

Real Application Centric Kernelで実現する 「クラウドネイティブアプリケーション」

伊藤忠テクノソリューションズ株式会社
金子 雄大

- ▶ 「クラウドネイティブ」アプリケーション
- ▶ Real Application Centric Kernel(R・A・C・K)
- ▶ ケーススタディ powered by RACK
- ▶ RACKの今後の展望

- ▶ 「クラウドネイティブ」アプリケーション
- ▶ Real Application Centric Kernel(R・A・C・K)
- ▶ ケーススタディ powered by RACK
- ▶ RACKの今後の展望

「クラウドネイティブ」アプリケーション



今後求められていく

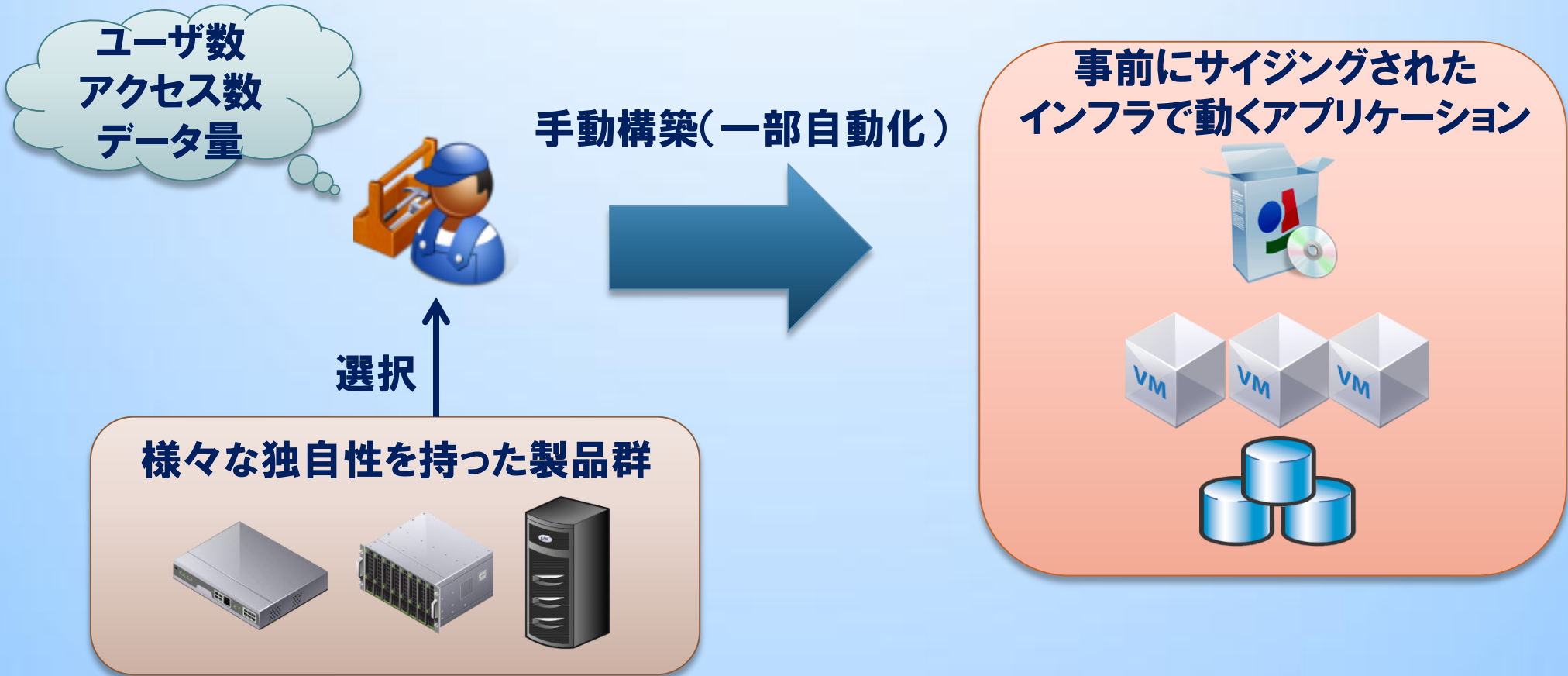
新しいアーキテクチャを持ったアプリケーション

なぜ新しいアーキテクチャが必要なのか？

ユーザ要件

- ✓ 業務効率化
- ✓ コスト削減

事前に予測された量の処理を早く正確にこなす



ビッグデータというキーワードの登場…

ユーザ要件

- ✓ 未来予測
- ✓ 価値創造

予測困難な量の処理をこなす



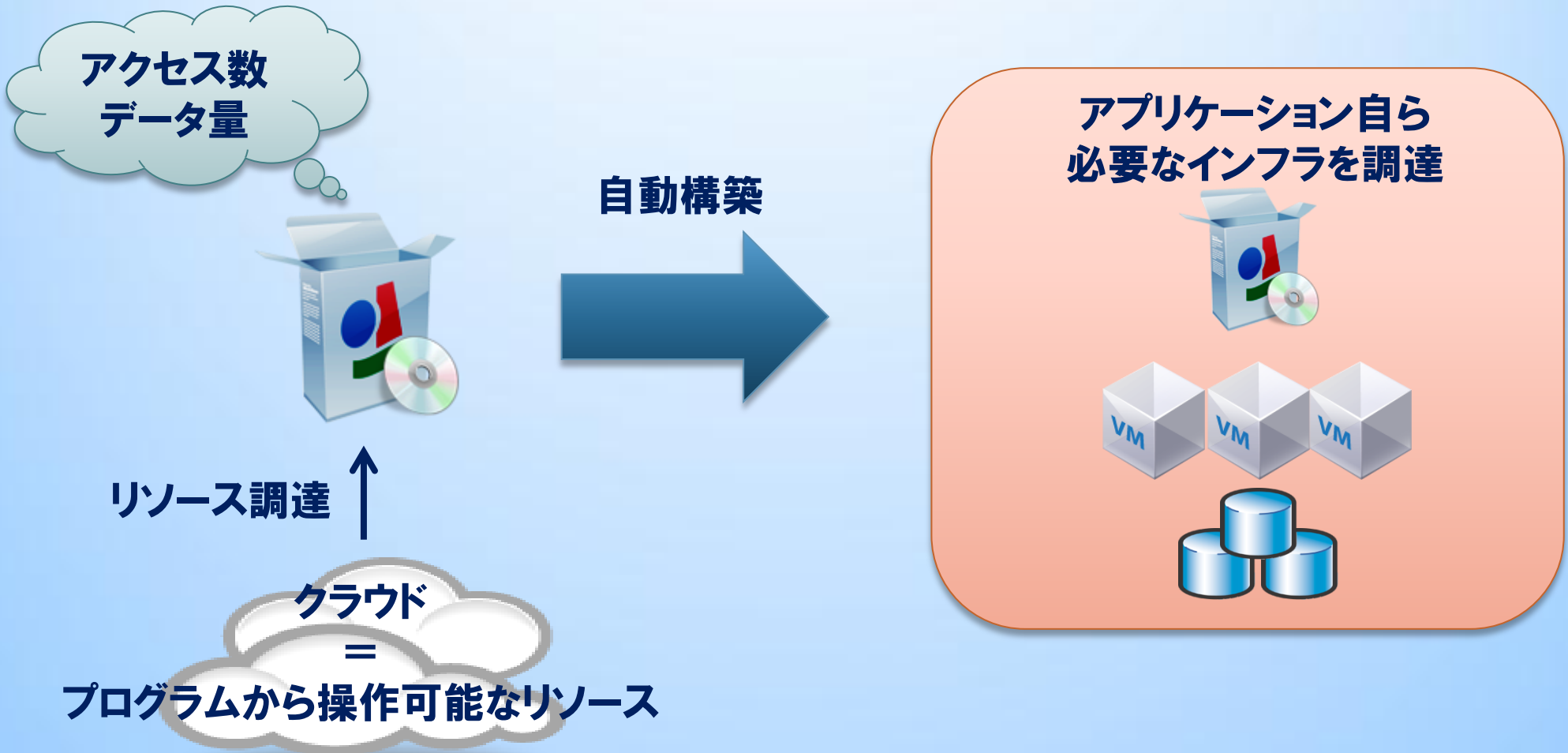
目的・時期・タイミングで必要なリソース量が変化する

事前のサイジングは困難

人手で対応することは不可能

クラウドネイティブアプリケーション

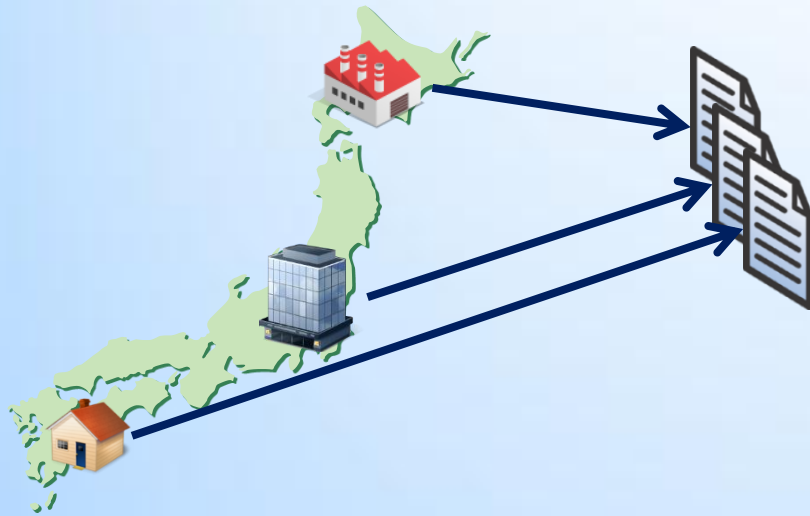
自分に必要なリソース量を判断しクラウドから調達して動作する



クラウドネイティブアプリケーションって どんなもの？

例えばこんなアプリケーション

日本全国の家庭・ビル・工場の一
日の電気使用量のデータ…など

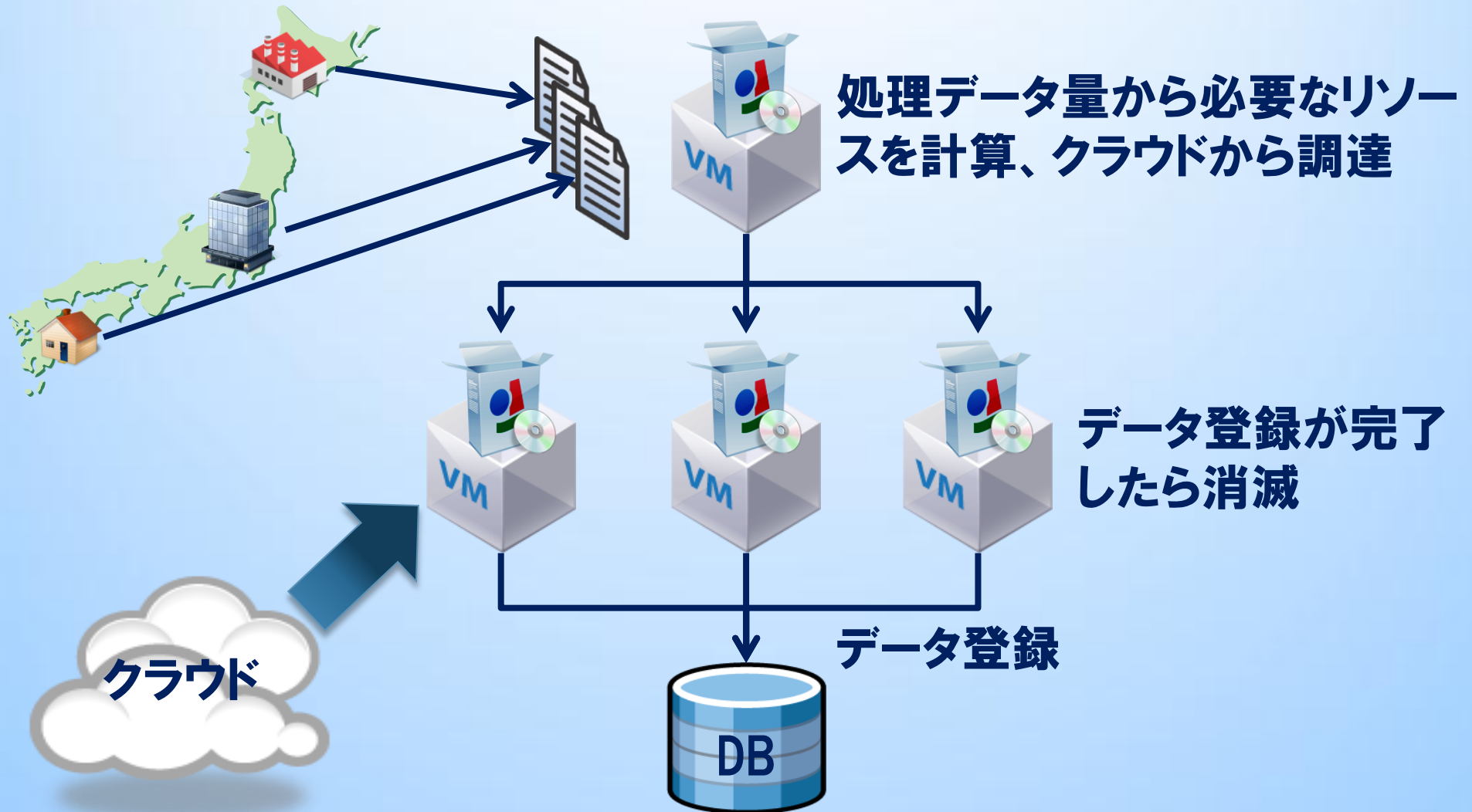


処理データ量から必要なリソー
スを計算、クラウドから調達



例えばこんなアプリケーション

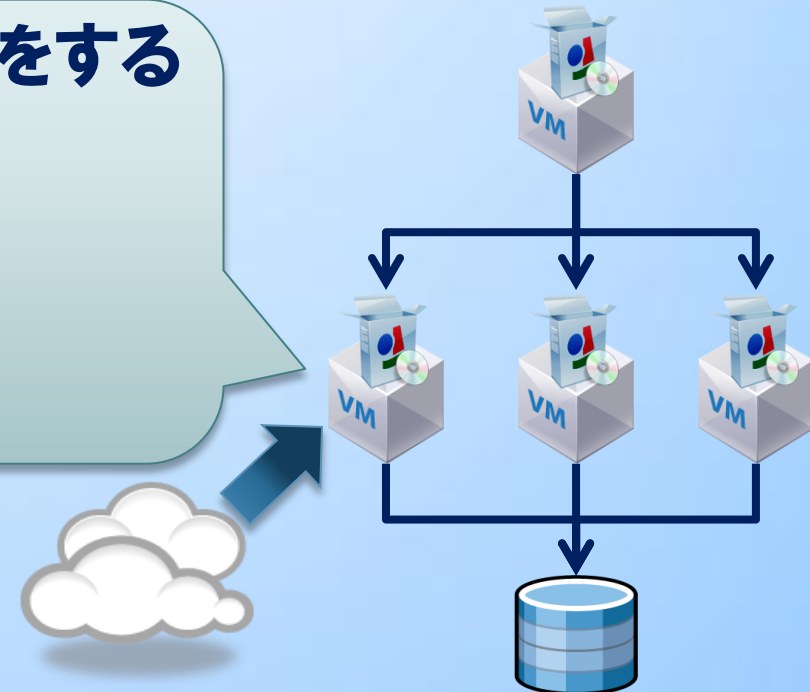
日本全国の家庭・ビル・工場の一
日の電気使用量のデータ…など



アプリケーション内部にリソースを制御する実装が必要

- ✓ 処理するデータ量から必要なVM台数を計算
- ✓ VMを起動し、ネットワーク接続を行う
- ✓ 起動させたVMの準備が完了するまで待つ
- ...

- ✓ データ登録を行うために必要な設定をする
- ...
- ✓ 処理が終わったら自分を削除
- ...



アプリケーション内部にリソースを制御する実装が必要

✓ 処理するデータ量から必要なVM台数を計算

✓ VMを起動し、ネットワーク接続を行う

✓ 起動されたVMが準備が完了するまで待つ

1. VMテンプレートイメージIDを調べる

2. ネットワークIDを調べる

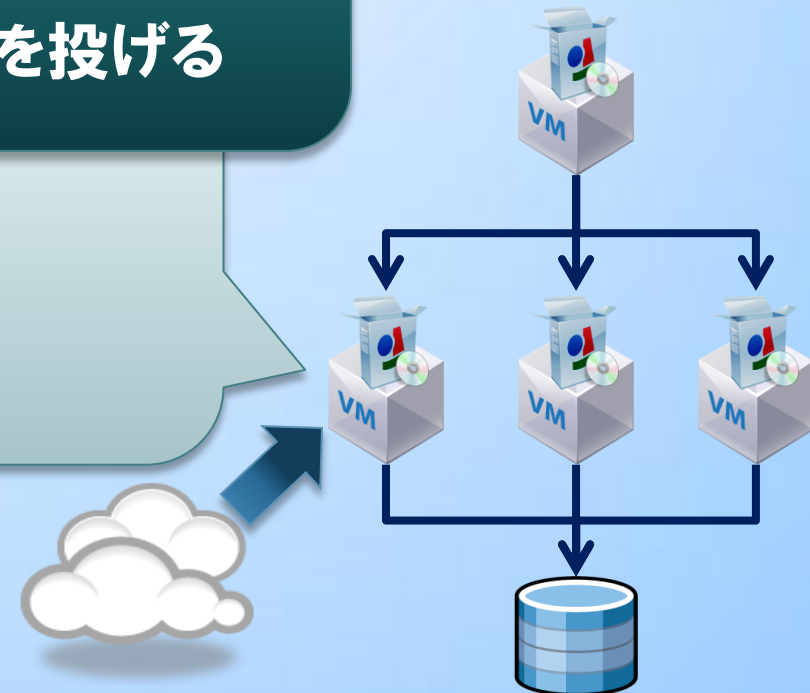
3. クラウドにVMを起動するリクエストを投げる

✓ データを処理する

...

✓ 処理が終わったら自分を削除

...



アプリケーション内部にリソースを制御する実装が必要

- ✓ 処理するデータ量から必要なVM台数を計算
- ✓ VMを起動し、ネットワーク接続を行う
- ✓ 起動させたVMの準備が完了するまで待つ

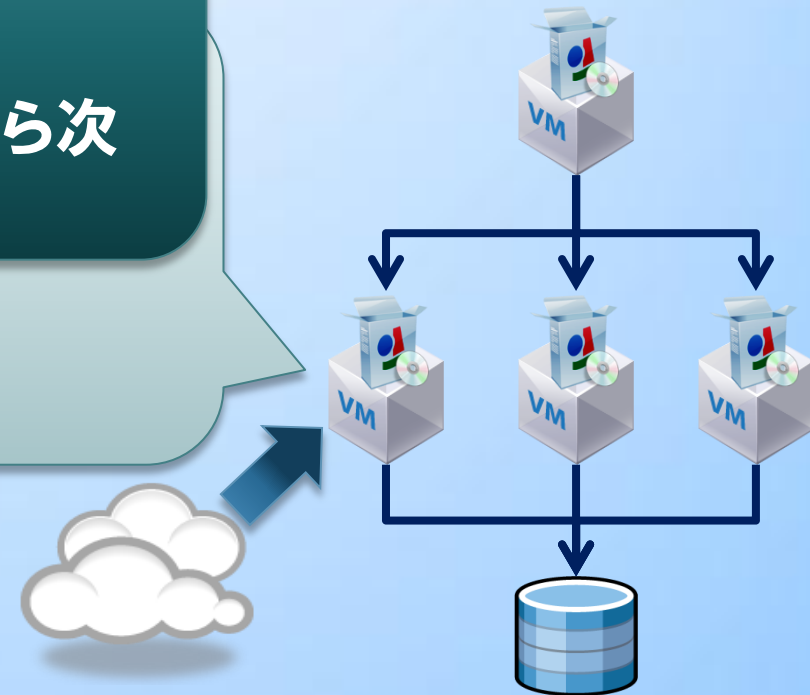
1. 起動させたVMのステータスを定期的に確認する
2. VMのステータスがACTIVEになったら次の処理に移る

✓ デー

...

✓ 処理が終わったら自分を削除

...

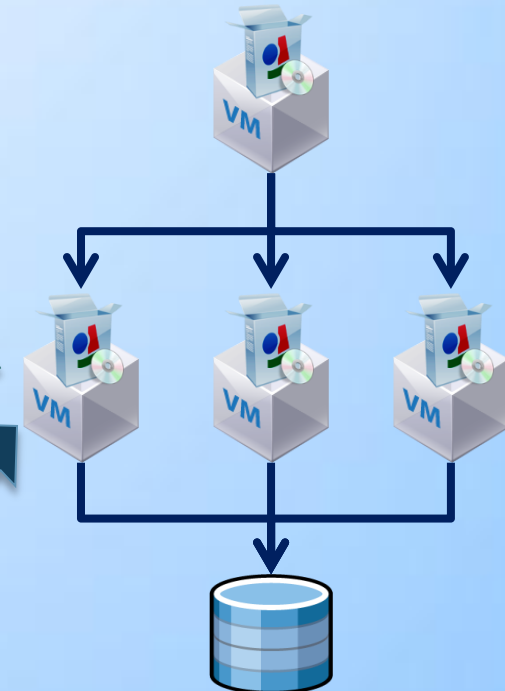


アプリケーション内部にリソースを制御する実装が必要

- ✓ 処理するデータ量から必要なVM台数を計算
- ✓ VMを起動し、ネットワーク接続を行う
- ✓ 起動させたVMの準備が完了するまで待つ
- ...

✓ データ登録を行うために必要な設定をする

1. 自分の役割を確認する
2. DBのIPアドレスを調べる
3. 設定ファイルを修正し、アプリケーションを起動する



**プログラムからクラウドを操作しようとする
実装が複雑になってしまう**



**クラウドをプログラムから扱いやすくする
ための仕組みが必要**

- ▶ 「クラウドネイティブ」アプリケーション
- ▶ Real Application Centric Kernel(R・A・C・K)
- ▶ ケーススタディ powered by RACK
- ▶ RACKの今後の展望

ソフトウェア制御型次世代情報処理基盤技術開発プロジェクト

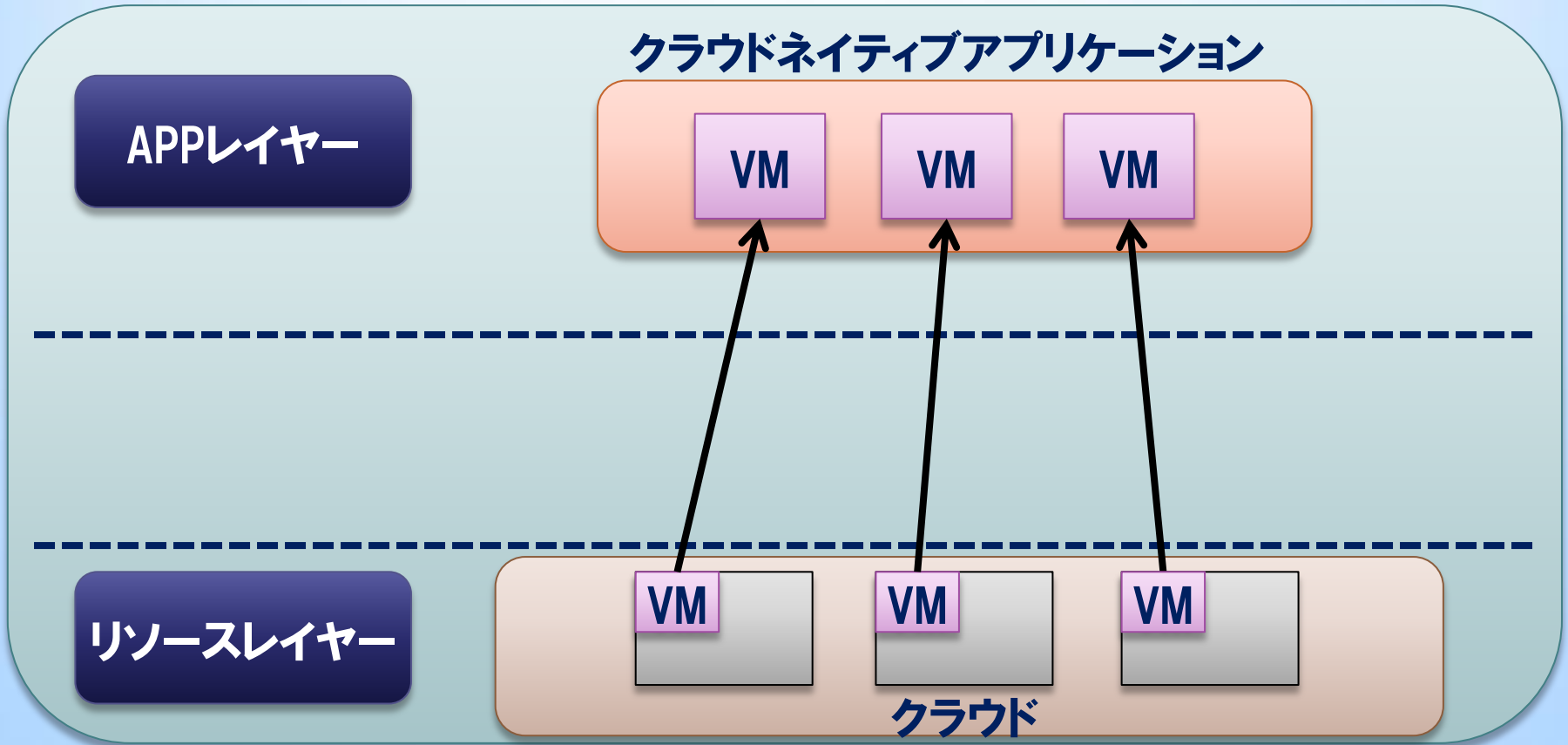
- ✓ クラウドの利活用促進を目的とした基盤技術開発
- ✓ クラウド基盤の効率的な制御による消費電力削減



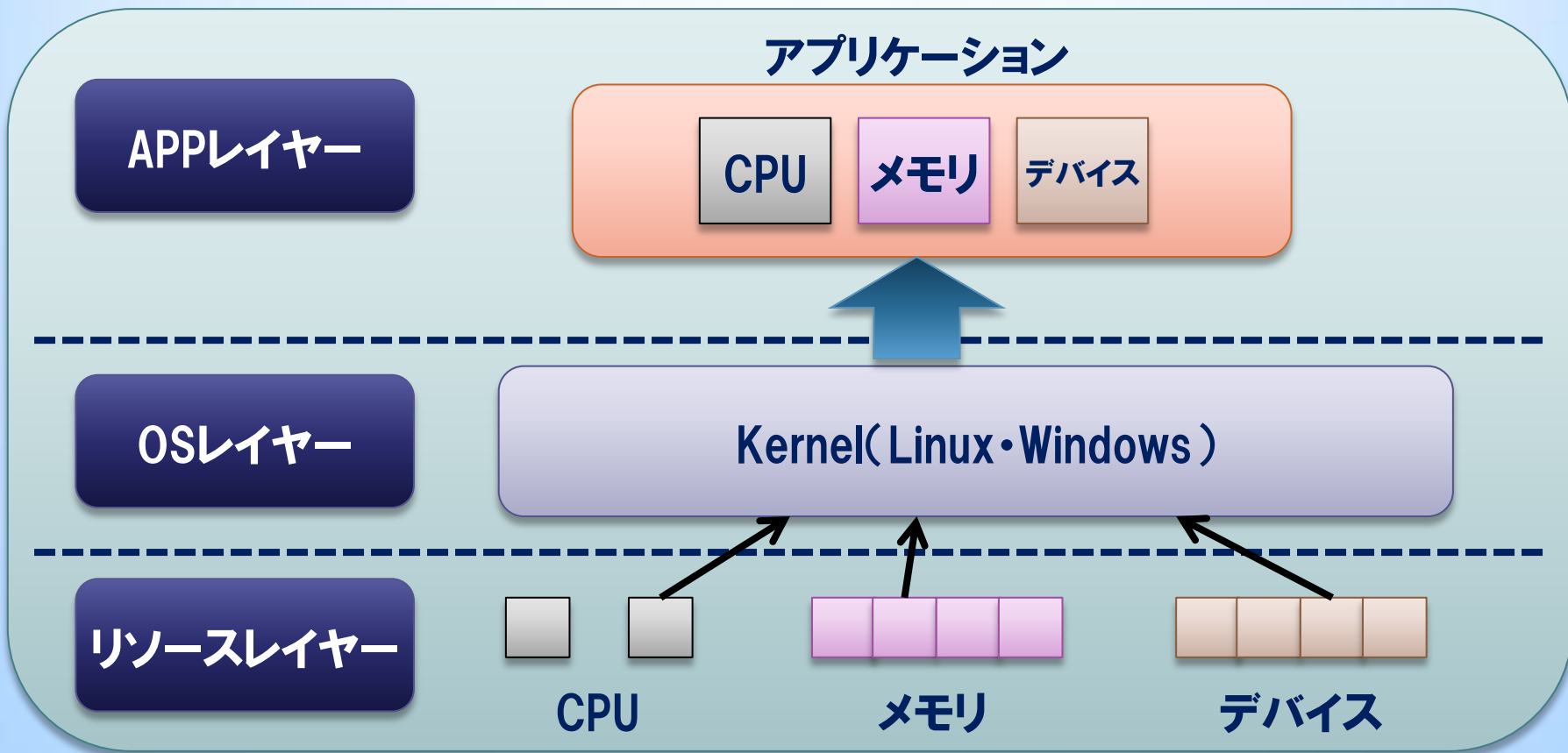
民間企業、研究開発機関等
(採択事業者: **CTC**、IIJ、TIS)

クラウドをプログラムから扱いやすくする 仕組みってなに？

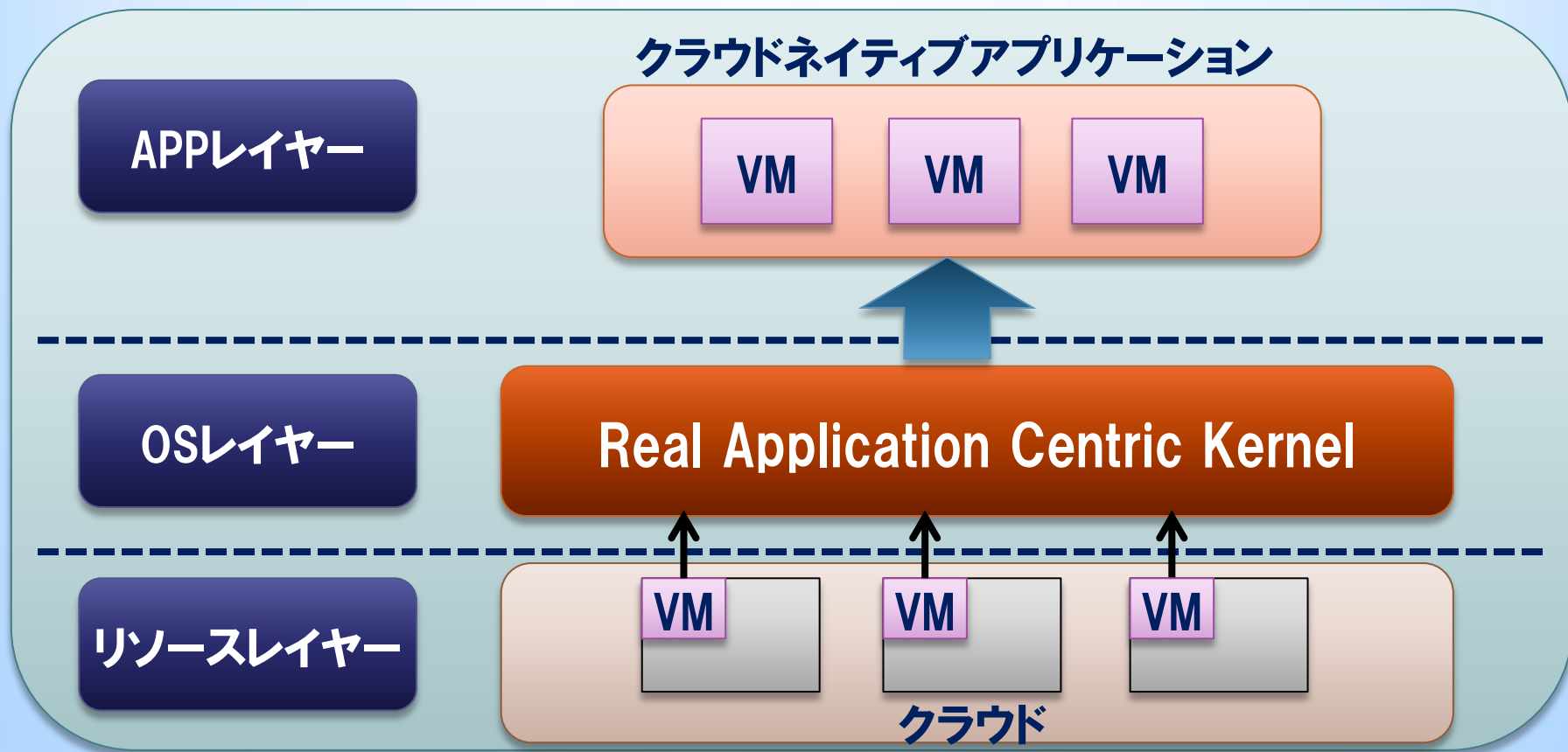
アプリケーションがリソースを確保する流れ



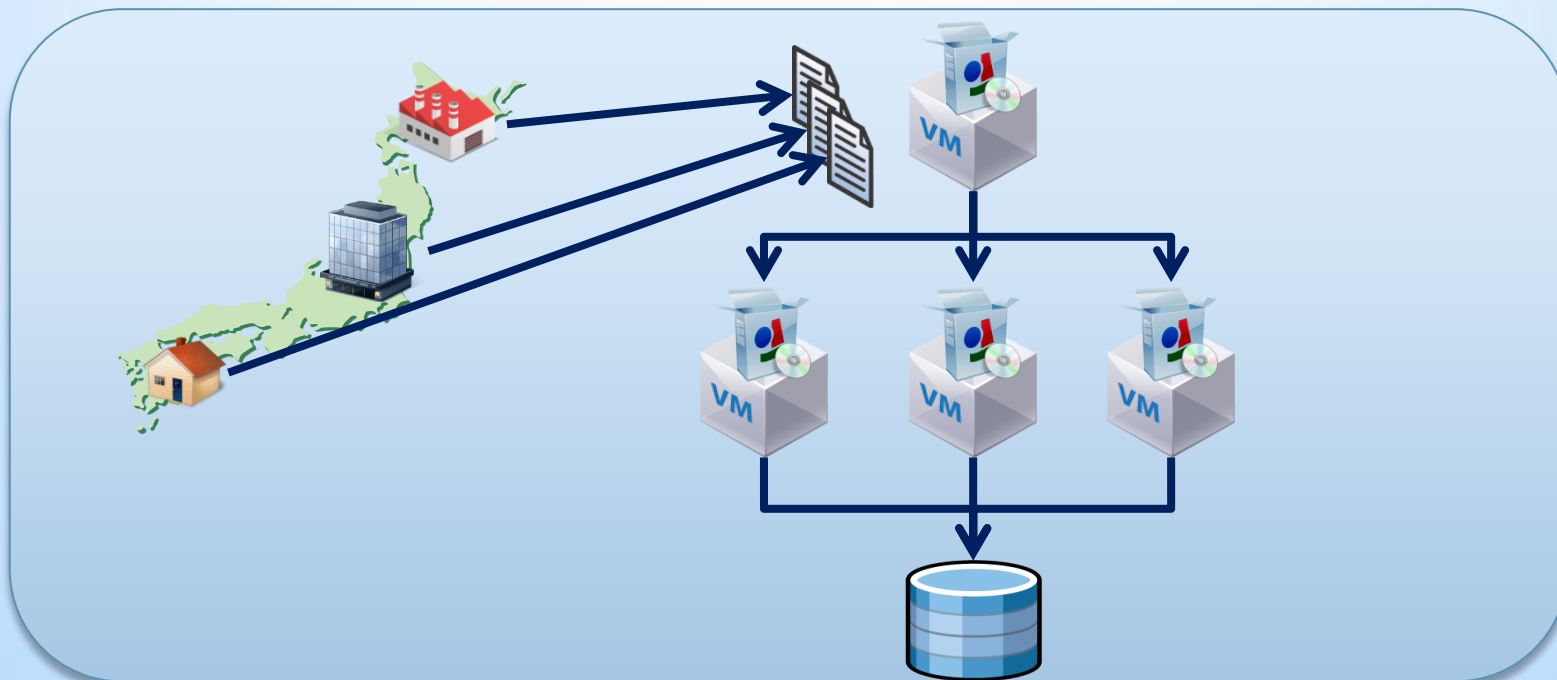
アプリケーションがリソースを確保する流れ



アプリケーションがリソースを確保する流れ



こんなアプリケーションが**簡単に作れる!**



RACK

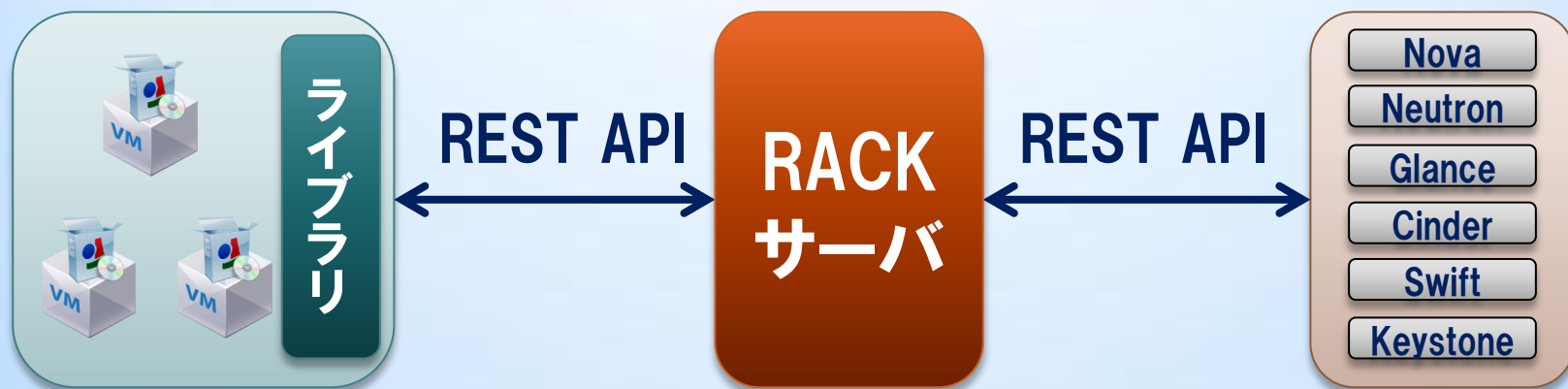


クラウド

RACKは何をしてくれるの？

クラウドネイティブ
アプリケーション

OpenStack



- ✓ VMに追加のデータ構造を持たせる(ライブラリ)
- ✓ プロセス間通信
- ✓ 共有メモリ・ファイルシステム

VMに追加のデータ構造を持たせる

VMにLinuxのプロセスモデルを適用

プロセスグループ(GID=XXX)

プロセス(PID=100)



Fork

Fork



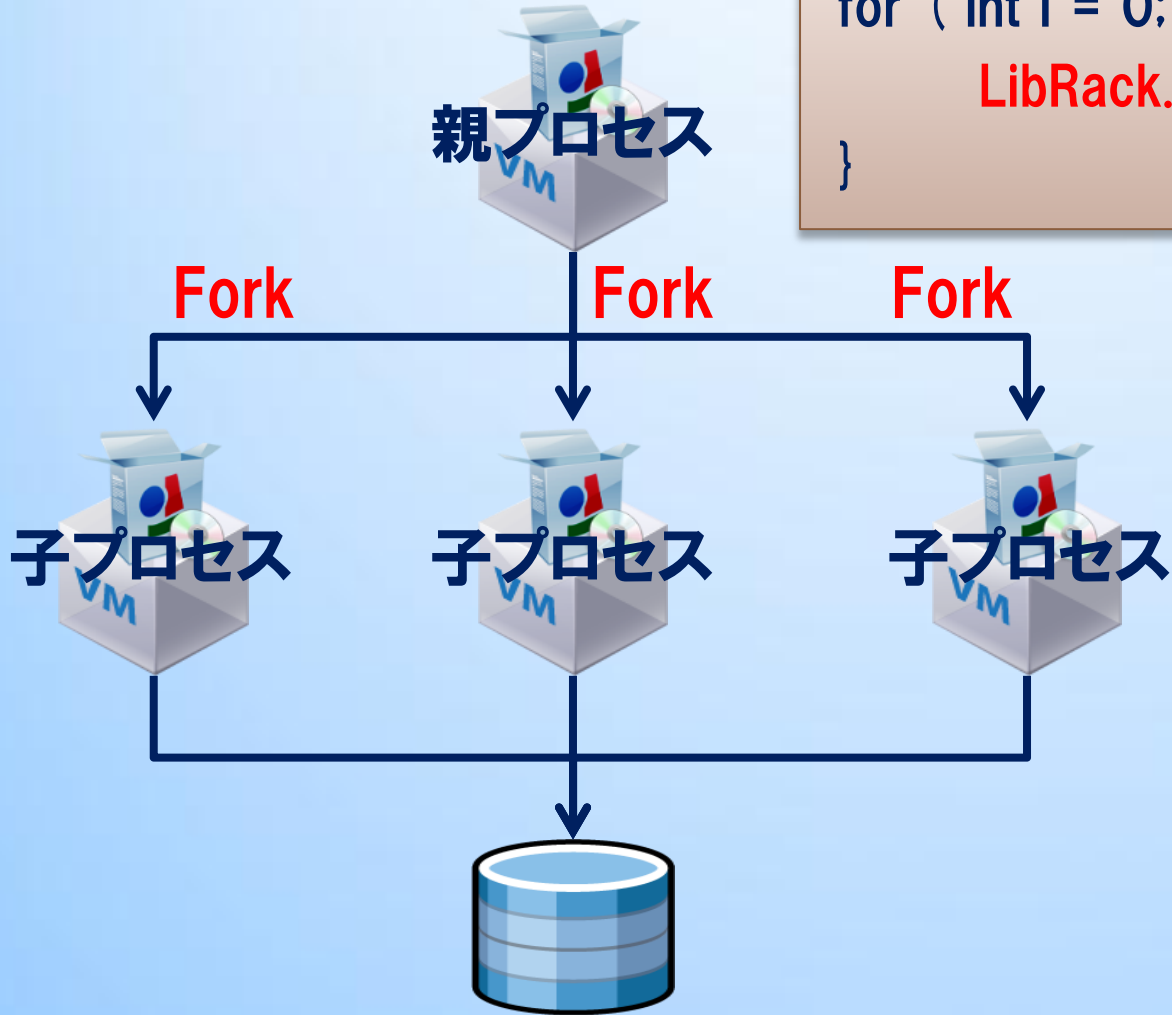
プロセス(PID=110
PPID=100)



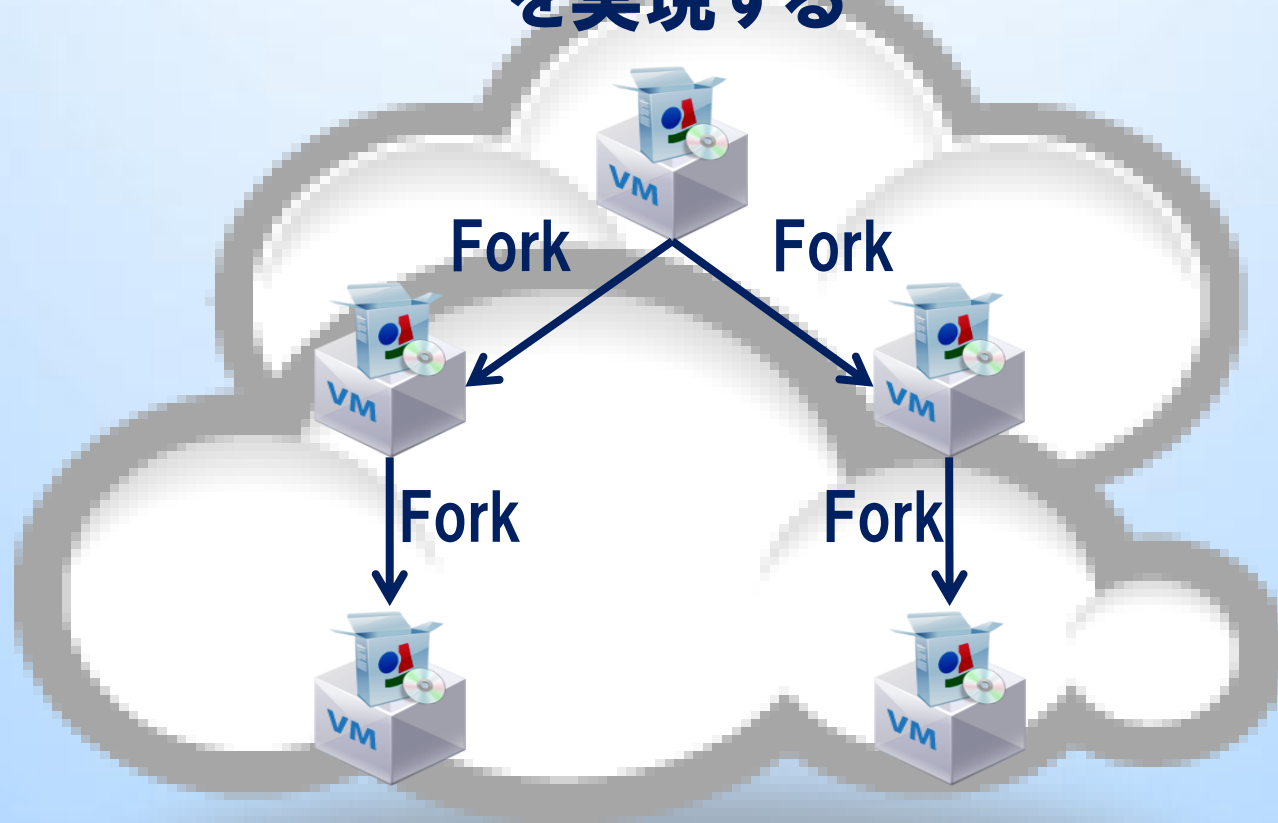
プロセス(PID=120
PPID=100)

Javaプログラム

```
import Librack  
for ( int i = 0; i < childNum; i++ ) {  
    LibRack.fork ();  
}
```



クラウド上で マルチプロセスマルチスレッドプログラミング を実現する

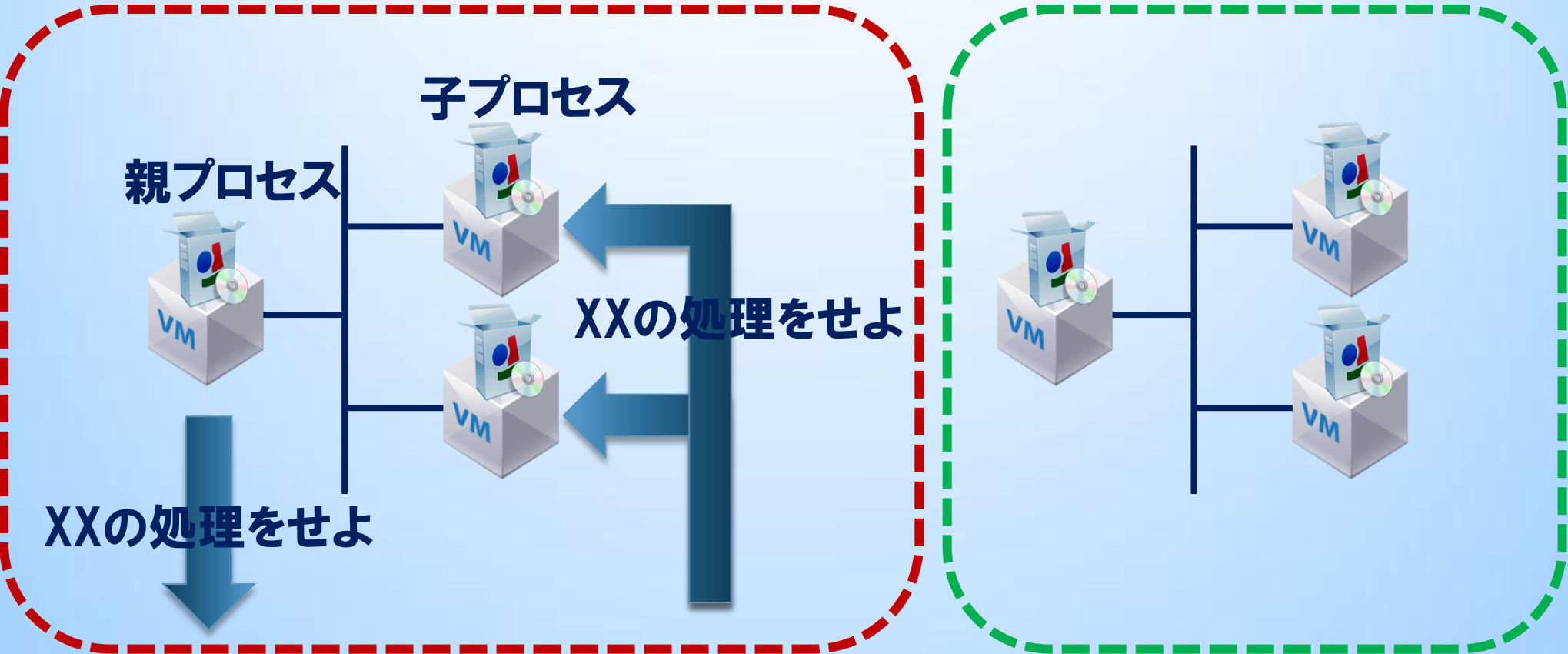


プロセス間通信

プロセスグループ内でメッセージのやり取りを行うための仕組み

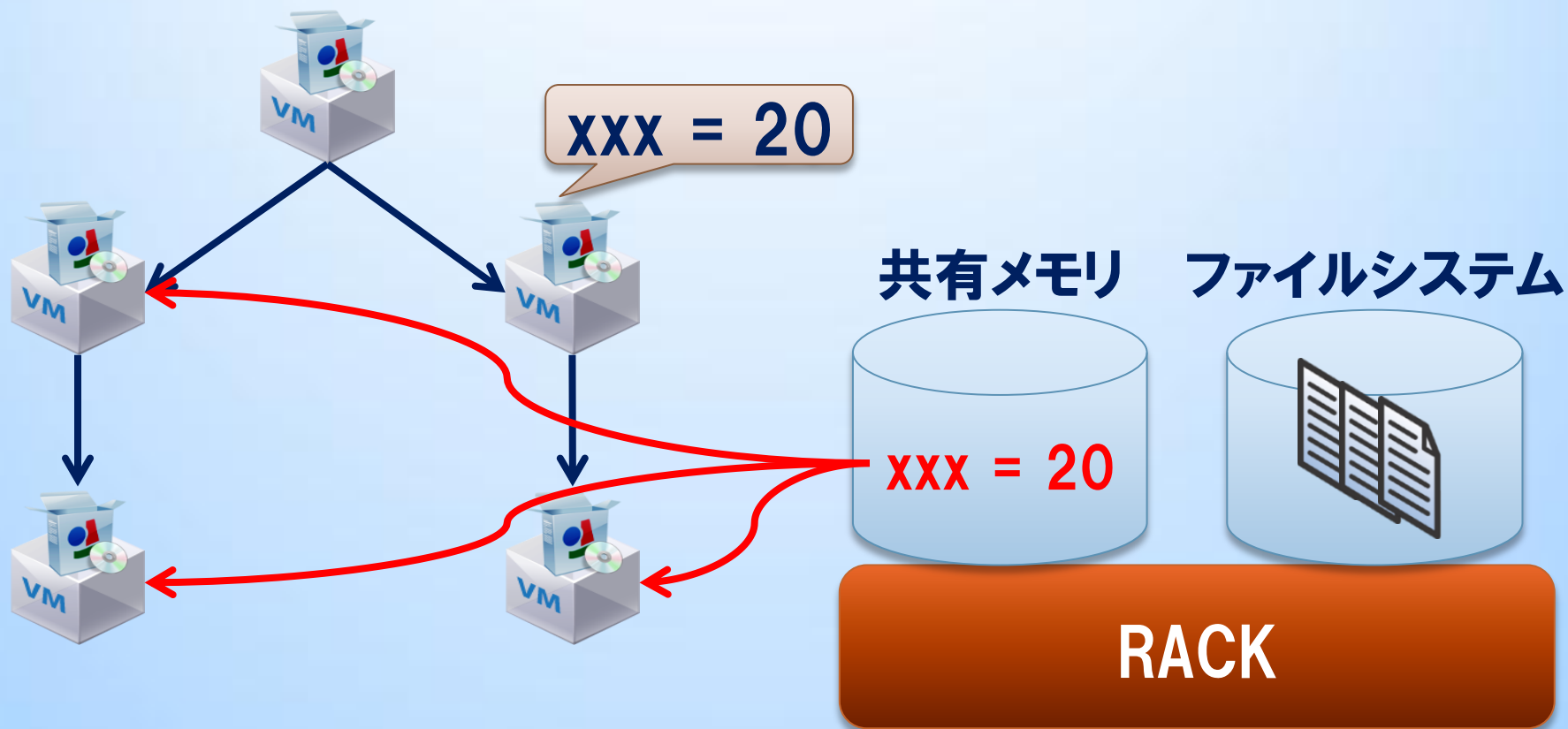
プロセスグループA

プロセスグループB



RACK

プロセスグループで共有するデータ・ファイルを格納



RACKでこんなに簡単になる

✓ 処理するデータ量から必要なVM台数を計算

✓ VMを起動し、ネットワーク接続を行う

✓ 起動されたVMの準備が完了するまで待つ

1. VMテンプレートイメージIDを調べる

2. ネットワークIDを調べる

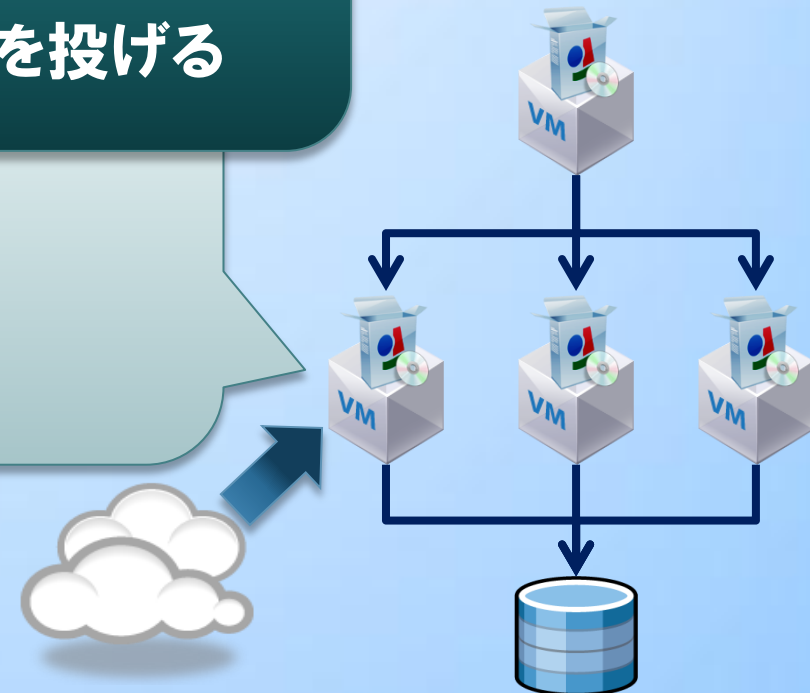
3. クラウドにVMを起動するリクエストを投げる

✓ データを

...

✓ 処理が終わったら自分を削除

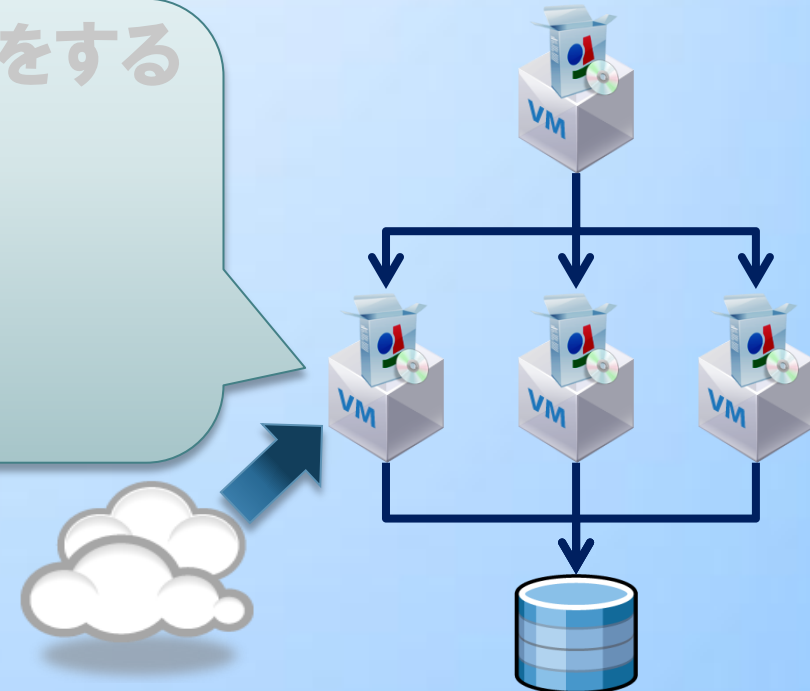
...



RACKがある場合

- ✓ 処理するデータ量から必要なVM台数を計算
 - ✓ VMを起動し、ネットワーク接続を行う
 - ✓ 起動されたVMの準備が完了するまで待つ
- ...
- 1. Forkメソッド(コマンド)を実行する**

- ✓ データ登録を行うために必要な設定をする
- ...
- ✓ 処理が終わったら自分を削除
- ...

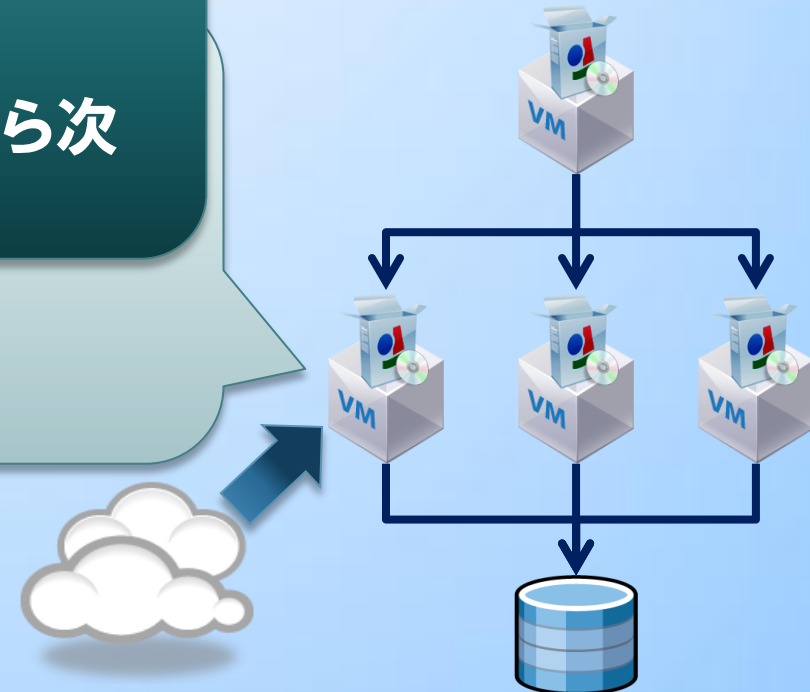


RACKがない場合

- ✓ 処理するデータ量から必要なVM台数を計算
- ✓ VMを起動し、ネットワーク接続を行う
- ✓ **起動させたVMの準備が完了するまで待つ**

1. 起動させたVMのステータスを定期的に確認する
2. VMのステータスがACTIVEになったら次の処理に移る

✓ 処理が終わったら自分を削除

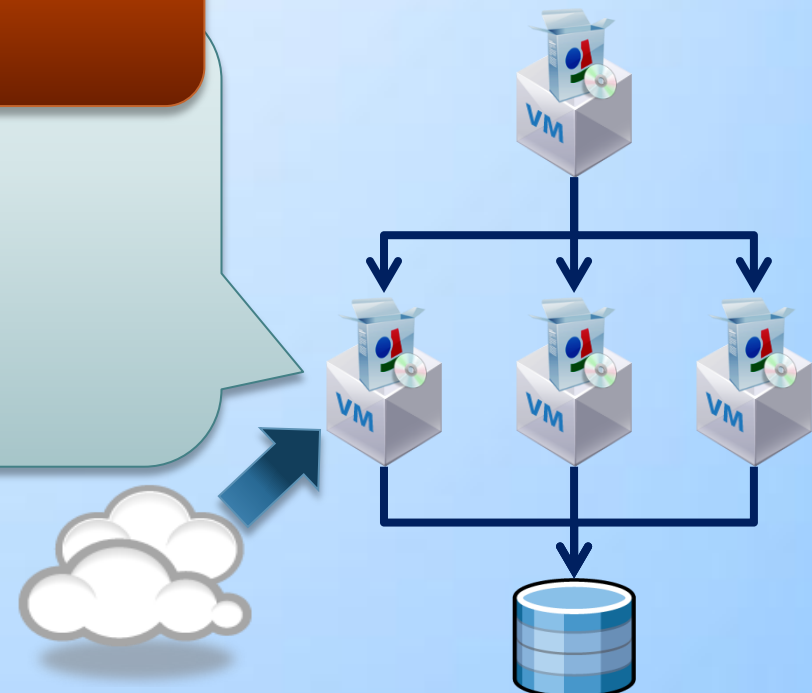


RACKがある場合

- ✓ 処理するデータ量から必要なVM台数を計算
- ✓ VMを起動し、ネットワーク接続を行う
- ✓ **起動させたVMの準備が完了するまで待つ**

1. 子プロセスから起動完了のメッセージがきたら次の処理に移る

- ✓ デー...
- ...
- ✓ 処理が終わったら自分を削除
- ...

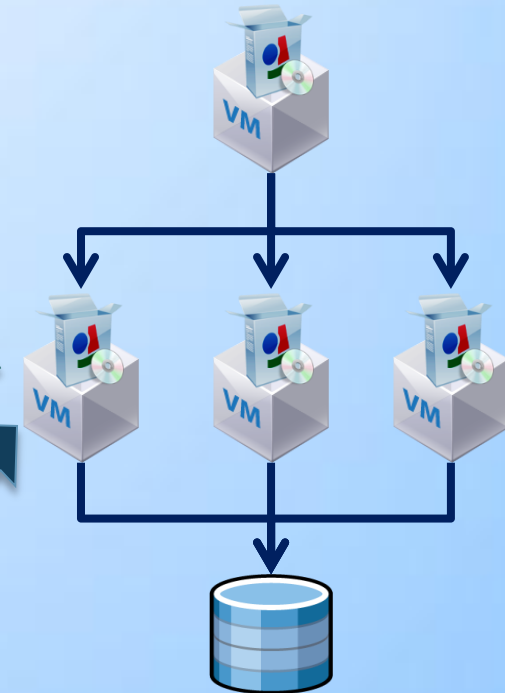


RACKがない場合

- ✓ 処理するデータ量から必要なVM台数を計算
- ✓ VMを起動し、ネットワーク接続を行う
- ✓ 起動させたVMの準備が完了するまで待つ
- ...

✓ データ登録を行うために必要な設定をする

1. 自分の役割を確認する
2. DBのIPアドレスを調べる
3. 設定ファイルを修正し、アプリケーションを起動する

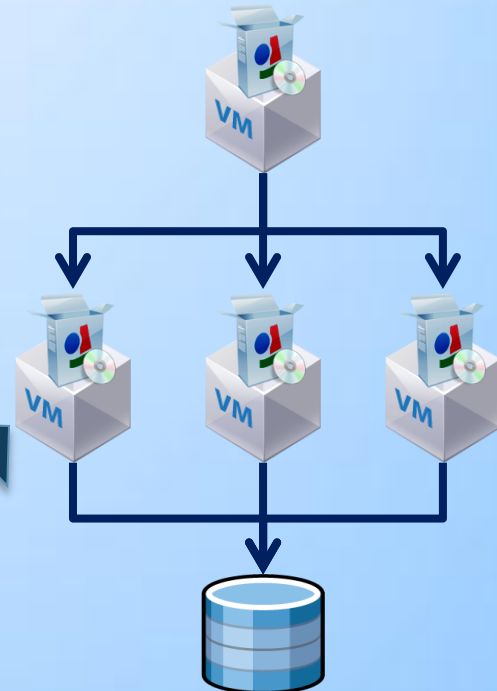


RACKがある場合

- ✓ 処理するデータ量から必要なVM台数を計算
- ✓ VMを起動し、ネットワーク接続を行う
- ✓ 起動させたVMの準備が完了するまで待つ
- ...

✓ データ登録を行うために必要な設定をする

1. 親プロセスか子プロセスかを判断
2. 子プロセス用の設定ファイルをファイルシステムから取得し、アプリケーションを起動する



- ✓ クラウドを**プログラムから扱いやすくするための仕組みを提供**
 - ➡ プロセスモデル、プロセス間通信、共有メモリ/ファイルシステム
 - RACKを使うことにより、**実装がシンプルになる**

- ▶ 「クラウドネイティブ」アプリケーション
- ▶ Real Application Centric Kernel(R・A・C・K)
- ▶ ケーススタディ powered by RACK
- ▶ RACKの今後の展望

スクラッチ開発

データ入力

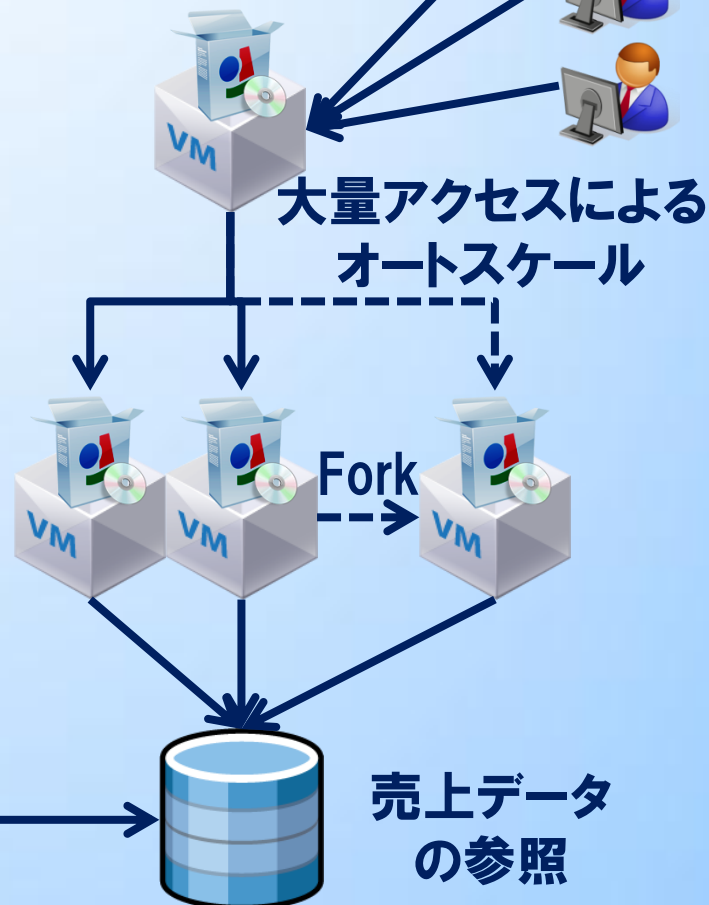
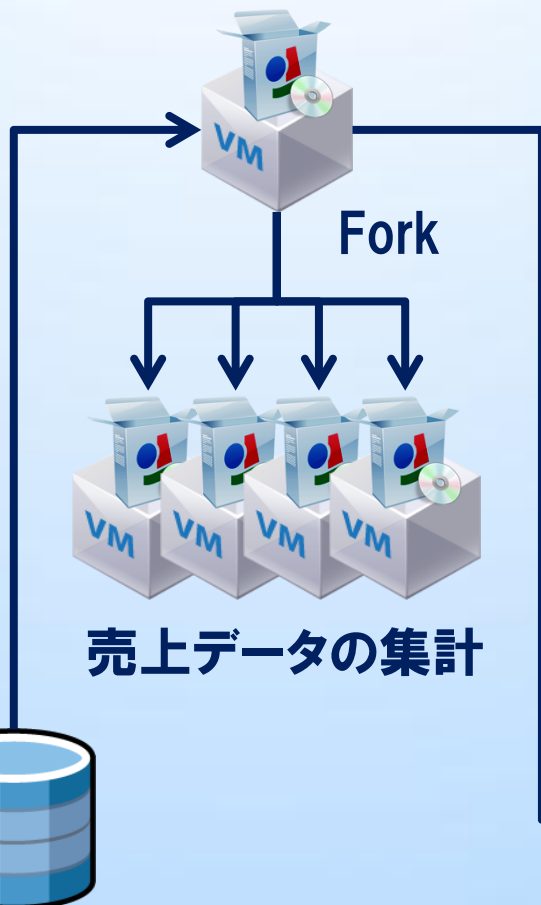
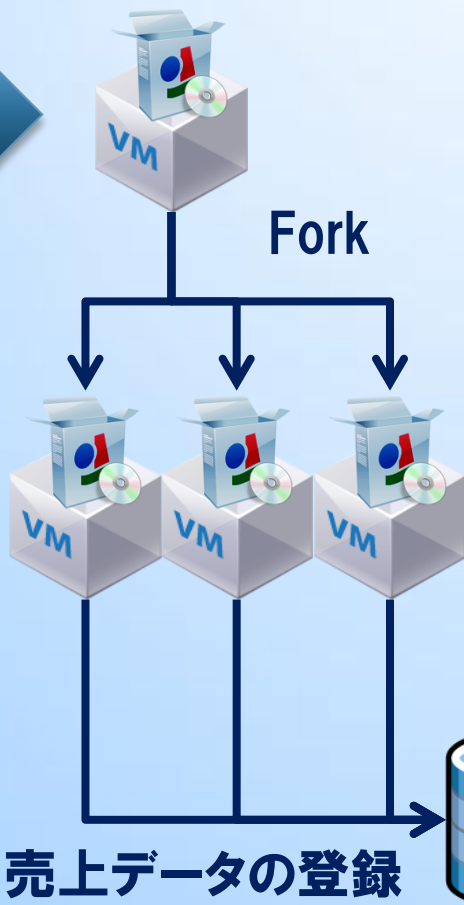
既存アプリとの組合せ

データ処理

既存アプリの追加開発

データ参照

大量の
売上データ

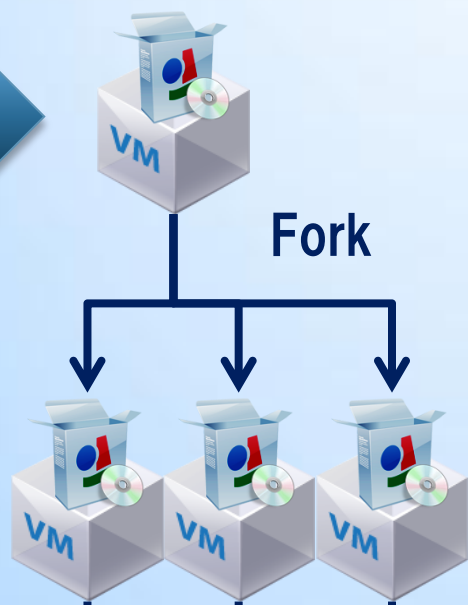


大量アクセスによる
オートスケール

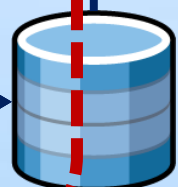
大量の
売上データ

スクラッチ開発

データ入力

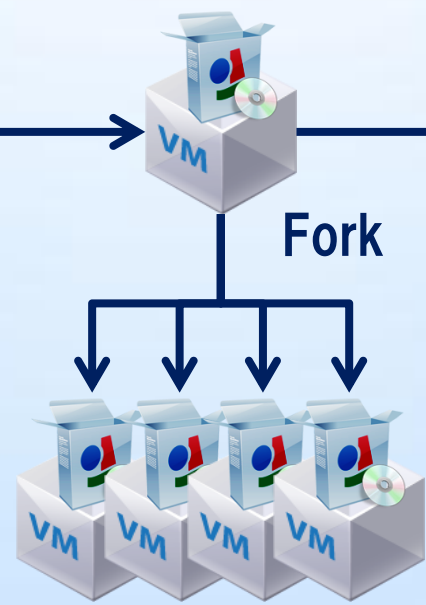


売上データの登録



既存アプリとの組合せ

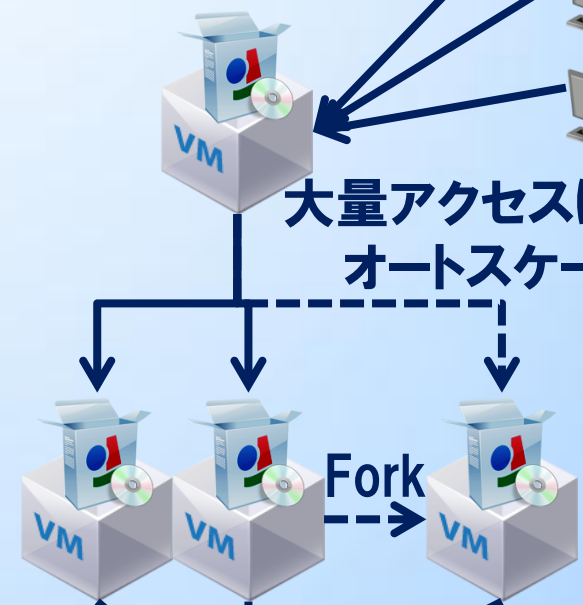
データ処理



売上データの集計

既存アプリの追加開発

データ参照

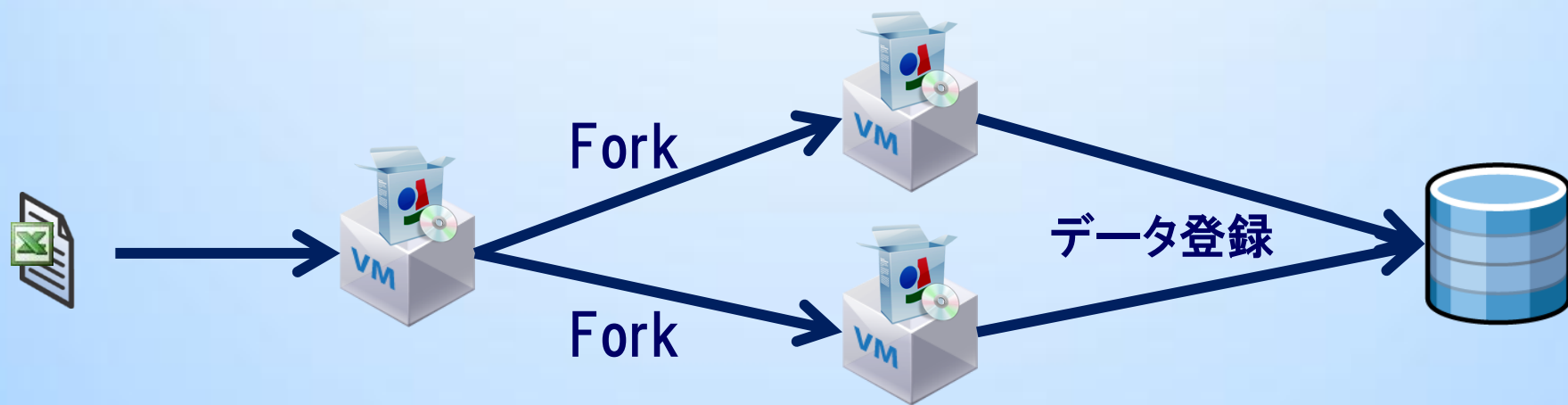


大量アクセスによる
オートスケール

売上データの
参照



- ✓ 受け取ったファイルの**データ量から処理に必要なプロセス数を計算しForkする**
- ✓ 起動した子プロセスはファイルデータの登録処理を開始する
- ✓ 全てのデータの登録処理が完了すると、子プロセスは消滅する



RACK

クラウド(OpenStack)

デモ1(実際には動画を使用しました)

アプリケーションの進捗を表示

The screenshot shows a web browser window with the title "Cloud Native Application Sho" and the URL "10.0.1.4:1111/index...". The main content area displays "CSVアプリ" and a progress indicator showing "0%". Below the progress indicator is a large, light gray gauge with a black needle pointing to the 0% mark. At the bottom of the browser window, a terminal window is visible with the prompt "kaneko@console:~" and the command "[kaneko@console ~]\$./csv-demo.sh". The terminal output shows "売上データファイルをアップロードします [y/n]:" followed by a red cursor. The terminal window also shows a status bar at the bottom with the text "[16:53] 0 console 1 rack".

デモプログラムの状態

The screenshot shows the OpenStack Dashboard interface for "Network Topology". The URL is "https://horizon.hpcloud.com/project/network_topology/". The page title is "Network Topology" and it includes user information "cto-cloud-native-2014" and "Sign Out" and "Support" links. There are buttons for "Launch Instance", "Create Network", and "Create Router". The main content area displays a network topology diagram with four vertical bars representing networks: "Ext Net" (blue), "87358913-bf0b-4500-876d-bb239781555d" (orange), "cto-cloud-native-2014-network" (green), and "2c56d701-ec33-4f6a-9d74-ff1a2dd4ee65" (red). The diagram shows connections between these networks and various components. The bottom right corner has a "Chat with HP Cloud" button.

OpenStack Dashboard

デモ1(実際には動画を使用しました)

Cloud Native Application Sho x
10.0.1.4:1111/index.html

CSVアプリ

0%

```
kaneko@console:~  
[kaneko@console ~]$ ./csv-demo.sh  
売上データファイルをアップロードします [y/n]: y  
売上データ登録中...
```

ファイルのアップロードを開始します

[16:53] 0 console 1 rack

hp Network Topology - Cloud Ac x
https://horizon.hpcloud.com/project/network_topology/

cto-cloud-native-2014 Sign Out Support

Small Normal Launch Instance Create Network Create Router

Ext-Net
8755943-bfab-4600-876d-bb239781555d 10.0.1.0/24
etc-cloud-native-2014-network 10.0.0.0/24
2e3ed701-eec3-448a-947-a2dd4ee685 10.0.3.0/24

プロセスがファイルを受信

Chat with HP Cloud

デモ1(実際には動画を使用しました)

Cloud Native Application Sho x
10.0.1.4:1111/index.html

CSVアプリ

0%

```
kaneko@console:~  
[kaneko@console ~]$ ./csv-demo.sh  
売上データファイルをアップロードします [y/n]: y  
売上データ登録中...  
[16:53] 0 console 1 rack
```

hp Network Topology - Cloud Ac x
https://horizon.hpcloud.com/project/network_topology/
cto-cloud-native-2014 Sign Out Support

Small Normal Launch Instance Create Network Create Router

アップロードされたファイルのファイルサイズをもとに、処理に必要なリソースを計算し、子プロセスが”Fork”されます。

Chat with HP Cloud

デモ1(実際には動画を使用しました)

The screenshot displays a multi-pane interface. The top-left pane shows a web browser at `10.0.1.4:1111/index.html` with the title "CSVアプリ" and a progress indicator at "0%". The bottom-left pane is a terminal window with the following text:

```
kaneko@console:~  
[kaneko@console ~]$ ./csv-demo.sh  
売上データファイルをアップロードします [y/n]: y  
売上データ登録中...  
[16:55] 0 console 1 rack
```

The top-right pane shows a "Network Topology" dashboard with a diagram of four vertical racks (blue, orange, green, red). A callout box points to the orange rack with the text: "プロセスが起動すると一斉にCSVファイル処理が始まります。" (When the process starts, CSV file processing begins simultaneously.)

デモ1(実際には動画を使用しました)

Cloud Native Application Sho x

10.0.1.4:1111/index.htm

CSVアプリ

10%

”Fork”された子プロセスによる CSVファイル処理の進捗を示しています。

```
kaneko@console:~  
[kaneko@console ~]$ ./csv-demo.sh  
売上データファイルをアップロードします [y/n]: y  
売上データ登録中...  
[16:55] 0 console 1 rack
```

hp Network Topology - Cloud Ar x

https://horizon.hpcloud.com/project/network_topology/

Network Topology

cto-cloud-native-2014 Sign Out Support

Small Normal Launch Instance Create Network Create Router

Chat with HP Cloud

デモ1(実際には動画を使用しました)

Cloud Native Application Sho x

10.0.1.4:1111/index.htm

CSVアプリ

80%

”Fork”された子プロセスによる CSVファイル処理の進捗を示しています。

hp Network Topology - Cloud Ar x

https://horizon.hpcloud.com/project/network_topology/

Network Topology

cto-cloud-native-2014 Sign Out Support

Small Normal Launch Instance Create Network Create Router

kaneko@console:~

```
[kaneko@console ~]$ ./csv-demo.sh  
売上データファイルをアップロードします [y/n]: y  
売上データ登録中...  
[16:57] 0 console 1 rack
```

Chat with HP Cloud

デモ1(実際には動画を使用しました)

The image displays a composite screenshot of a cloud application interface. On the left, a browser window shows a progress bar for a CSV application, labeled "CSVアプリ" and "100%". A red gauge indicates the completion status. A speech bubble above the gauge contains the text "処理が終了しました。" (Processing completed). Below the browser, a terminal window shows the execution of a script: `[kaneko@console ~]$./csv-demo.sh`, with output: `売上データファイルをアップロードします [y/n]: y` and `売上データ登録中...[OK]`. The bottom status bar of the terminal shows `[16:57] 0 console 1 rack`. On the right, a "Network Topology" interface is visible, featuring a diagram with four vertical racks (blue, orange, green, red) and a network connection between the blue and orange racks. The interface includes buttons for "Launch Instance", "Create Network", and "Create Router".

デモ1(実際には動画を使用しました)

Cloud Native Application Sho x

10.0.1.4:1111/index.html

CSVアプリ

100%

```
kaneko@console:~  
[kaneko@console ~]$ ./csv-demo.sh  
売上データファイルをアップロードします [y/n]: y  
売上データ登録中...[OK]  
[kaneko@console ~]$
```

[16:57] 0 console 1 rack

Network Topology - Cloud Ar x

https://horizon.hpcloud.com/project/network_topology/

cto-cloud-native-2014 Sign Out Support

Small Normal Launch Instance Create Network Create Router

処理を終えた子プロセスたちは自身をkillし、もとの2プロセスの状態に戻ります。(開始～終了まで約2分)

Chat with HP Cloud

- ✓ **検証対象:**
CSVファイル処理アプリケーション
- ✓ **検証内容:**
データ量の違うファイルを処理させ、
その実行時間を測定する
- ✓ **検証環境:**
“HP Public Cloud” <http://www.hpcloud.com/>



OpenStackベースのPublicクラウドサービス
OpenStackのAPIを利用可能

RACKが利用可能



✓ 膨大なデータを高速に処理することが可能

➡ クラウドという膨大なリソース群からリソースを調達できるため、大規模データ処理でも高いパフォーマンスが得られる

✓ リソースの効率的な利用ができる

➡ 必要な時だけリソースを消費する為、コストを抑えることができる

スクラッチ開発

データ入力

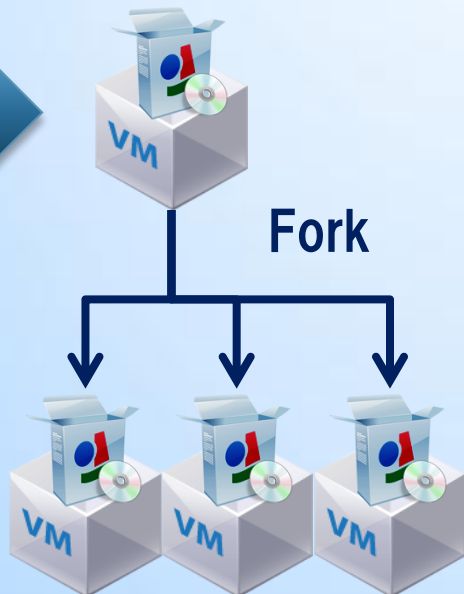
既存アプリとの組合せ

データ処理

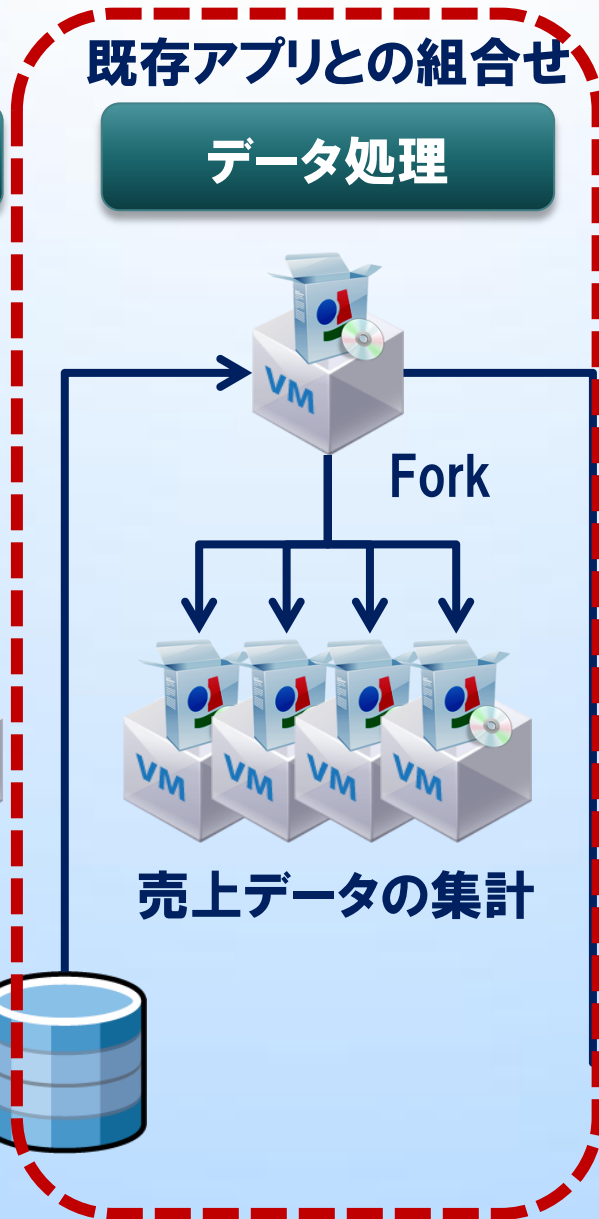
既存アプリの追加開発

データ参照

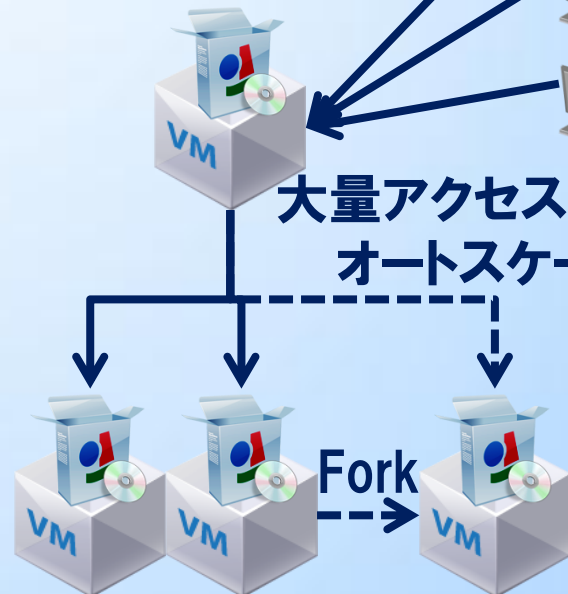
大量の
売上データ



売上データの登録



売上データの集計

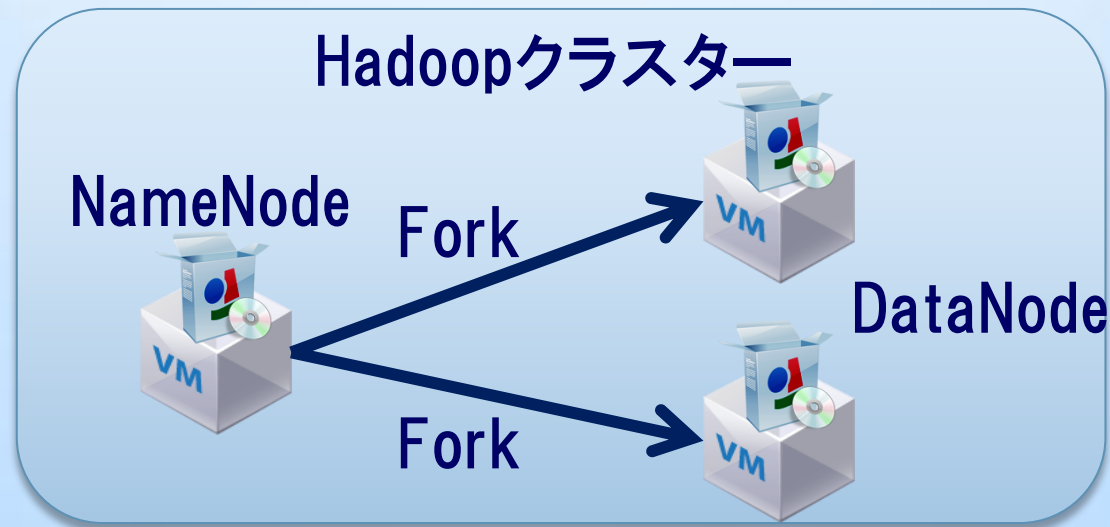


売上データの
参照



大量アクセスによる
オートスケール

- ✓ 親プロセス(NameNodeプロセス)を起動する
- ✓ 親プロセスがForkし、子プロセス(DataNodeプロセス)が複数起動する
- ✓ Hadoop環境をデプロイし、バッチプログラムを実行する
- ✓ プロセスグループを削除する



RACK

クラウド(OpenStack)


デモ2(実際には動画を使用しました)

Cloud Native Application Sho x

10.0.1.4:1111/index.html

バッチアプリ

0%



```
kaneko@console:~  
[kaneko@console ~]$ ./asakusa_hadoop.sh  
プロセスグループを作成します...  
  
[17:35] 0 console 1 rack
```

バッチ処理を開始します。

Network Topology - Cloud Ar x

https://horizon.hpcloud.com/p

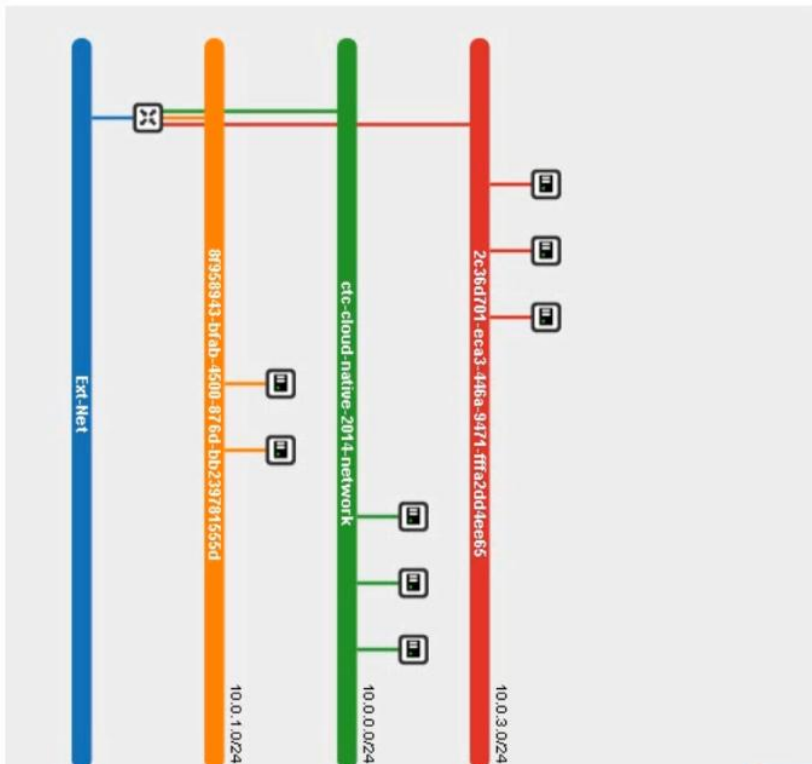
Network Topology

cto-cloud-native-2014

Out Support

Small Normal

Launch Instance Create Network Create Router



Ext-Net

8f95819-43-b7ad-4500-876d-bb239781555d 10.0.1.0/24

cto-cloud-native-2014-network 10.0.0.0/24

2c36d701-eca3-498a-9d74-ff82add4ee65 10.0.3.0/24

Chat with HP Cloud

バッチ処理なので、普段はリソースが起動していません。

デモ2(実際には動画を使用しました)

Cloud Native Application Sho x

10.0.1.4:1111/index.html

バッチアプリ

0%

```
kaneko@console:~  
[kaneko@console ~]$ ./asakusa_hadoop.sh  
プロセスグループを作成します...[OK]  
バッチプログラムファイルを共有ストレージにアップロードしま  
す...[OK]  
Hadoopクラスターをデプロイします...  
[17:35] 0 console 1 rack
```

Network Topology - Cloud Ac x

https://horizon.hp

Network Topology

Small Normal

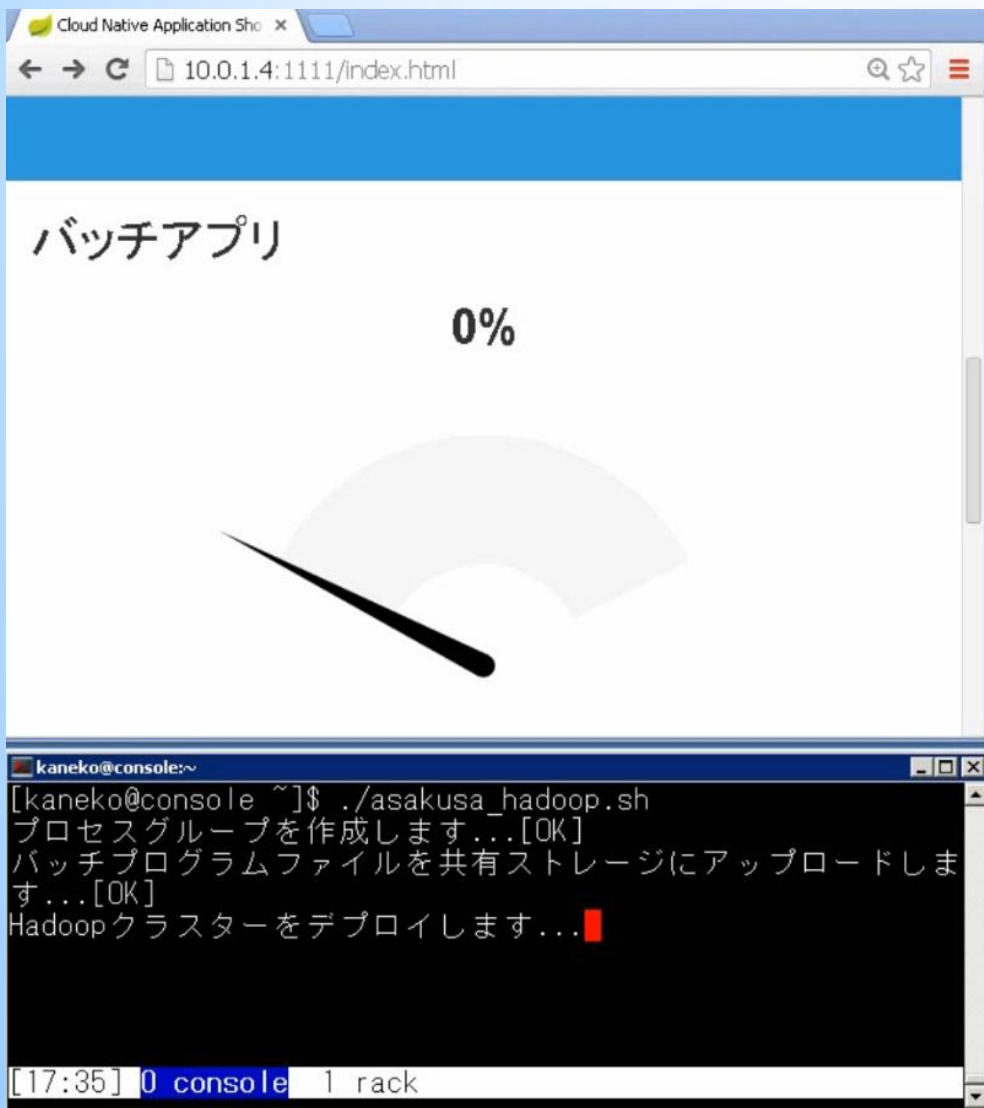
Launch Force Create Network Create Router

仮想ネットワークが作成されまし
た。

The diagram illustrates a network topology with the following components and connections:

- Ext-Net** (Blue vertical bar) is connected to the central network via a switch icon.
- Central Network** (Green vertical bar, labeled "ctc-cloud-native-2014-network") is connected to three server racks: 10.0.1.0 (Orange), 10.0.2.0 (Red), and 10.0.3.0 (Purple).
- Server Racks** are connected to the central network via switch icons. The racks are labeled with their respective IP ranges: 10.0.1.0, 10.0.2.0, and 10.0.3.0.

デモ2(実際には動画を使用しました)



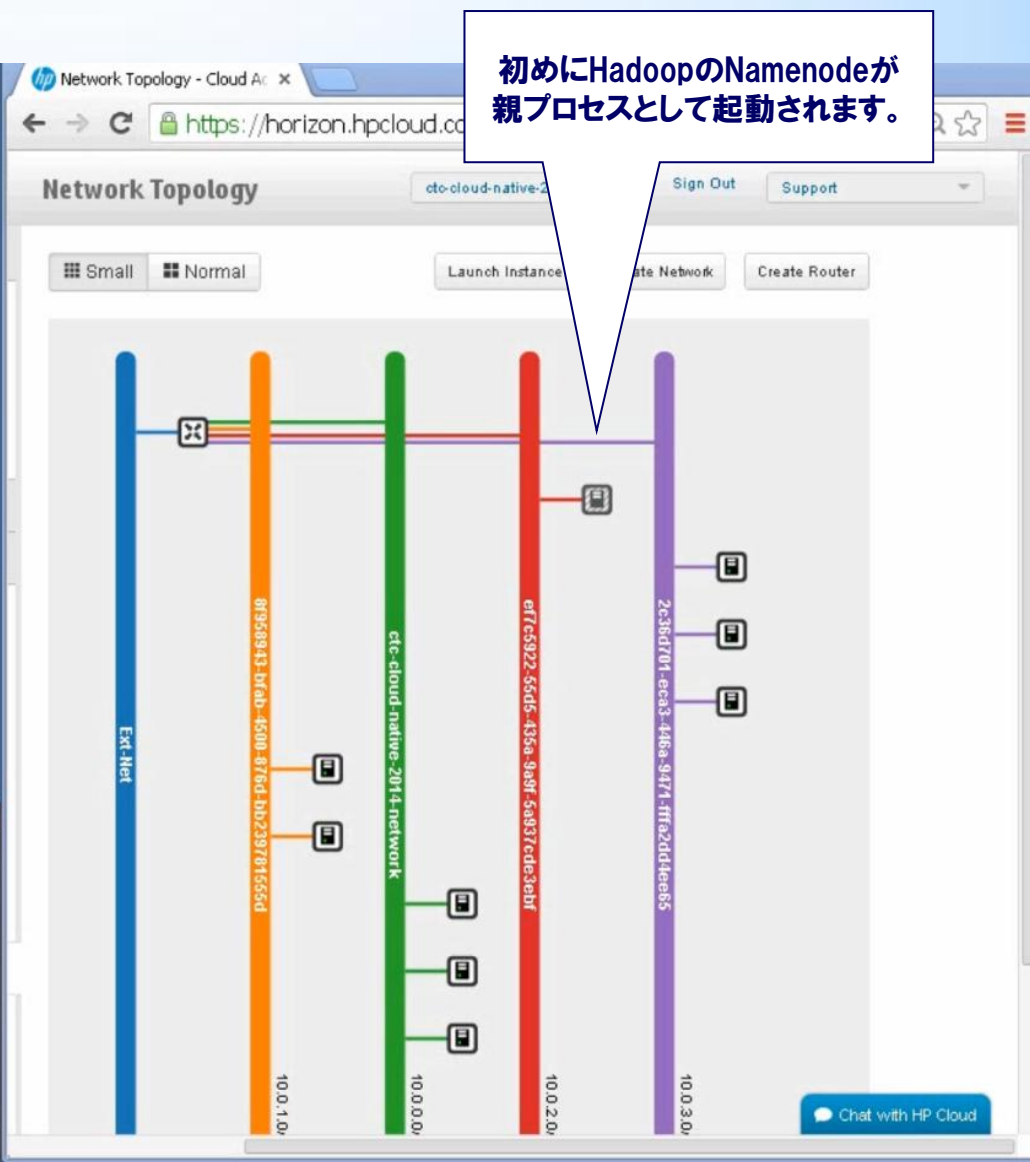
Cloud Native Application Sho x

10.0.1.4:1111/index.html

バッチアプリ

0%

```
kaneko@console:~  
[kaneko@console ~]$ ./asakusa_hadoop.sh  
プロセスグループを作成します...[OK]  
バッチプログラムファイルを共有ストレージにアップロードしま  
す...[OK]  
Hadoopクラスターをデプロイします...  
[17:35] 0 console 1 rack
```



Network Topology - Cloud Ar x

https://horizon.hpcloud.co

Network Topology

cto-cloud-native-2

Sign Out Support

Small Normal

Launch Instance Create Network Create Router

Ext-Net

8f950913-bfab-4500-870d-b0239701555d

ctc-cloud-native-2014-network

e7f63922-56d6-436a-9a9f-6a837ede3e9f

2c36d701-eca3-4f6a-947f-ff0a2dd4ee85

10.0.1.0/24

10.0.2.0/24

10.0.3.0/24

Chat with HP Cloud

初めにHadoopのNamenodeが親プロセスとして起動されます。

デモ2(実際には動画を使用しました)

Cloud Native Application Sho x

10.0.1.4:1111/index.html

バッチアプリ

0%

```
kaneko@console:~  
[kaneko@console ~]$ ./asakusa_hadoop.sh  
プロセスグループを作成します...[OK]  
バッチプログラムファイルを共有ストレージにアップロードしま  
す...[OK]  
Hadoopクラスターをデプロイします...  
[17:37] 0 console 1 rack
```

hp Network Topology - Cloud Ac x

https://horizon.hpcloud.com/project/network_topology/

Network Topology

cto-cloud-native-2014 Sign Out Support

Small Normal Launch Instance Create Network Create Router

次にHadoopのDataNodeが親プロセスから”Fork”されます

Chat with HP Cloud

デモ2(実際には動画を使用しました)

必要なプロセスが起動すると、実際の処理が開始されます。
このデモでは Asakusa Frameworkを使った処理プログラムが投入されます。

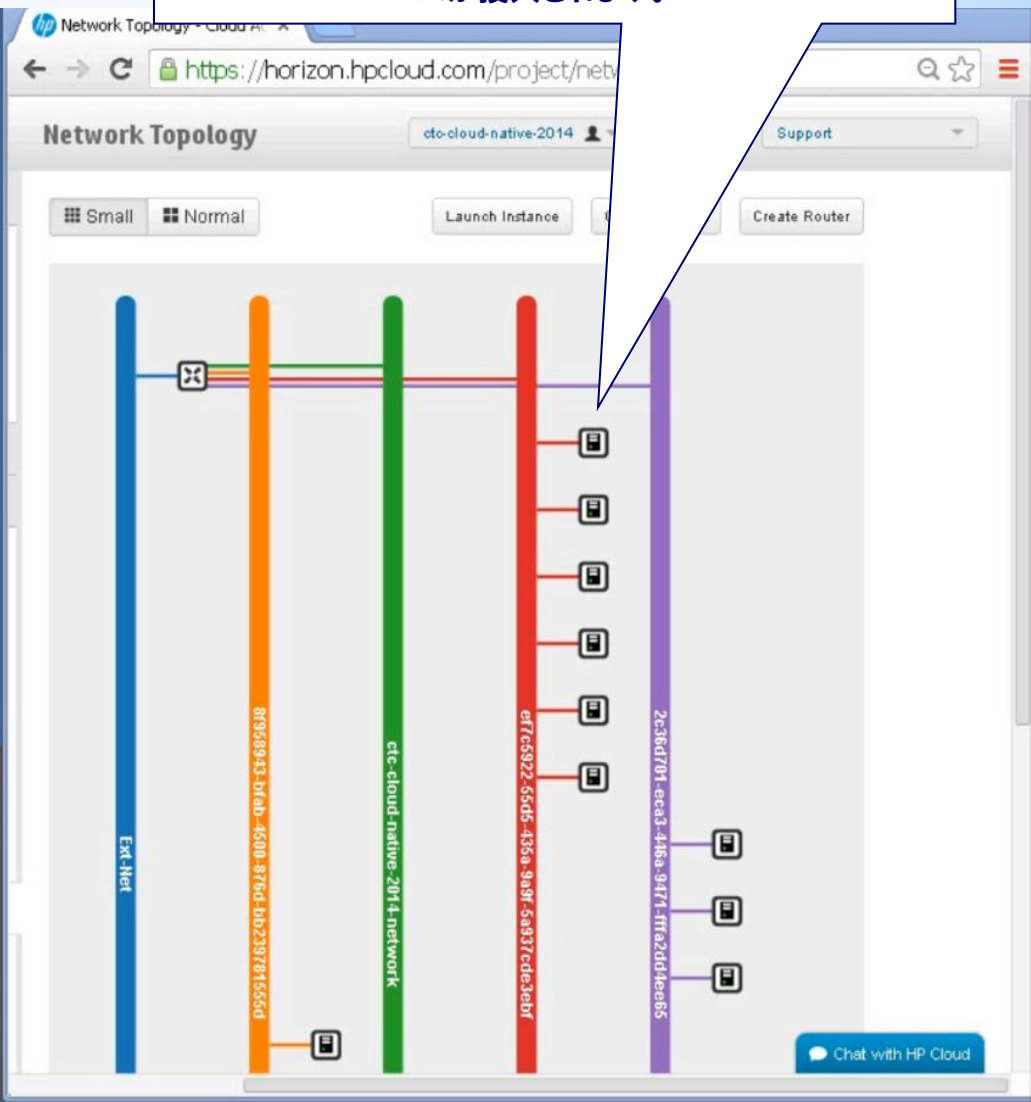
Cloud Native Application Sho x
10.0.1.4:1111/index.html

バッチアプリ

0%

A progress indicator showing 0% completion for a batch application. The indicator is a grey arc with a black needle pointing to the 0% mark.

```
kaneko@console:~  
[kaneko@console ~]$ ./asakusa_hadoop.sh  
プロセスグループを作成します...[OK]  
バッチプログラムファイルを共有ストレージにアップロードしま  
す...[OK]  
Hadoopクラスターをデプロイします...  
[17:38] 0 console 1 rack
```



デモ2(実際には動画を使用しました)

Hadoopでデータが処理されています。

Cloud Native Application Sho x
10.0.1.4:1111/index.html

バッチアプリ

45%

```
kaneko@console:~  
[kaneko@console ~]$ ./asakusa_hadoop.sh  
プロセスグループを作成します...[OK]  
バッチプログラムファイルを共有ストレージにアップロードしま  
す...[OK]  
Hadoopクラスターをデプロイします...[OK]  
バッチプログラムを実行中です...  
[17:39] 0 console 1 rack
```

Network Topology - Cloud Ac x
https://horizon.hpcloud.com/project/network_topology/

Network Topology

cto-cloud-native-2014 Sign Out Support

Small Normal Launch Instance Create Network Create Router

Ext-Net
81958943-b7ad-4500-876d-bb239781555d
ctc-cloud-native-2014-network
e17c5922-55df-435a-99f-5a937edc3e1f
2c36d711-eca3-448a-9471-ff42dd4ee85

Chat with HP Cloud

デモ2(実際には動画を使用しました)

全ての処理が終わると、バッチプログラム用のリソースは不要になります。

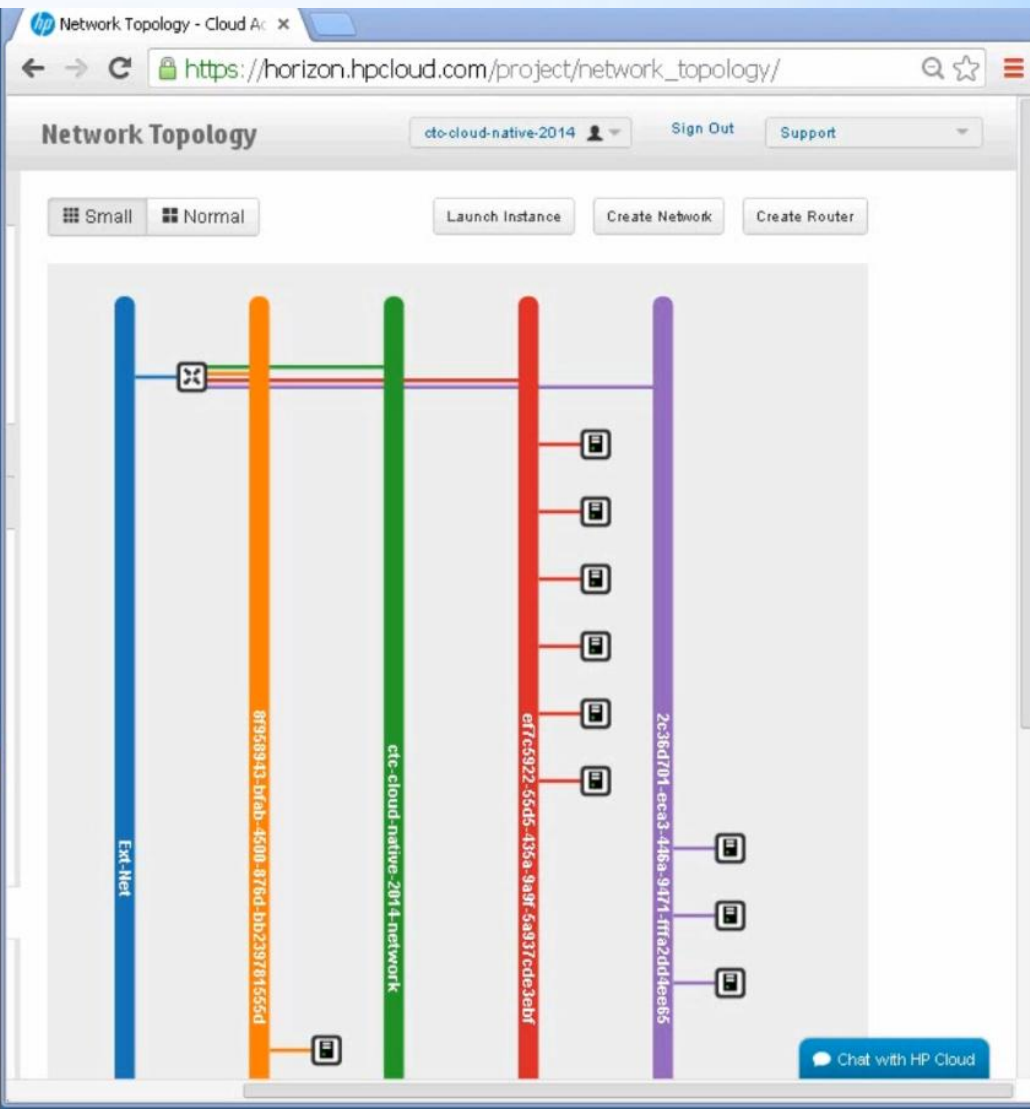
Cloud Native Application Sho x

10.0.1.4:1111/index.html

バッチアプリ

100%

```
kaneko@console:~  
[kaneko@console ~]$ ./asakusa_hadoop.sh  
プロセスグループを作成します...[OK]  
バッチプログラムファイルを共有ストレージにアップロードしま  
す...[OK]  
Hadoopクラスターをデプロイします...[OK]  
バッチプログラムを実行中です...[OK]  
Hadoopクラスターを削除します...  
[17:41] 0 console 1 rack
```



デモ2(実際には動画を使用しました)

Cloud Native Application Sho x

10.0.1.4:1111/index.html

バッチアプリ

100%

```
kaneko@console:~  
[kaneko@console ~]$ ./asakusa_hadoop.sh  
プロセスグループを作成します...[OK]  
バッチプログラムファイルを共有ストレージにアップロードしま  
す...[OK]  
Hadoopクラスターをデプロイします...[OK]  
バッチプログラムを実行中です...[OK]  
Hadoopクラスターを削除します...  
[17:41] 0 console 1 rack
```

Network Topology - Cloud Ar x

https://horizon.hpcloud.com/pr...work_topology/

Network Topology

cto-cloud-native

Sign Out Support

Small Normal

Launch Instan Create Network Create Router

Ext-Net

8f958943-bfab-4608-876d-bd239781556d

cto-cloud-native-2014-network

e7f65922-55d6-435a-9a9f-5a937cde2abf

2c36d701-ec93-448a-9474-f7f2dd4ee65

Chat with HP Cloud

まずプロセスが停止します。

デモ2(実際には動画を使用しました)

次いで仮想ネットワークが削除され、処理が終了となります。
(開始～終了まで約3分)

Cloud Native Application Sho x

10.0.1.4:1111/index.html

バッチアプリ

100%

```
kaneko@console:~  
[kaneko@console ~]$ ./asakusa_hadoop.sh  
プロセスグループを作成します...[OK]  
バッチプログラムファイルを共有ストレージにアップロードしま  
す...[OK]  
Hadoopクラスターをデプロイします...[OK]  
バッチプログラムを実行中です...[OK]  
Hadoopクラスターを削除します...[OK]  
[kaneko@console ~]$
```

[17:41] 0 console 1 rack

hp Network Topology - Cloud Ac x

https://horizon.hpcloud.com/project/cto-cloud-native-2014/network-topology/

Network Topology

cto-cloud-native-2014 Sign Out Support

Small Normal Launch Instance Create Network Create Router

Ext-Net

81958943-brab-4500-876d-b0239781555d 10.0.1.0/24

ctc-cloud-native-2014-network 10.0.0.0/24

2c36d701-ec83-446a-9471-ff6a2dd4ee85 10.0.3.0/24

Chat with HP Cloud

スクラッチ開発

データ入力

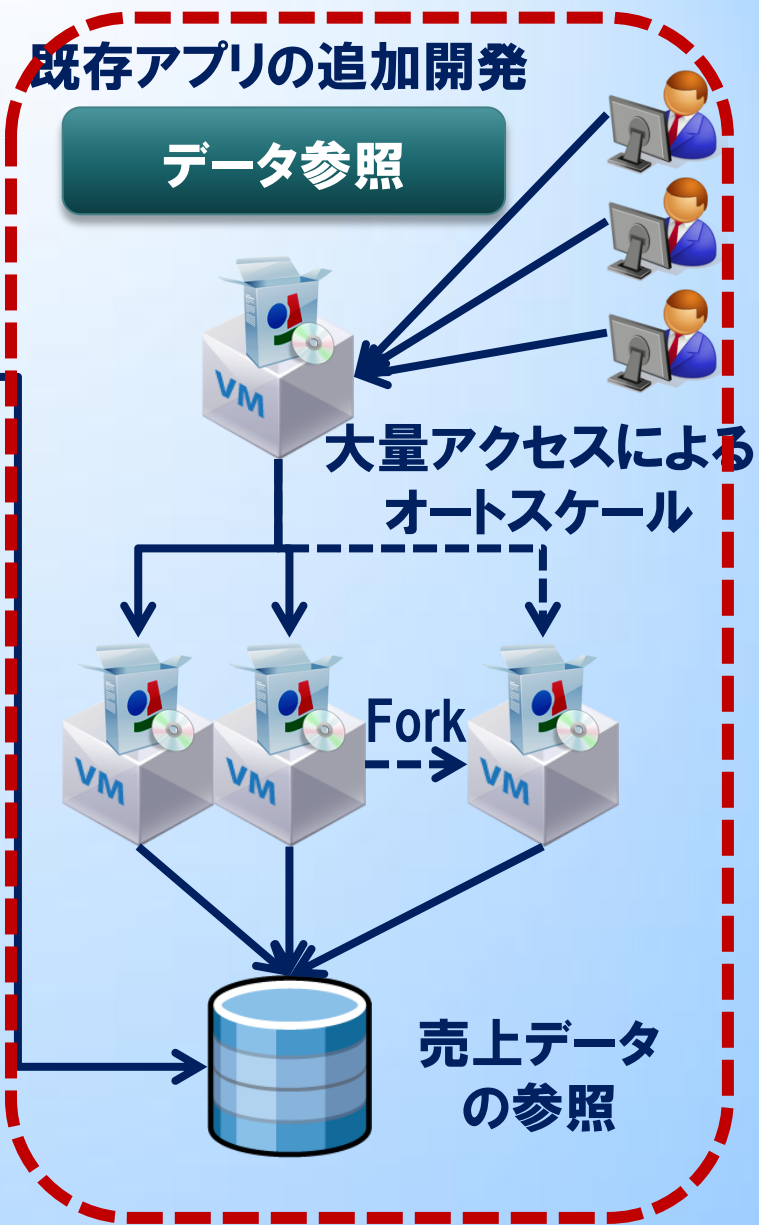
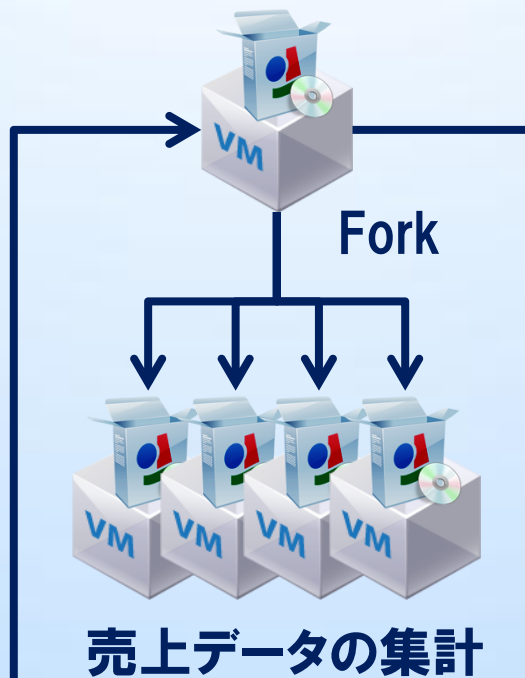
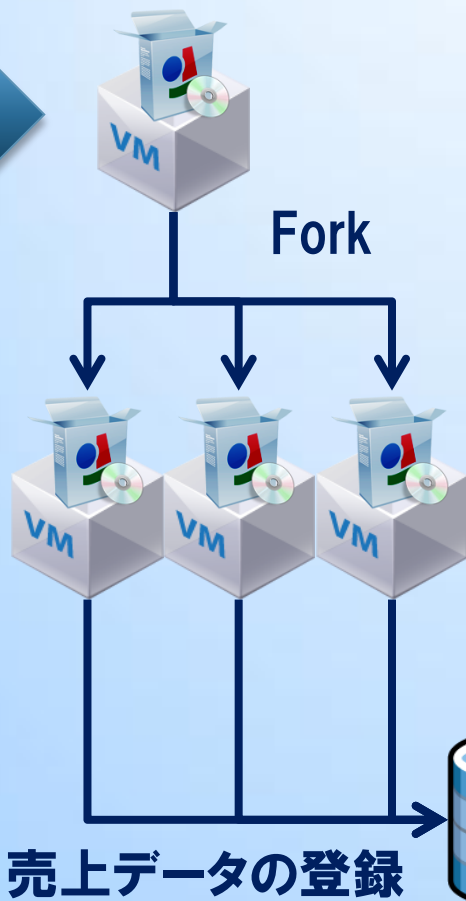
既存アプリとの組合せ

データ処理

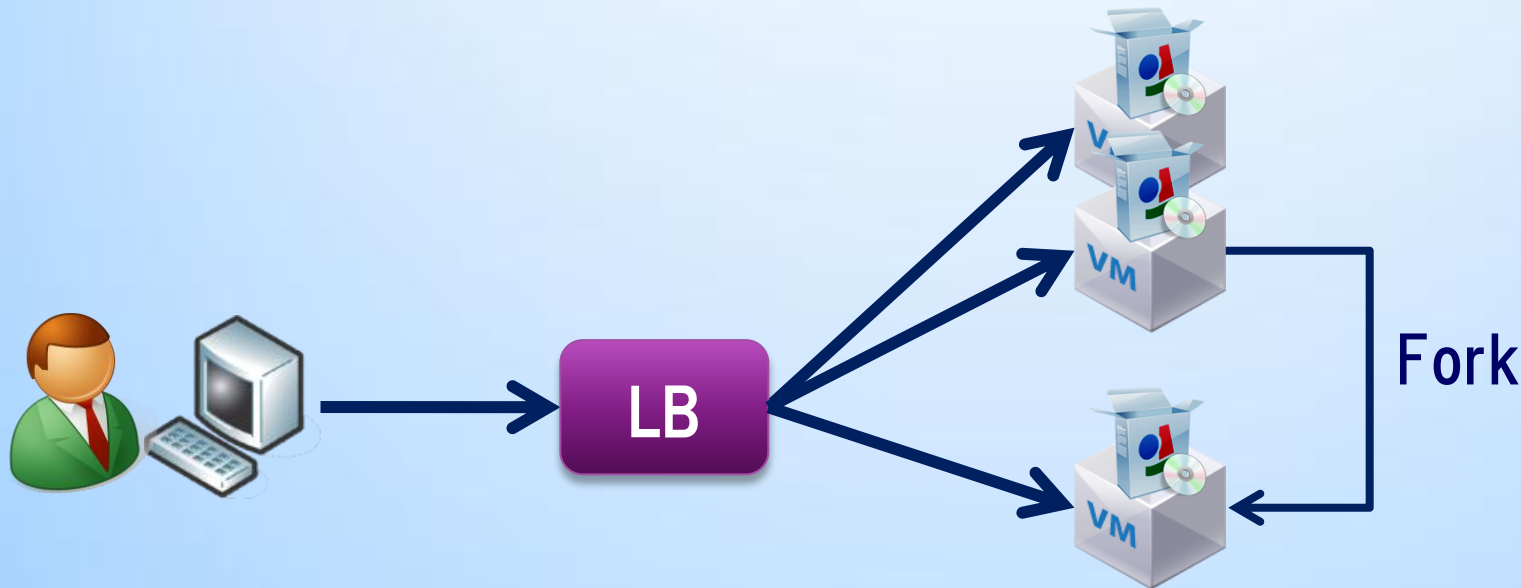
既存アプリの追加開発

データ参照

大量の
売上データ



- ✓ Webサーバプロセスは、自身の負荷が高いと判断するとForkする
- ✓ 子プロセス(Webサーバプロセス)は自身が起動するとLBに通知し、アクセスの振り分け先に追加させる
- ✓ 子プロセスは負荷が低くなったと判断すると、消滅する



RACK

クラウド(OpenStack)

デモ3(実際には動画を使用しました)

Cloud Native Application Sho x Cloud Native Application Sho x

10.0.1.4:1111/index.html

WEBアプリ

CPU: 50

SESSION: 50

送信

```
kaneko@console:~
インポート中... [OK]
+-----+-----+-----+-----+
| SID | CATEGORY_CODE | AMOUNT | SELLING_PRICE |
+-----+-----+-----+-----+
| 1   | 1300          | 7920   | 1053360        |
| 2   | 1401          | 8660   | 848680         |
| 3   | 1600          | 18604  | 3788520        |
+-----+-----+-----+-----+
[kaneko@console ~]$
[17:42] 0 console 1 rack
```

hp Network Topology - Cloud Ar x

https://horizon.hpcloud

Network Topology

Small Normal Launch Insta Network Create Router

初期状態ではLBプロセス、Webサーバプロセス、DBプロセスが起動しています。

Chat with HP Cloud

デモ3(実際には動画を使用しました)

稼働するWebサーバに対して、外部から負荷を与えます。

The screenshot shows a web browser displaying a page titled "WEBアプリ" (Web Application) with a CPU usage of 92% and a session count of 98. A slider control is visible. Below the browser is a terminal window showing a command prompt and a table of data.

SID	CATEGORY_CODE	AMOUNT	SELLING_PRICE
1	1300	7920	1053360
2	1401	8660	848680
3	1600	18604	3788520

The screenshot shows a "Network Topology" management interface. It features a diagram with four vertical bars representing network components: "Ext-Net" (blue), "8f9c58943-brad-4500-876d-bb23781555d" (orange), "ctc-cloud-native-2014-network" (green), and "2c36d701-ec33-446a-9474-fffa2dd4ee65" (red). Each bar has several server icons connected to it. The interface includes buttons for "Launch Instance", "Create Network", and "Create Router".

デモ3(実際には動画を使用しました)

Cloud Native Application Showcase

RACK(Real Application Centric Kernel)を利用したクラウド ネイティブなアプリケーションのサンプルです。

カテゴリ別売上集計

カテゴリコード	合計数量	合計金額
1300	7920	1053360
1401	8660	848680
1600	18604	3788520

Copyright (c) 2013 ITOCHU Techno-Solutions Corporation.

```
kaneko@console:~
インポート中...[OK]
+-----+-----+-----+-----+
| SID | CATEGORY_CODE | AMOUNT | SELLING_PRICE |
+-----+-----+-----+-----+
| 1 | 1300 | 7920 | 1053360 |
| 2 | 1401 | 8660 | 848680 |
| 3 | 1600 | 18604 | 3788520 |
+-----+-----+-----+-----+
[kaneko@console ~]$
[17:43] 0 console 1 rack
```

Network Topology

cto-cloud-native-2014 Sign Out Support

Small Normal Launch Instance Create Network Create Router

Ext-Net

01956933-b1ab-4500-876d-b022976

cto-cloud-native-2014-network

2c36d701-eca3-446a-9471-f7a2dd1

Chat with HP Cloud

負荷に応じて、起動済みWebサーバ用プロセスが”Fork”され、追加のプロセスが起動してきます。

デモ3(実際には動画を使用しました)

次に負荷を低下させます。

The screenshot shows a web browser window displaying a 'WEBアプリ' (Web Application) interface. It features two sliders: 'CPU: 18' and 'SESSION: 18'. A blue button labeled '送信' (Send) is visible. Below the browser is a terminal window with the following content:

```
kaneko@console:~  
インポート中... [OK]  
+-----+-----+-----+-----+  
| SID | CATEGORY_CODE | AMOUNT | SELLING_PRICE |  
+-----+-----+-----+-----+  
| 1 | 1300 | 7920 | 1053360 |  
| 2 | 1401 | 8660 | 848680 |  
| 3 | 1600 | 18604 | 3788520 |  
+-----+-----+-----+-----+  
[kaneko@console ~]$  
[17:44] 0 console 1 rack
```

The screenshot shows a 'Network Topology' management console. It displays a network diagram with four vertical bars representing network segments: 'Ext-Net' (blue), '89569433-br-ab-4500-876d-b023978' (orange), 'ctc-cloud-native-2014-network' (green), and '2c36d701-eca3-4afe-9471-ff0a2dd' (red). A switch icon is connected to the orange and green bars. On the right, a red bar is connected to a vertical stack of server icons. The interface includes buttons for 'Launch Instance', 'Create Network', and 'Create Router', along with a 'Chat with HP Cloud' button at the bottom right.

デモ3(実際には動画を使用しました)

余剰プロセスは自身を”kill”し、プロセスを終了させます。

Cloud Native Application Showcase

RACK(Real Application Centric Kernel)を利用したクラウド ネイティブなアプリケーションのサンプルです。

カテゴリ別売上集計

カテゴリコード	合計数量	合計金額
1300	7920	1053360
1401	8660	848680
1600	18604	3788520

Copyright (c) 2013 ITOCHU Techno-Solutions Corporation.

```
kaneko@console:~
インポート中...[OK]
+-----+-----+-----+-----+
| SID | CATEGORY_CODE | AMOUNT | SELLING_PRICE |
+-----+-----+-----+-----+
| 1 | 1300 | 7920 | 1053360 |
| 2 | 1401 | 8660 | 848680 |
| 3 | 1600 | 18604 | 3788520 |
+-----+-----+-----+-----+
[kaneko@console ~]$
[17:45] 0 console 1 rack
```

Network Topology

cto-cloud-native-20

Sign Out Support

Small Normal

Launch Instance Create Network Create Router

Ext-Net

cto-cloud-native-2014-network

cto-cloud-native-2014-network

100.1.0/24

10.0.0/24

10.0.3.0/24

Chat with HP Cloud

✓ 応用範囲が広い

➡ スクラッチ開発、既存のアプリケーションと組み合わせる、既存のアプリケーションに機能追加する

RACK自体は原始的な機能しか持たないため、**応用範囲に制限がない**

✓ より**厳密なスケール条件**が設定可能

➡ CPU使用率、メモリ消費量などの負荷情報ではなく、データ量、セッション数など**アプリケーション内部しか知りえない情報**をスケール条件に設定できる

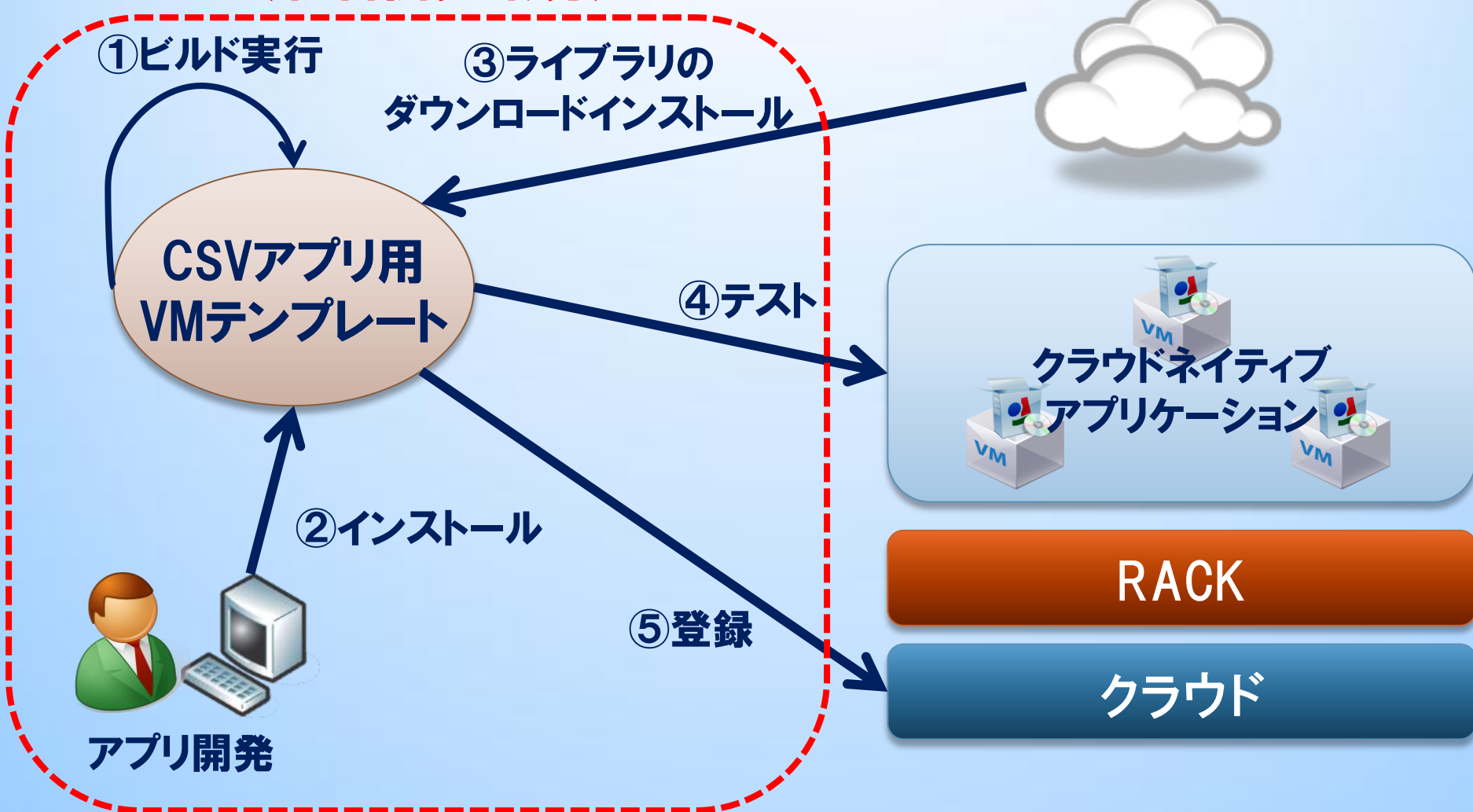
- ▶ 「クラウドネイティブ」アプリケーション
- ▶ Real Application Centric Kernel(R・A・C・K)
- ▶ ケーススタディ powered by RACK
- ▶ RACKの今後の展望



機能拡張(予定)

VMテンプレートのビルド・テストをしてくれるツール

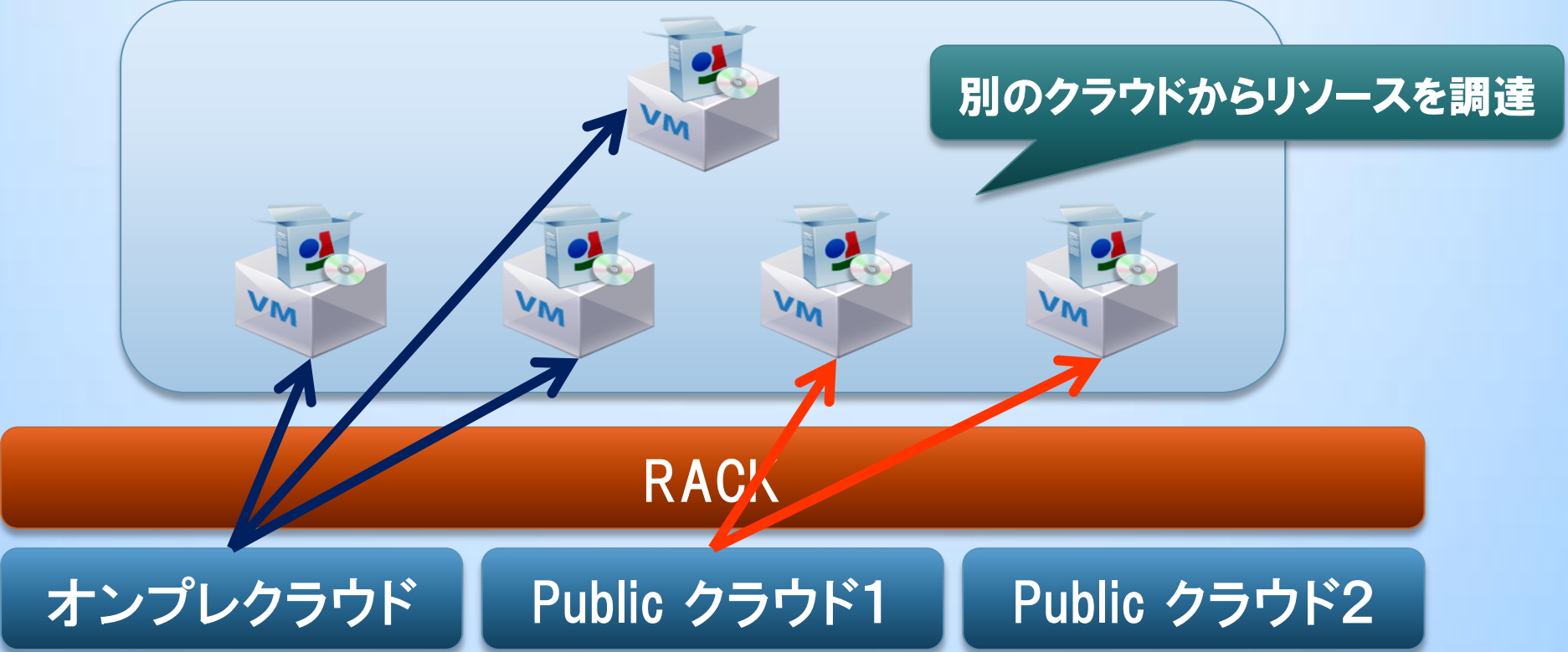
IDE(統合開発環境)



マルチクラウド化

オンプレクラウドで足りなくなったリソースを別のクラウドから調達する

クラウドネイティブアプリケーション



これ以上VMが作れない！

まとめ

- ▶ 「クラウドネイティブ」アプリケーション
- ▶ Real Application Centric Kernel(R・A・C・K)
- ▶ ケーススタディ powered by RACK
- ▶ RACKの今後の展望



「クラウドネイティブ」アプリケーション

システム要件が**未来予測**、**価値創造**に変化



必要なリソースを自動で調達して動作する

クラウドネイティブアプリケーション

という新しいアーキテクチャが今後必要になる



Real Application Centric Kernel(R・A・C・K)

クラウドを**プログラムから扱いやすくする仕組みが必要**



CTCが**RACK**を開発

RACKは**クラウドをプログラムから扱いやすくする**

RACKを使うと**実装をシンプルにできる**



ケーススタディ powered by RACK

RACKは応用範囲が広い
より厳密なスケール条件が設定可能



RACKの今後の展望

2014年3月までに**OSSとして公開**予定

実用化に向けた**機能追加・実証実験**

マルチクラウド対応などさらなる利便性向上

ご清聴ありがとうございました！！

CTC

▼ *Challenging Tomorrow's Changes*