amazon Web Services (AWS). The page title is "AMI編(DefaultVPC)" and the URL is "www.primecloud-controller.org/documentation/easy/ami.html". The page content includes the following sections:

- 25分で出来る簡単スタートアップ方法(AMI編)**
簡単な動作の確認や、検証にご利用いただけるように DefaultVPC環境を前提としてセットアップ済みのAMIを作りました。インスタンス作成後、パラメータ変更用のscriptを実行するだけで利用を開始することができます。
- 想定している構成**
 - ・ TokyoリージョンのDefault VPC環境
 - ・ Elastic IPをPCCサーバに割り当て利用します。
 - ・ PCCサーバが起動しているsubnet上にPCCを利用してインスタンスを作成します。
- 利用手順**
 - AWS環境の確認**
AWSアカウントには、アカウントの取得時期や利用するリージョン等の違いにより AmazonEC2で利用可能なプラットフォームが異なる場合があります。
 - ・ EC2-Classic
 - ・ EC2-VPC (デフォルト VPC *本手順ではDefaultVPCと表記します)
 - ・ EC2-VPC (デフォルトではない VPC)

SCSKにて、下記を提供します。

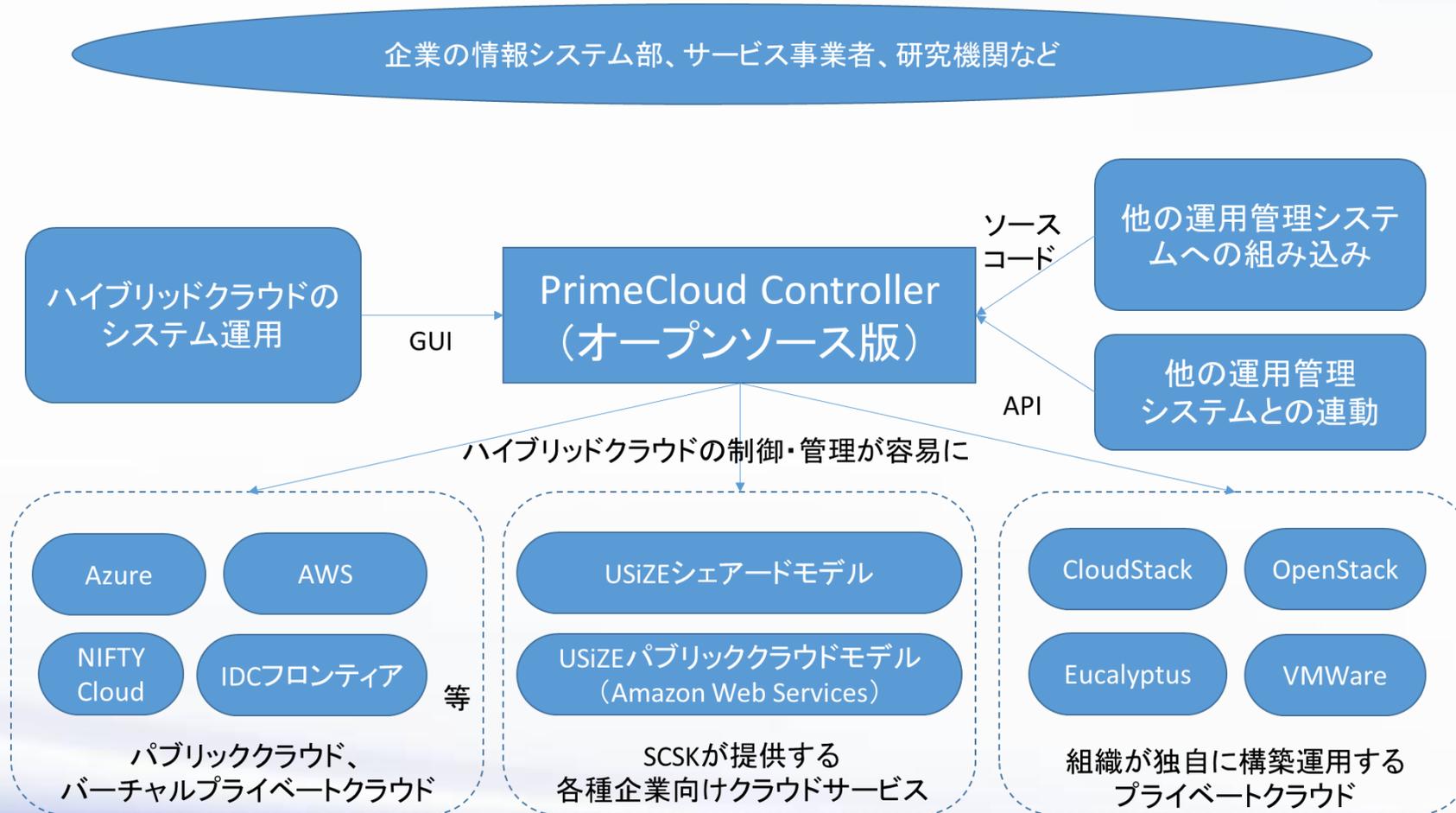
- ①PrimeCloud Controller/OSSサポートサービス
 - ②PrimeCloud Controller/OSS環境構築支援
-

あと、

Phase2で提供予定のソフトウェア配信において、
パートナーとして興味をお持ちの方いましたら、ご連絡ください。

1. PrimeCloud Controllerとは？
2. 動作の概要・仕組み
3. 近日リリース予定の新機能
4. インストール方法
- 5. まとめ**

PrimeCloud Controllerはハイブリッドクラウド環境と、運用の自動化を実現。
安心・安全なIT環境の実現を目指しています。



ぜひ、インストールして評価を！

Prime Cloud Controller

<http://www.primecloud-controller.org/>