

オープン系インフラは仮想化からSoftware Definedへ ～オープンソース・オーケストレーションツール CloudConductorのご紹介～

2015年5月29日

TIS株式会社
戦略技術センター
油谷実紀

自己紹介

油谷実紀(ゆたにみき)

TIS株式会社 戦略技術センター

Twitter: @blackaplysia

Facebook: 油谷実紀

- 新規技術の企画、検証、導入
 - ✓ Software-Defined Infrastructure (SDI)
 - ✓ Artificial Intelligence (人工知能)
- OSSデザイン指向クラウドオーケストレータ
CloudConductorのプロダクトマネージャー



Agenda

- ▶ 1. クラウドを取り巻く現状の振り返り
- 2. CloudConductorの概要
- 3. CloudConductorの今後の展開
- 4. お知らせ



クラウド市場の拡大

- 2013年度のクラウド市場は6,257億円であり、2018年度には2.9倍の1兆8,000億円まで拡大
- そのうちプライベートクラウドの比率は2013年度で70.1%を占めるが、2018年度には73.0%と緩やかにシェアを高める

利用許諾未取得のため図表を削除

MM総研, "国内クラウドサービス需要動向(2014年版)",
2014-11, <http://www.m2ri.jp/newsreleases/main.php?id=010120141104500>

パブリッククラウドの市場推移

- AWSの一人勝ちだが、Microsoftが急速に追いついてきている
- 一方、厳しい戦いを強いられているクラウドも多い

利用許諾未取得のため図表を削除

出展 : Gartner “Magic Quadrant for Cloud Infrastructure as a Service” 2012年度～2015年度を加工

LEADERS:

市場ニーズに対応する成熟した製品をリリースしており、市場が進化した場合でもリーダーの座を維持できるビジョンを明示している

VISIONARIES:

市場の進化に関するビジョンを持っているが、ビジョンを実現する能力が実証されていない

CHALLENGERS:

実行能力に優れているが、新たな顧客に最新かつ強力な価値を提案する戦略を欠く

NICHE PLAYERS:

特定の市場セグメントで成功を収めているか、またはイノベーションを実現する能力や競合他社を上回るために必要な能力が限られている

プライベートクラウドの市場推移

- 実際に利用しているプライベートクラウドはVMWare製品が多いが、検証中・計画中も合わせるとOpenStackも比肩する
- エンジニアが注目するOSSクラウド基盤ツールはOpenStackが支配的で、CloudStackやEucalyptusの将来は厳しいと言わざるを得ない

利用許諾未取得のため図表を削除

出展：RightScale “Cloud Computing Trends” 2014年と2015年を加工

利用許諾未取得のため図表を削除

クラウドサービスの継続性への期待

- クラウドサービス導入時には、クラウドサービスの継続性が保障されていることが重視されている

利用許諾未取得のため図表を削除

出展 : NRI Secure Technologies, “企業における情報セキュリティ実態調査2011”を加工

クラウドサービスは終了するという現実

- クラウドサービスは永遠に継続されるわけではなく、クラウド事業者の戦略変更のために利用者が不利益を被ることがある
 - クラウドストレージサービスを展開していたNirvanix社が**倒産**し、2013年9月に月末で**サービスを終了**することを告知
 - OpenStackをベースにしたプライベートクラウド構築用アプライアンスを販売していたNebula社は、2015年4月、**会社を廃業**
 - ✓ OpenStack創始者の1人であるChris Kemp氏が2011年に設立
 - LINEは2013年6月、オンラインストレージサービスやオンラインアルバムサービス等を終了し、**収益性の高いサービス開発に集中**
 - Googleは2013年7月、RSSリーダーのGoogle Readerを**終了**

クラウドはあたりまえのインフラになった一方、普及期に入った故の課題もまた明らかになってきた
しかし、もはや後戻りはできない

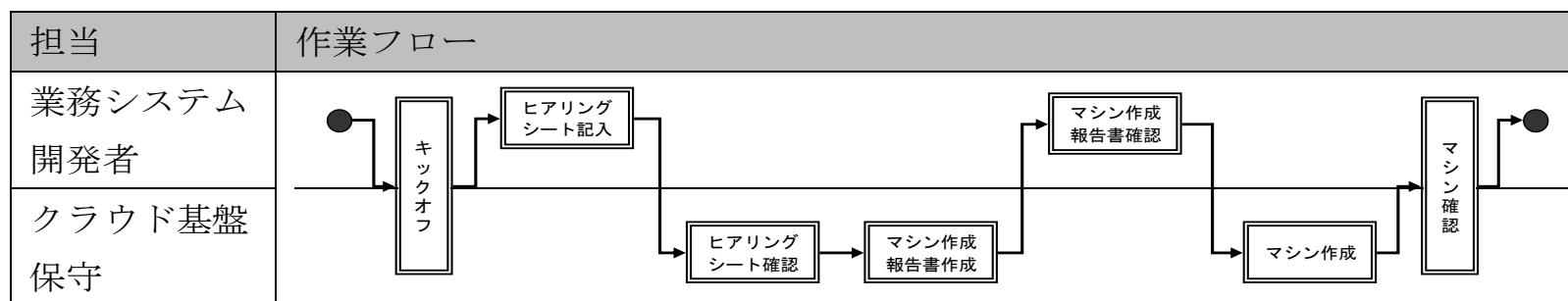
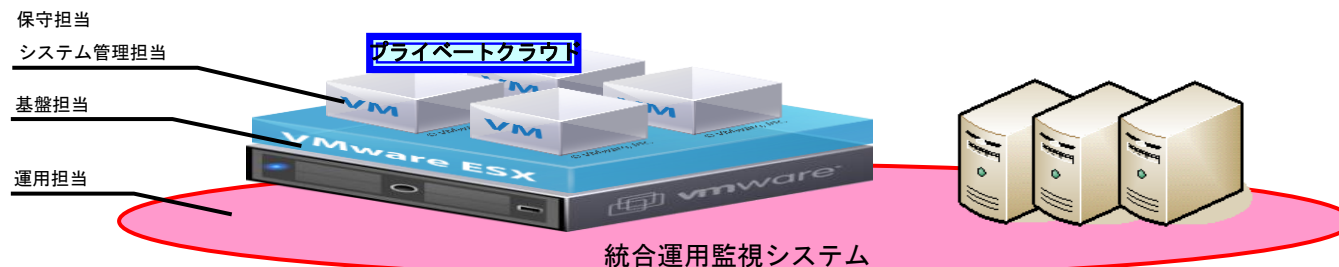
Agenda

1. クラウドを取り巻く現状の振り返り
- ▶ 2. CloudConductorの概要
3. CloudConductorの今後の展開
4. お知らせ



CloudConductorの開発動機=SDIのメリット享受

● SIにおけるクラウド活用の現実



→ 1~2週間 (3~5人日)

→ 1~2週間 (3~5人日)

→ 1週間 (2~3人日)

→ 1週間 (2~3人日)

構築準備 4.5~7週間 (11~19人日)

構築作業 0.2~0.5週間 (1~2人日)

構築確認 1週間 (1~3人日)

→ 0.5~1週間 (1~3人日)

0.2~0.5週間 (1~2人日)

1週間 (1~3人日)

出展：TIS, “運用側面からのCloudConductor調査”

2014-03, <http://download.cloudconductor.org/whitepaper/運用側面からのCloudConductor評価.pdf>

CloudConductorの開発動機=SDIのメリット享受

- 仮想化によって解決できるシステムインテグレーションの課題はごく一部に限定される



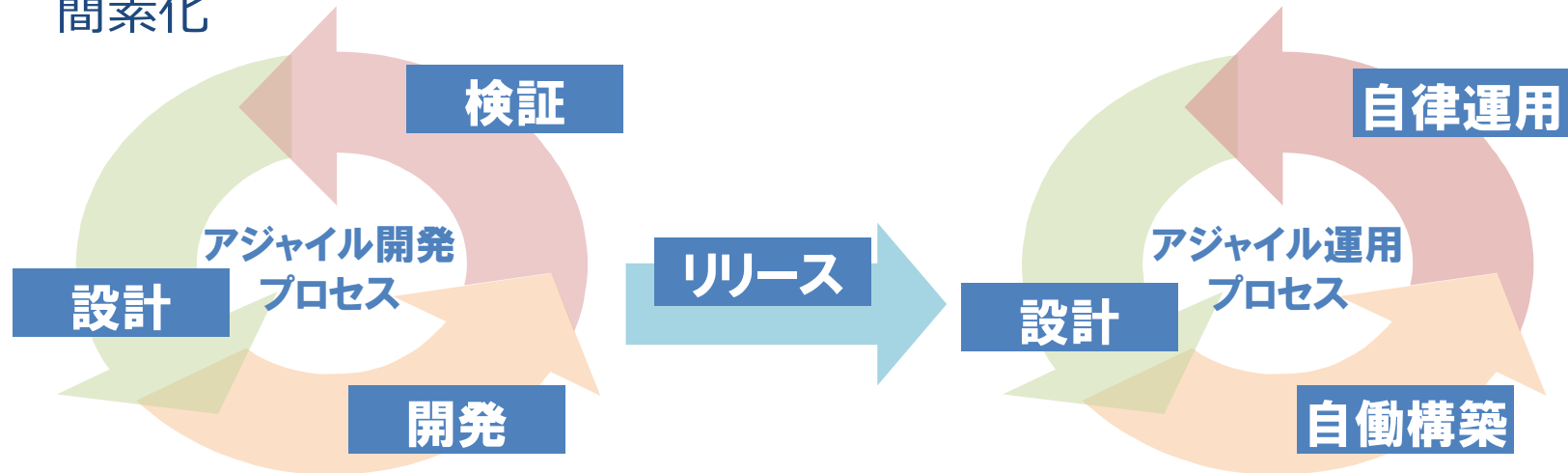
- 「必要なリソースを必要に応じてAPIを通じて取得できる」というクラウドの特性(**Software-Defined Infrastructure = SDI Architecture**)を活用し、

システムインテグレーションのプロセス自体をクラウドロックインされない形で

クラウドに適合させて、初めて**SIにおいてもクラウド化**の効果を得られる

アジャイルなインフラ運用プロセス

- システムの設計ノウハウを抽象化して標準化し、パターンとして実装することで、実績のある構成を再利用し、運用設計・構築プロセスを簡素化



パターン化された設計、自働化された構築、標準化された運用

(Hypervisor, Orchestrator = CloudConductor, Platform-as-a-Service, ...)

標準運用を含むインフラパターン

(Build/Bootstrap/Test Scripts, Configuration Parameters, ...)

ミドルウェア標準

(Web/APサーバ、RDBMS、冗長化ソフトウェア、認証認可システム、統合ログ管理システム、統合監視システム等)

インフラ方式標準

(冗長化方式、負荷分散方式、DR/バックアップ方式、ネットワーク設計、統合ログ管理方式、統合監視方式等)

運用標準

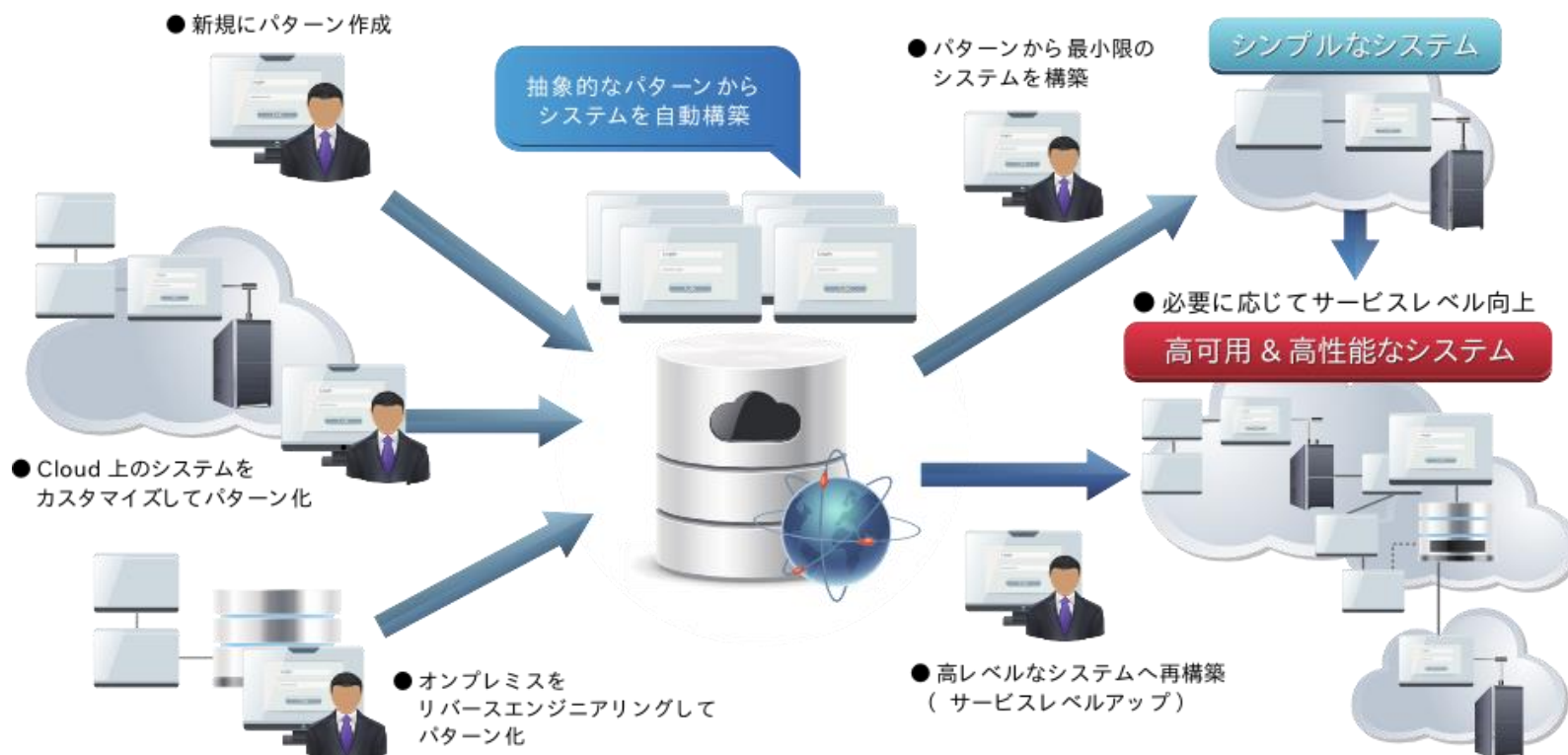
(QMS/ITIL、SLAレベル、業務継続計画 (BCP)、クラウドポートフォリオ)

アジャイルなインフラ運用を実現するためのツール

- デザイン指向クラウドオーケストレータ CloudConductor

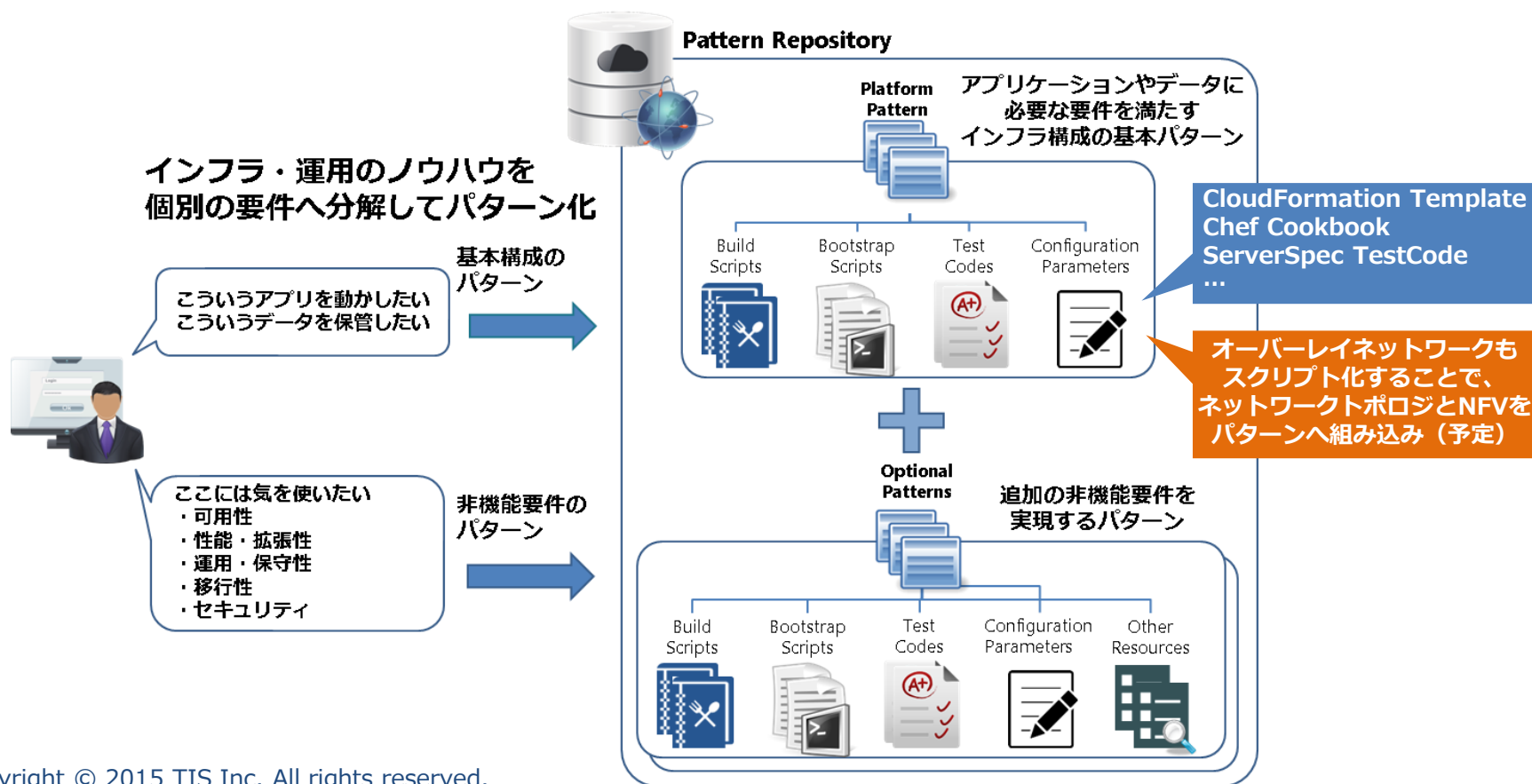
- **Software-Defined Infrastructure (SDI)時代のインフラ構築**

= インフラ・運用のノウハウを含めた **パターン** を中心に、
いつでも誰でも **どのクラウド** にでも、その時点で最適な
非機能要件を持ったシステムを簡単に構築することができる



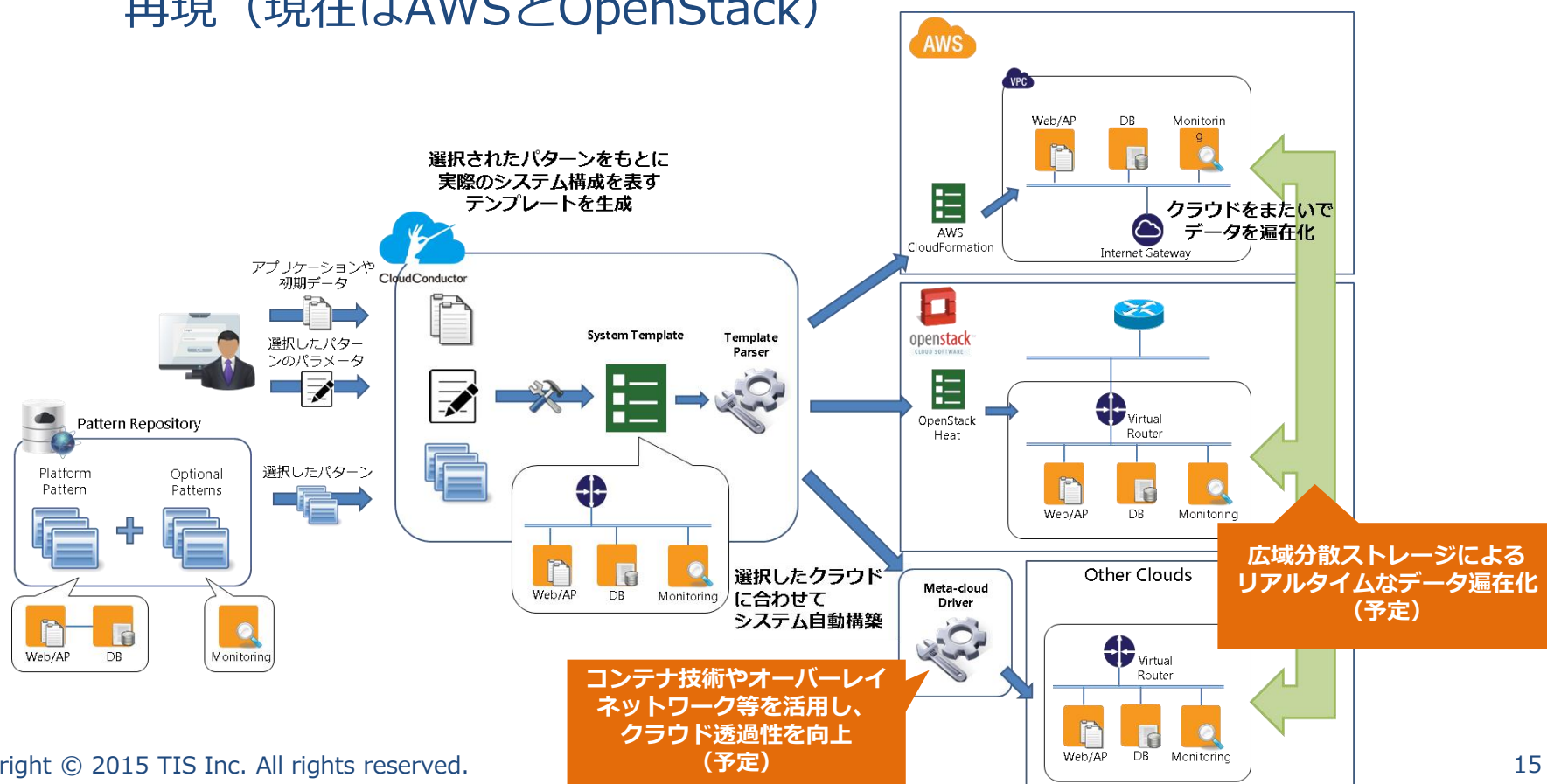
CloudConductorの特徴①

- Infrastructure Design Patterns as Code
 - インフラ・運用のノウハウを依存関係を整理してパターン化
 - パターンを機械可読な形式で集積し、集合知化



CloudConductorの特徴②

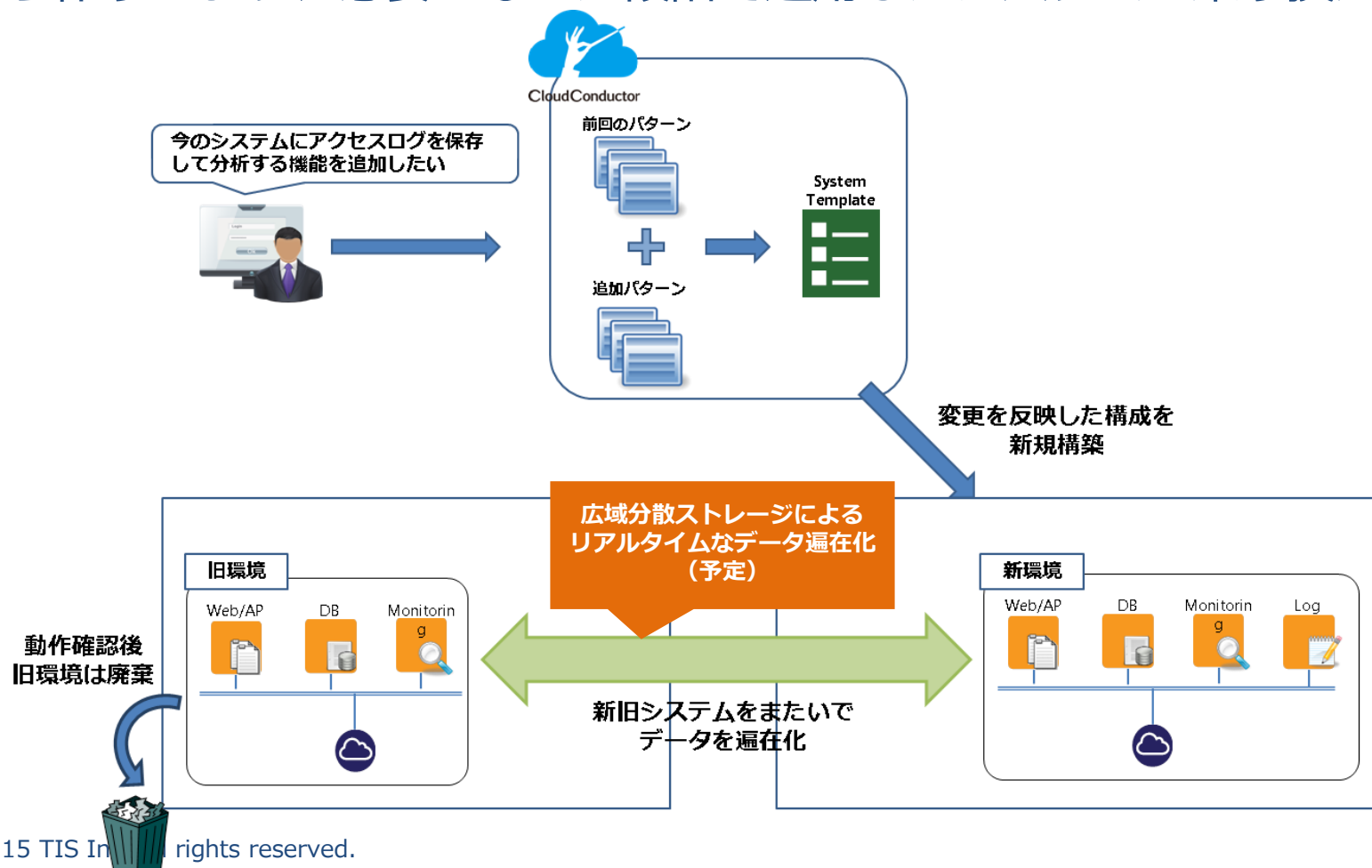
- Everyone, EveryTime & EveryCloud
 - 必要なパターンを組み合わせることで誰でも最適なインフラ設計を獲得
 - クラウドを跨りデータを遍在化させ、どのクラウドでもシステムを再現（現在はAWSとOpenStack）



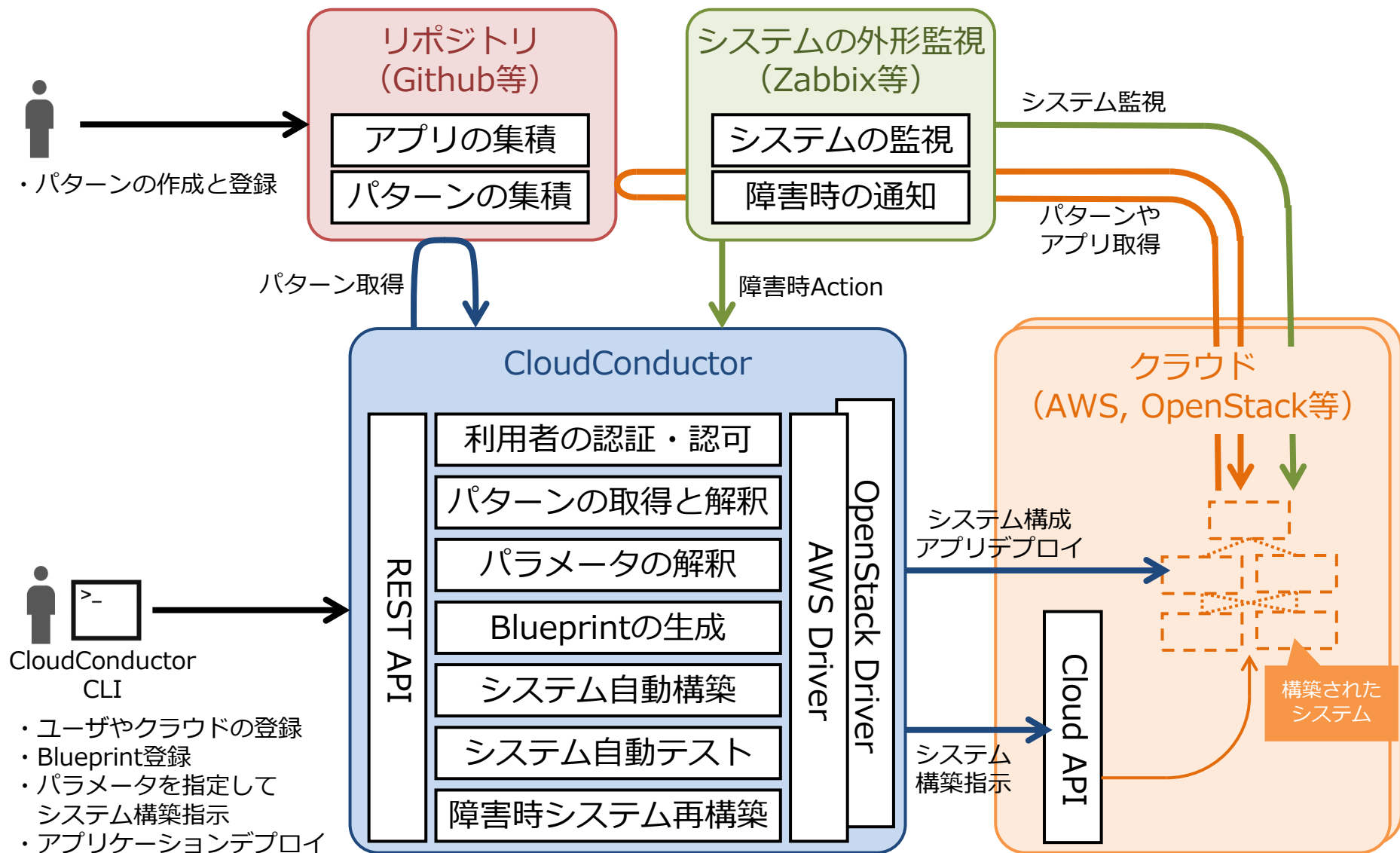
CloudConductorの特徴③

- OnDemand Service Level

- 「負荷分散」「災害対策」「ログ分析」等の非機能要件は最初から作りこまず、必要になった段階で適用したシステムへ乗り換え

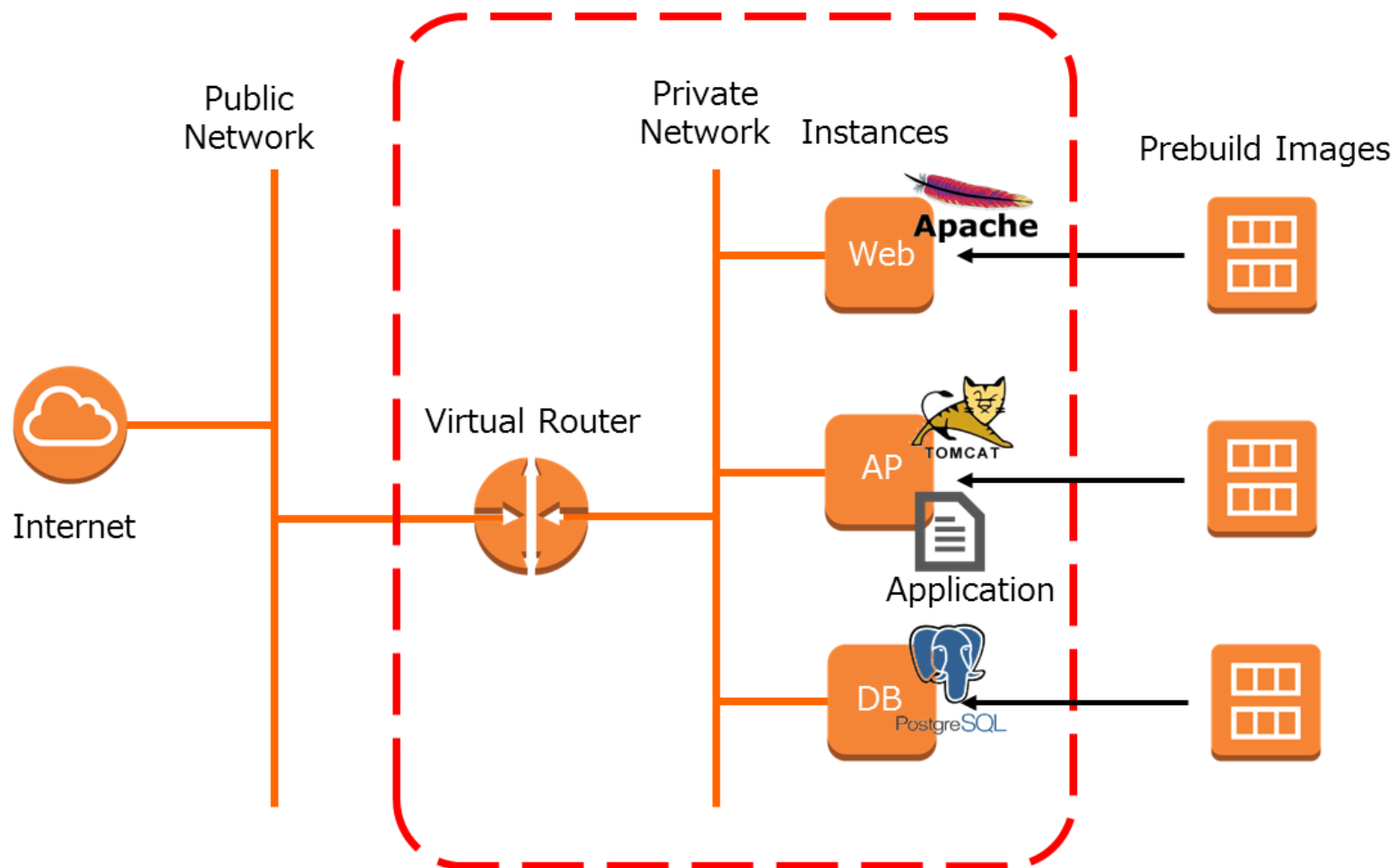


CloudConductor最新版の全体構成



CloudConductorのインフラデザインパターンとは

- たとえば以下のような構成の構築スクリプトをパターンとして定義
(この構成は実際に公式サイトにて公開済み)



CloudConductorによるシステム構築（1/3）

① OSやミドルウェアが導入された仮想マシンイメージの生成

The screenshot displays the OpenStack dashboard interface on the left and a terminal window on the right. The dashboard shows the 'Network Topology' section with a diagram of a public and private network. The terminal window shows the execution of CloudConductor commands to create a cloud and a blueprint.

OpenStack Dashboard:

- Project: demo
- Network Topology: Compact, Standard
- Networks: public (10.255.506.0/24), private (192.168.0.0/24)

Terminal Output:

```
tominaga@conductor-dev:~/v1.0.2
[tominaga@conductor-dev v1.0.2]$ cc-cli cloud show openstack

+-----+-----+
| property | value |
+-----+-----+
| id        | 1     |
| project_id | 1     |
| name      | openstack |
| description |      |
| type      | openstack |
| entry_point | http://10.255.197.110:5000/ |
| key       | demo  |
| secret    | ***** |
| tenant_name | demo  |
| created_at | 2015-05-22T01:10:06.999Z |
| updated_at | 2015-05-22T01:10:06.999Z |
+-----+-----+

[tominaga@conductor-dev v1.0.2]$ cc-cli blueprint create --project sample_project --name tomcat --description "Apache, Tomcat, PostgreSQL" --patterns-json '{"url": "https://github.com/cloudconductor-patterns/tomcat_pattern.git", "revision": "master"}'
Create completed successfully.

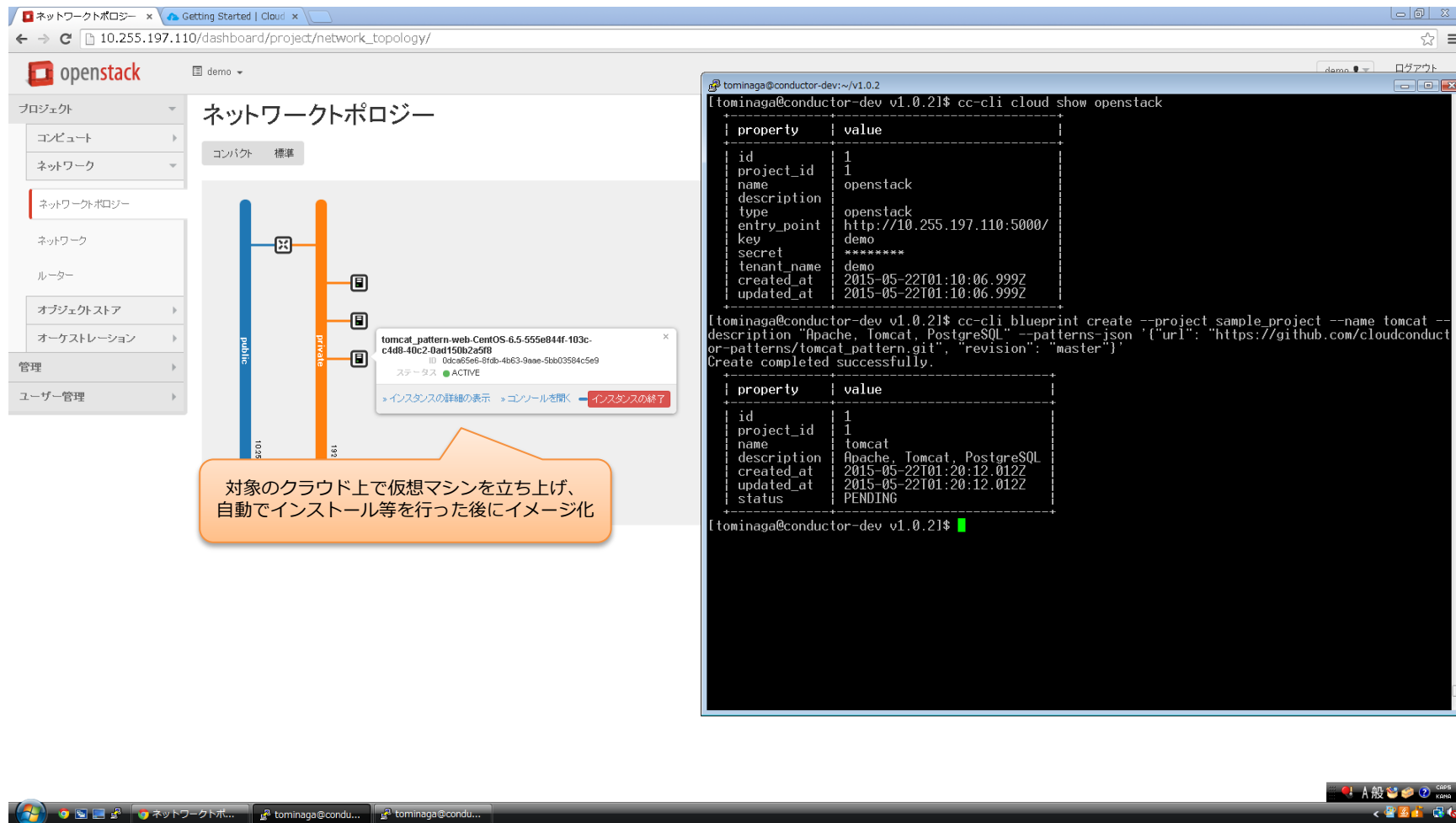
+-----+-----+
| property | value |
+-----+-----+
| id        | 1     |
| project_id | 1     |
| name      | tomcat |
| description | Apache, Tomcat, PostgreSQL |
| created_at | 2015-05-22T01:20:12.012Z |
| updated_at | 2015-05-22T01:20:12.012Z |
| status    | PENDING |
+-----+-----+

[tominaga@conductor-dev v1.0.2]$
```

※実行するコマンドの詳細は、CloudConductor公式サイトのGetting Startedを参照
<http://cloudconductor.org/documents/getting-started>

CloudConductorによるシステム構築（1/3）

① OSやミドルウェアが導入された仮想マシンイメージの生成



The screenshot shows the OpenStack dashboard on the left and a terminal window on the right. The dashboard displays the 'Network Topology' section with a diagram of public and private networks. A callout box indicates that a virtual machine is being launched and imaged on the cloud. The terminal window shows the execution of the 'cc-cli cloud show openstack' command, which returns details about the OpenStack cloud configuration. Below this, the 'cc-cli blueprint create' command is shown, which creates a new blueprint named 'tomcat' for a sample project.

OpenStack Dashboard: ネットワークポロジ

コンパクト 標準

対象のクラウド上で仮想マシンを立ち上げ、自動でインストール等を行った後にイメージ化

Terminal Output:

```
tominaga@conductor-dev:~/v1.0.2
[tominaga@conductor-dev v1.0.2]$ cc-cli cloud show openstack
```

property	value
id	1
project_id	1
name	openstack
description	
type	openstack
entry_point	http://10.255.197.110:5000/
key	demo
secret	*****
tenant_name	demo
created_at	2015-05-22T01:10:06.999Z
updated_at	2015-05-22T01:10:06.999Z

```
[tominaga@conductor-dev v1.0.2]$ cc-cli blueprint create --project sample_project --name tomcat --description "Apache, Tomcat, PostgreSQL" --patterns-json '{"url": "https://github.com/cloudconductor-patterns/tomcat_pattern.git", "revision": "master"}'
```

Create completed successfully.

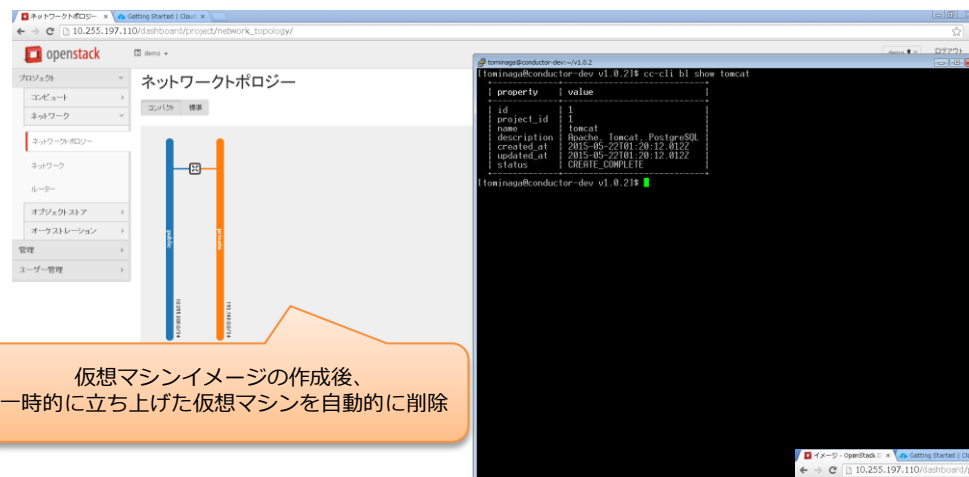
property	value
id	1
project_id	1
name	tomcat
description	Apache, Tomcat, PostgreSQL
created_at	2015-05-22T01:20:12.012Z
updated_at	2015-05-22T01:20:12.012Z
status	PENDING

```
[tominaga@conductor-dev v1.0.2]$
```

※実行するコマンドの詳細は、CloudConductor公式サイトのGetting Startedを参照
<http://cloudconductor.org/documents/getting-started>

CloudConductorによるシステム構築 (1/3)

① OSやミドルウェアが導入された仮想マシンイメージの生成



仮想マシンイメージの作成後、
一時的に立ち上げた仮想マシンを自動的に削除



生成された仮想マシンイメージ

※実行するコマンドの詳細は、CloudConductor公式サイトのGetting Startedを参照
<http://cloudconductor.org/documents/getting-started>

CloudConductorによるシステム構築 (1/3)

② パターンからシステムを自動構築

パターンと可変のパラメータを設定してシステム構築を指示

The screenshot displays the OpenStack dashboard interface. On the left, a sidebar menu shows navigation options: プロジェクト (Project), コンピュート (Compute), ネットワーク (Network), ネットワークポロジ (Network Topology), ネットワーク (Network), ルーター (Router), オブジェクトストア (Object Storage), オーケストレーション (Orchestration), 管理 (Management), and ユーザー管理 (User Management). The main content area is titled 'ネットワークポロジ' (Network Topology) and shows a diagram with three vertical bars representing network segments: 'public' (blue), 'private' (orange), and 'tomcat_system-f-tomcat-pattern-vpc-1024zone' (green). Each segment has a small icon and an IP address range: '10.0.0.0/24', '192.168.0.0/24', and '10.0.10.0/24' respectively. To the right, a terminal window shows the execution of CloudConductor commands. The commands include 'cc-cli environment create --system tomcat_system --name tomcat_en', 'environment --blueprint tomcat --clouds openstack', and 'Input tomcat_pattern Parameters'. The output shows the creation of the environment and a table of properties.

property	value
id	6
system_id	1
blueprint_id	2
name	tomcat_environment
description	
status	PENDING
ip_address	
created_at	2015-05-22T02:28:04.903Z
updated_at	2015-05-22T02:28:04.903Z
application_status	NOT_DEPLOYED

※実行するコマンドの詳細は、CloudConductor公式サイトのGetting Startedを参照
<http://cloudconductor.org/documents/getting-started>

CloudConductorによるシステム構築 (1/3)

② パターンからシステムを自動構築

The screenshot displays the OpenStack dashboard interface on the left and a terminal window on the right. The dashboard shows the 'Network Topology' section with a diagram of a network setup. The terminal window shows the execution of the 'cc-cli env show tomcat_environment' command, which returns a table of properties for the 'tomcat_environment'.

OpenStack Dashboard: ネットワークポロジ

コンパクト 標準

tomcat system - f-tomcat pattern - VPE - bizmonzo

ネットワークや仮想マシンが自動的に作成され OSやミドルウェアの設定等も自動で行われる

Terminal Output:

```
tominaga@conductor-dev:~/v1.0.2$ cc-cli env show tomcat_environment
```

property	value
id	6
system_id	1
blueprint_id	2
name	tomcat_environment
description	
status	PROGRESS
ip_address	
created_at	2015-05-22T02:28:04.903Z
updated_at	2015-05-22T02:28:04.952Z
application_status	NOT_DEPLOYED

```
tominaga@conductor-dev v1.0.2$ cc-cli env show tomcat_environment
```

property	value
id	6
system_id	1
blueprint_id	2
name	tomcat_environment
description	
status	CREATE_COMPLETE
ip_address	10.255.206.169
created_at	2015-05-22T02:28:04.903Z
updated_at	2015-05-22T02:32:11.359Z
application_status	NOT_DEPLOYED

```
tominaga@conductor-dev v1.0.2$
```

※実行するコマンドの詳細は、CloudConductor公式サイトのGetting Startedを参照
<http://cloudconductor.org/documents/getting-started>

CloudConductorによるシステム構築 (3/3)

③ アプリケーションをデプロイ

The screenshot displays the OpenStack dashboard on the left and a terminal window on the right. The dashboard shows the 'Network Topology' section with a diagram of a network setup. The terminal window shows the execution of commands to list the 'tomcat_system' and the 'app list'.

OpenStack Dashboard: ネットワークポロジ

コンバート 標準

tomcat_system = tomcat pattern: VPS - M100mzone

public 10.255.506.0/24
private 193.168.0.0/24
100.10.0/24

Terminal Window:

```
tominaga@conductor-dev:~/v1.0.2$ cc-cli system show tomcat_system
```

property	value
id	1
project_id	1
primary_environment_id	6
name	tomcat_system
description	tomcat_system_description
domain	sample.demo.cloudconductor.jp
created_at	2015-05-22T01:46:40.621Z
updated_at	2015-05-22T02:32:11.551Z

```
tominaga@conductor-dev v1.0.2$ cc-cli app list
```

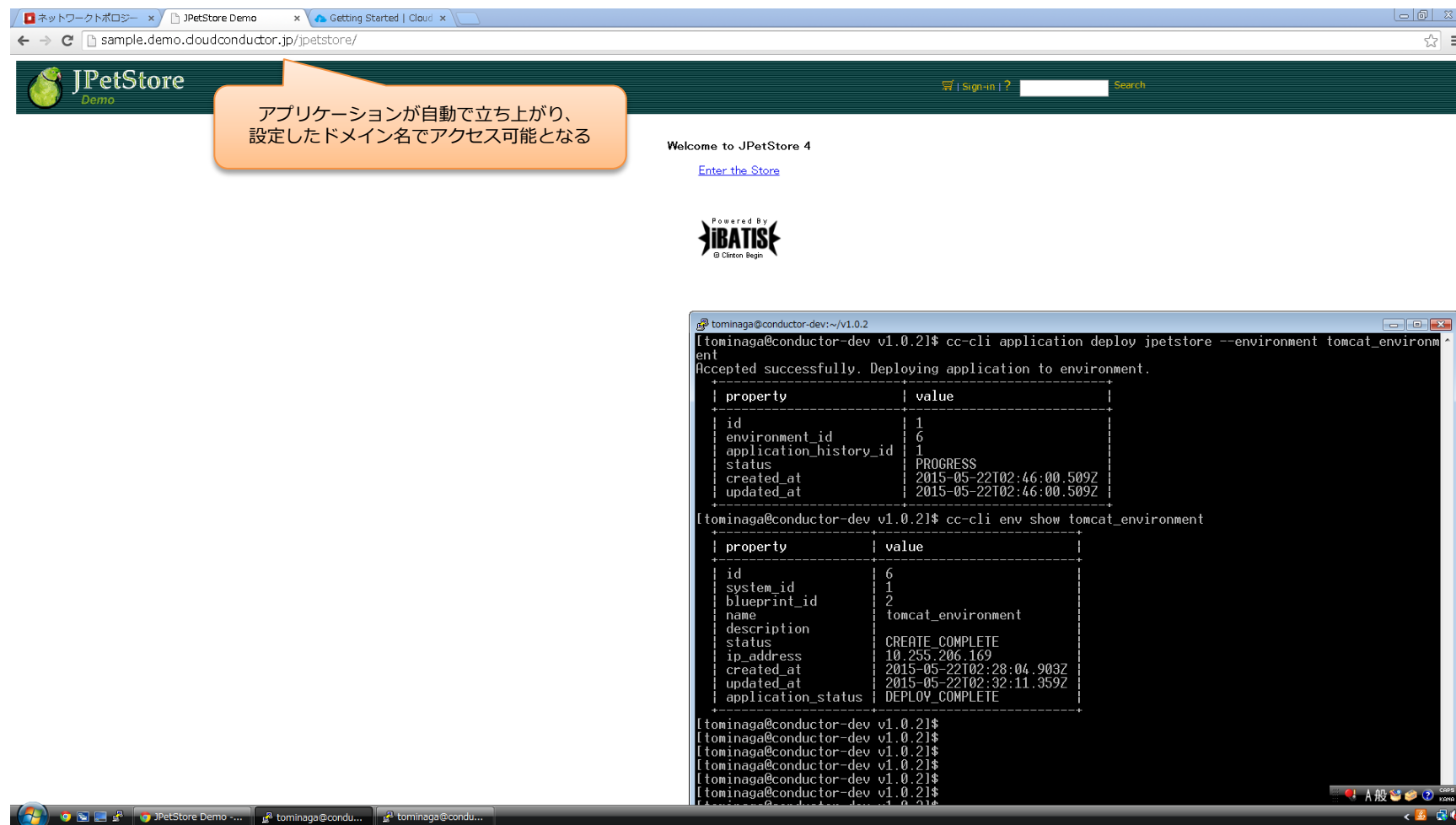
id	system_id	name	description	created_at	updated_at
1	1	jpetstore	jpetstore	2015-05-22T02:35:08.656Z	2015-05-22T02:35:08.656Z

構築したパターンに適合したアプリケーションをデプロイ

※実行するコマンドの詳細は、CloudConductor公式サイトのGetting Startedを参照
<http://cloudconductor.org/documents/getting-started>

CloudConductorによるシステム構築 (3/3)

③ アプリケーションをデプロイ



アプリケーションが自動で立ち上がり、
設定したドメイン名でアクセス可能となる

Welcome to JPetStore 4

[Enter the Store](#)

Powered By
IBATIS
© Clinton Begin

```
tominaga@conductor-dev:~/v1.0.2
[tominaga@conductor-dev v1.0.2]$ cc-cli application deploy jpetstore --environment tomcat_environment
Accepted successfully. Deploying application to environment.

+-----+-----+
| property | value |
+-----+-----+
| id        | 1     |
| environment_id | 6     |
| application_history_id | 1     |
| status    | PROGRESS |
| created_at | 2015-05-22T02:46:00.509Z |
| updated_at | 2015-05-22T02:46:00.509Z |
+-----+-----+

[tominaga@conductor-dev v1.0.2]$ cc-cli env show tomcat_environment

+-----+-----+
| property | value |
+-----+-----+
| id        | 6     |
| system_id | 1     |
| blueprint_id | 2     |
| name      | tomcat_environment |
| description |      |
| status    | CREATE_COMPLETE |
| ip_address | 10.255.206.169 |
| created_at | 2015-05-22T02:28:04.903Z |
| updated_at | 2015-05-22T02:32:11.359Z |
| application_status | DEPLOY_COMPLETE |
+-----+-----+

[tominaga@conductor-dev v1.0.2]$
[tominaga@conductor-dev v1.0.2]$
[tominaga@conductor-dev v1.0.2]$
[tominaga@conductor-dev v1.0.2]$
[tominaga@conductor-dev v1.0.2]$
[tominaga@conductor-dev v1.0.2]$
[tominaga@conductor-dev v1.0.2]$
```

※実行するコマンドの詳細は、CloudConductor公式サイトのGetting Startedを参照
<http://cloudconductor.org/documents/getting-started>

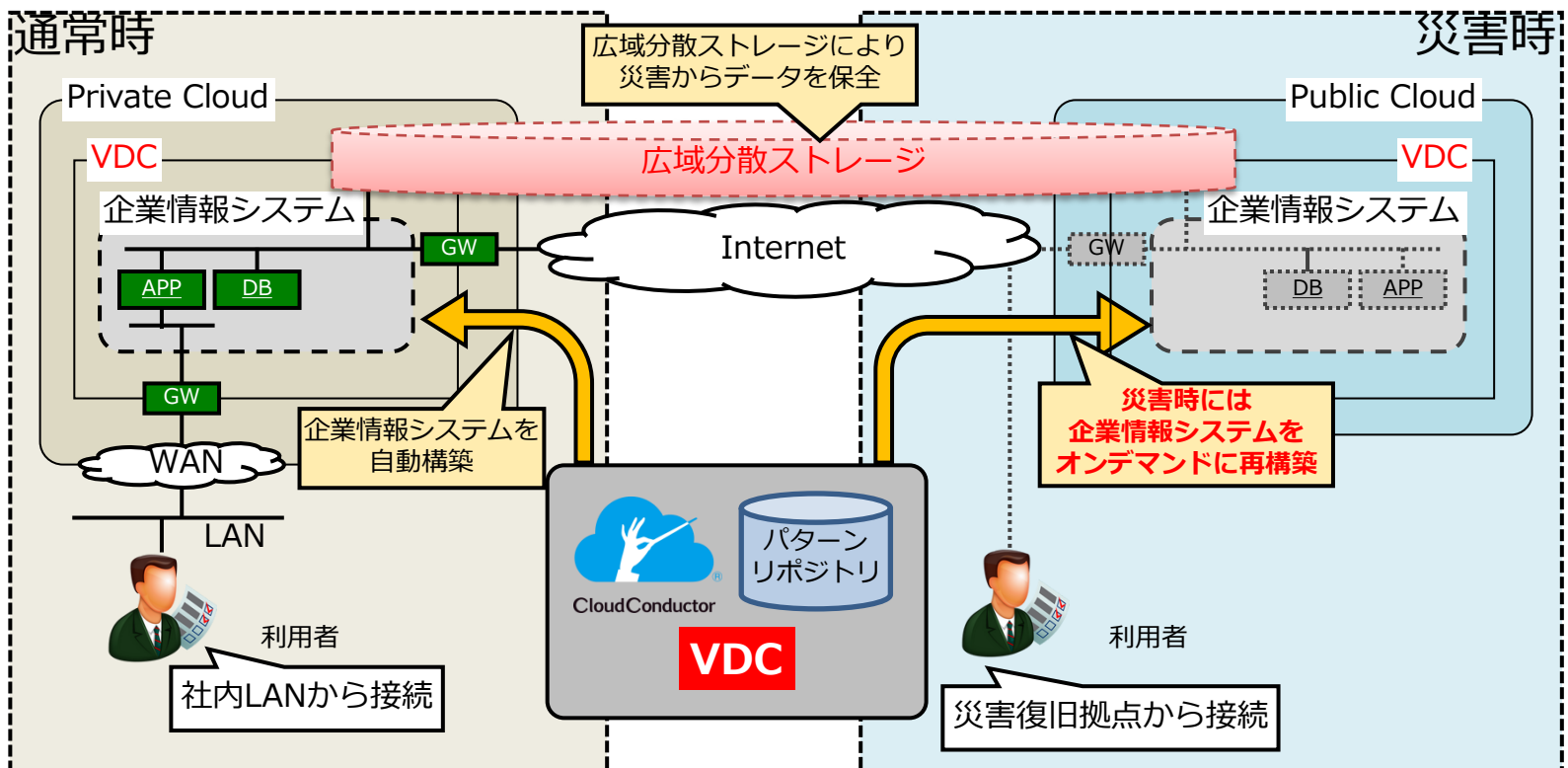
Agenda

1. クラウドを取り巻く現状の振り返り
2. CloudConductorの概要
- ▶ 3. CloudConductorの今後の展開
4. お知らせ



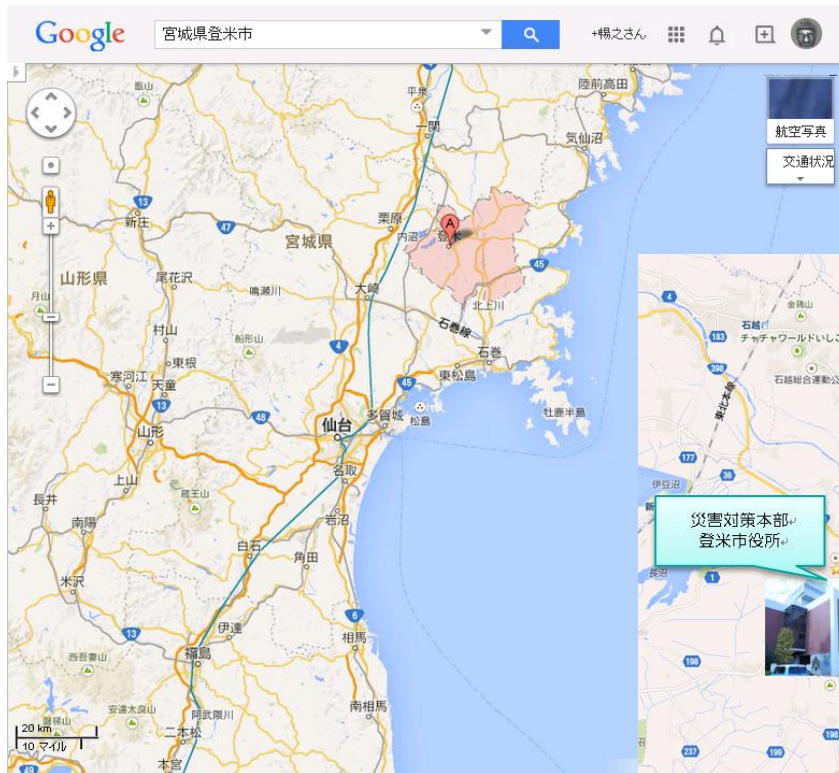
災害時のシステム復旧サービスへの応用

- 災害時に別クラウドへシステムを自動復元するシステム復旧サービス
 - 必要なときに初めて災対システムを立ち上げることで、災対サイトの維持管理コストが低減できる



災害時のシステム復旧サービスへの応用

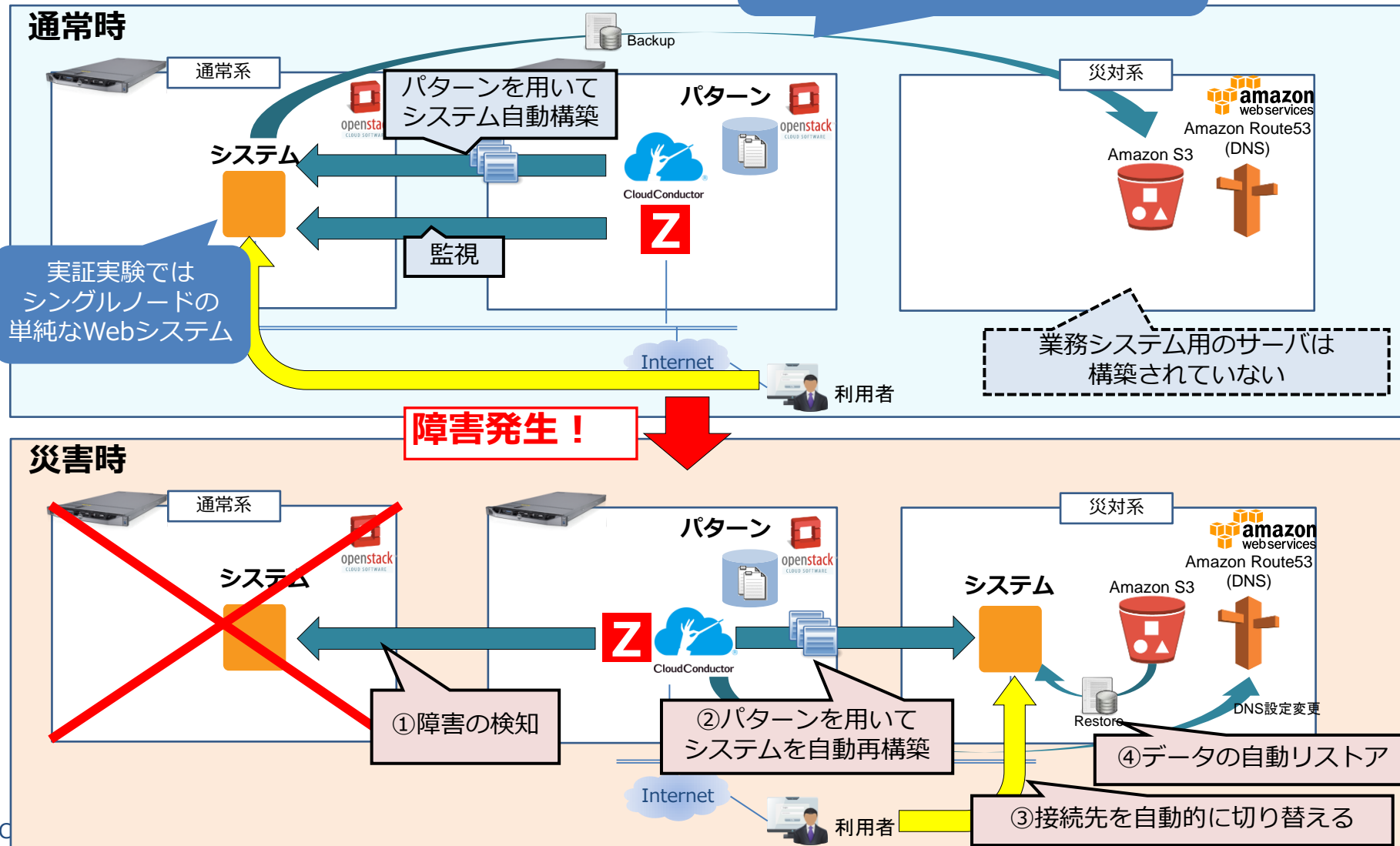
- 宮城県登米市・慶應大学・TISで災害対策の実証実験を実施
(2014/11/07)



災害時のシステム復旧サービスへの応用

● 実証実験の概要

実証実験ではVDCは未開発だったため
単純なバックアップ・リストア

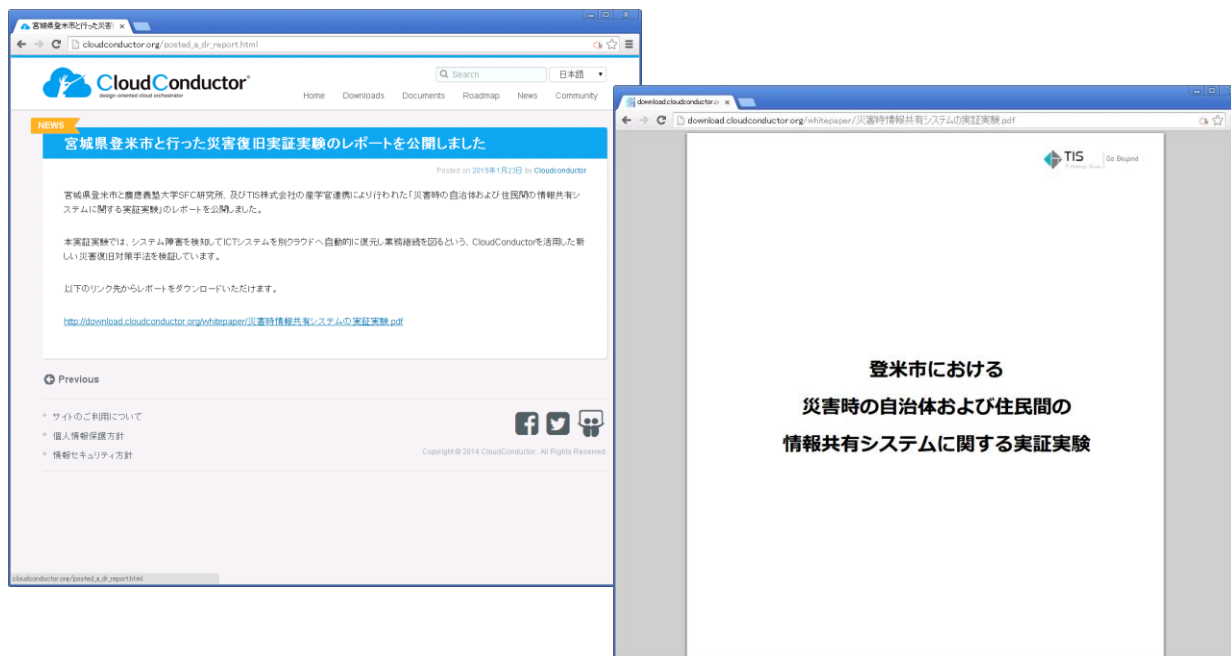


災害時のシステム復旧サービスへの応用

- システムの再構築に要した時間 = **6分53秒**

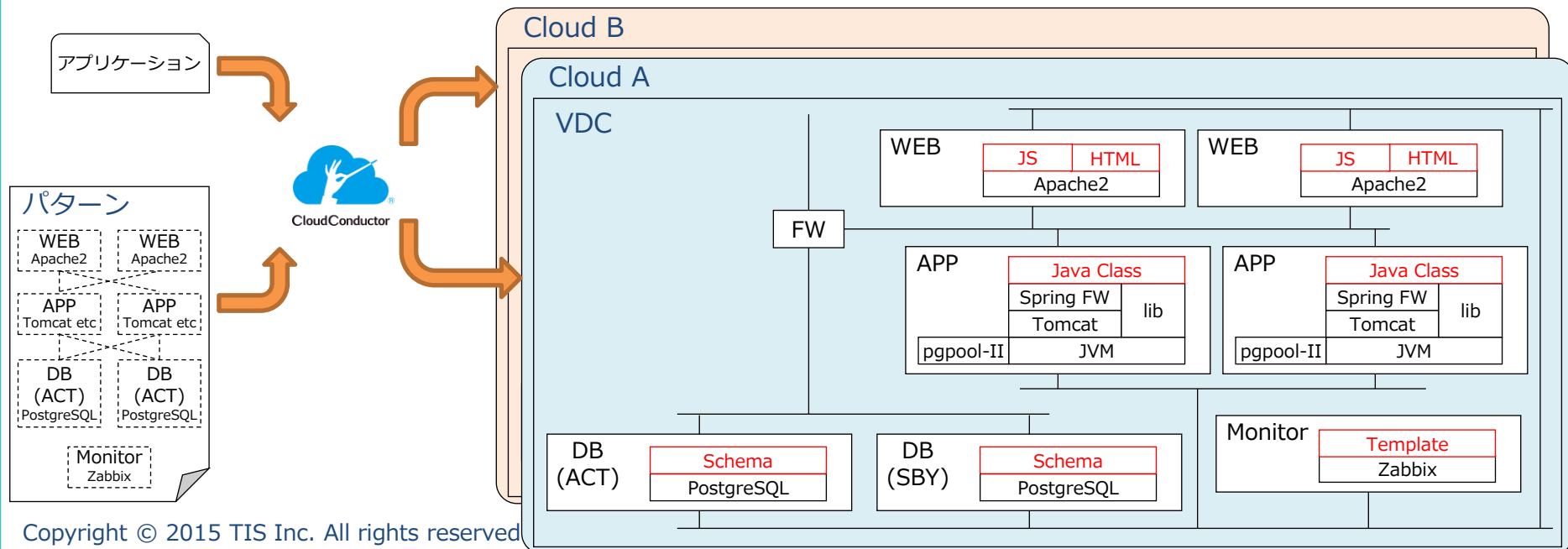
時刻	経過時間	イベント
7:05:50	00:00	災害発生（人為的にNICをダウン）
7:06:05	00:15	CloudConductorによる障害検知
7:06:21	00:31	CloudConductorによるシステム再構築開始
7:12:43	06:53	システム復旧

※実証実験の詳細は、CloudConductor公式サイトで公開しているレポートを参照
<http://cloudconductor.org/whitepaper/災害時情報共有システムの実証実験.pdf>



システムのボイラープレートへの応用

- システムの鋳型（ボイラープレート）によるサービスの個社展開
 - サーバやネットワーク、各種ミドルウェア、アプリケーションのフレームワークやライブラリ、データベース定義や初期データ、及び運用スクリプトまで全てをパッケージングしてパターン化
 - 「開発したアプリをデプロイするだけ」という視点はPaaSに似ているが、構築されたシステムは**従来通りのノウハウ**で運用できる

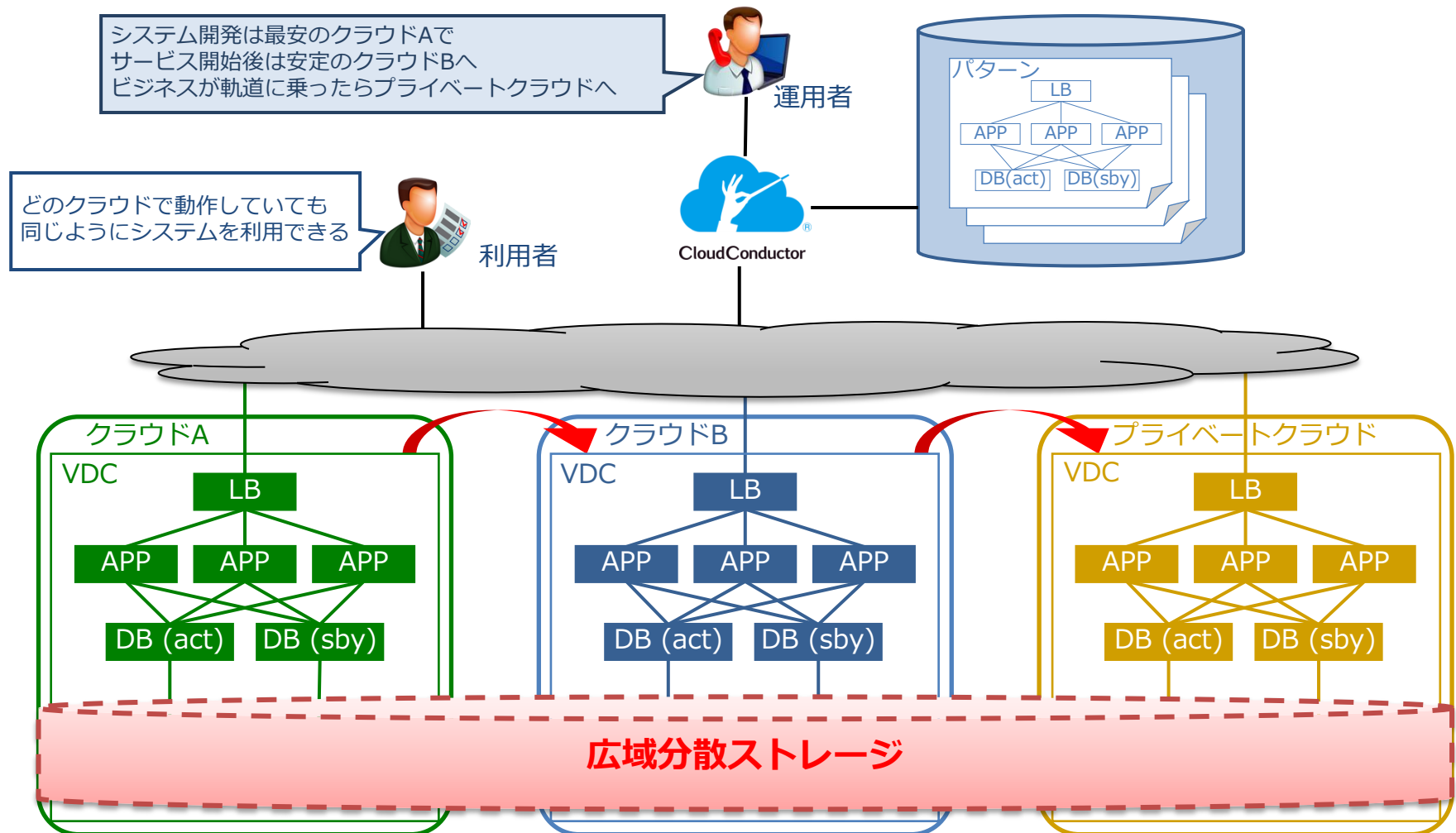


システムのボイラープレートへの応用

- システムの鋳型（ボイラープレート）によるサービスの個社展開
 - ERPのような、ある程度大規模で、かつパッケージのカスタマイズが主たる開発になるサービスを、短納期かつ高品質に個社展開する場合に最適
- あるサービスを動作させるためのインフラ構成やミドルウェア、弊社内製アプリケーションフレームワークやライブラリ等をCloudConductorで連携させるボイラープレートを企画開発中

ブローカーサービスへの応用

- システムのライフサイクルに従い、最適なクラウドへシステムを乗り換えさせることができるブローカーサービス



Agenda

1. クラウドを取り巻く現状の振り返り
2. CloudConductorの概要
3. CloudConductorの今後の展開
- ▶ 4. お知らせ



経済産業省 補助金事業への採択

● 平成25年度～平成27年度の経済産業省補助金事業に採択

- 平成25年度 産業技術実用化開発事業費補助金（ソフトウェア制御型クラウドシステム技術開発プロジェクト）
平成26年度 中小企業等のクラウド利用による革新的省エネ化実証支援事業 クラウド基盤ソフトウェア導入実証
平成27年度 中小企業等省エネルギー型クラウド利用実証支援事業（クラウド基盤ソフトウェア導入実証）

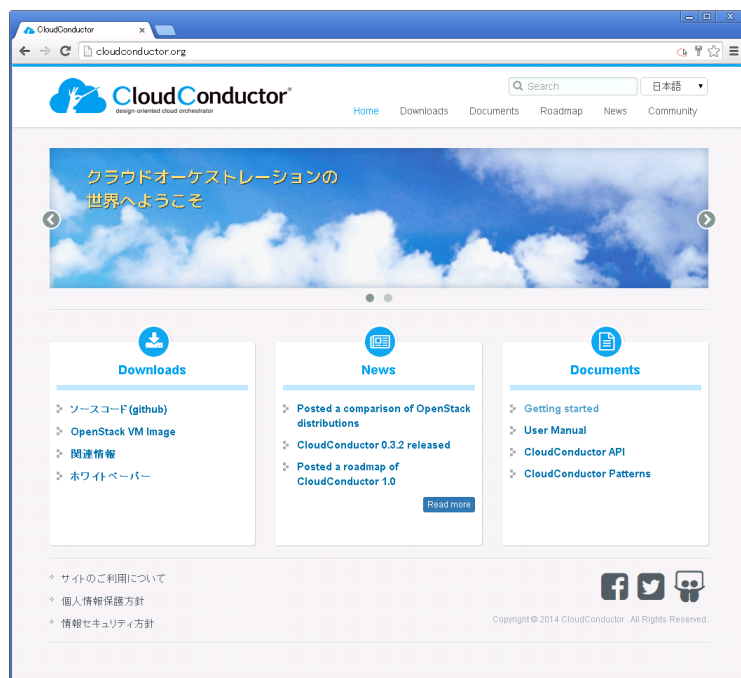


The image displays three overlapping screenshots of the Ministry of Economy, Trade and Industry (METI) website, specifically the 'Grant Selection Results' (採択結果) pages for different fiscal years and programs.

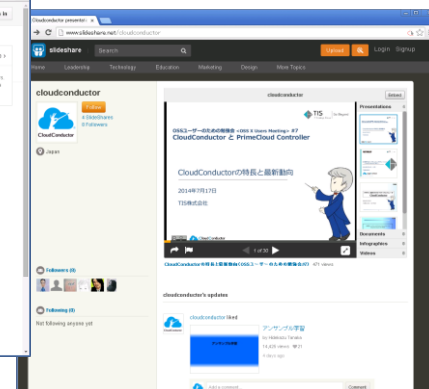
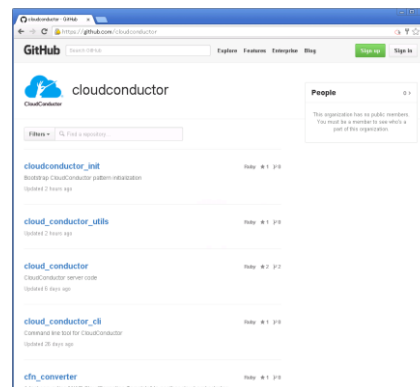
- Top Screenshot (Fiscal Year 25):** Shows the '平成25年度「産業技術実用化開発事業費補助金（ソフトウェア制御型クラウドシステム技術開発プロジェクト）」に係る交付先の採択結果について' (Selection results of grant recipients for the 'Industrial Technology Practicalization Development Grant (Software-controlled Cloud System Technology Development Project)' for FY25). It lists the selected recipients: 1. TIS株式会社, 2. 伊藤忠テクノソリューションズ株式会社, 3. 株式会社インターネットイニシアティブ.
- Middle Screenshot (Fiscal Year 26):** Shows the '平成26年度中小企業等のクラウド利用による革新的省エネ化実証支援事業クラウド基盤ソフトウェア導入実証' (Cloud-based software introduction real proof for innovative energy-saving support for small and medium-sized enterprises using cloud services for FY26). It lists the selected recipients: 1. 東京システムハウス株式会社, 2. ミラクルリナックス株式会社, 3. TIS株式会社, 4. 伊藤忠テクノソリューションズ株式会社, 5. 株式会社インターネットイニシアティブ, 6. 京セラコミュニケーションシステム株式会社.
- Bottom Screenshot (Fiscal Year 27):** Shows the '平成27年度中小企業等省エネルギー型クラウド利用実証支援事業（クラウド基盤ソフトウェア導入実証）に係る交付先の採択結果について' (Selection results of grant recipients for the 'Energy-saving cloud service support for small and medium-sized enterprises (Cloud-based software introduction real proof)' for FY27). It lists the selected recipients: 1. 伊藤忠テクノソリューションズ株式会社, 2. 株式会社インターネットイニシアティブ, 3. TIS株式会社, 4. 東京システムハウス株式会社, 5. ミッドラジャパン株式会社, 6. ミラクルリナックス株式会社.

CloudConductorの情報公開

- 開発成果は**全てオープンソース (Apache License 2.0)** として公開
※ VDCもオープンソース化予定



公式サイト <http://cloudconductor.org>



ソーシャル

<https://twitter.com/ccndctr>
<https://www.facebook.com/cloudconductor>
<https://github.com/cloudconductor>
<http://www.slideshare.net/cloudconductor>

Interop Tokyo 2015 SDI ShowCaseへの出展

- CloudConductorは **Interop Tokyo 2015** に出展予定
(2015年6月10日 水 ~ 12日 金 @ 幕張メッセ)
- オープンステージにてCloudConductorのセミナーを実施
✓ 10日 水 14:15 - 14:45 & 11日 木 15:45 - 16:15
- CloudConductorとVDC研究成果のライブデモを実施

