

クラウドソフトウェアの継続的デリバリについて

横山重俊[†] 吉岡信和[†]

クラウド環境がもたらすソフトウェア開発および運用変化の一例として、クラウドソフトウェアの継続的デリバリについて説明し、それを支援するツール *dodai* について紹介する。また、*dodai* を用いて試作したクラウド基盤の継続的インテグレーション環境についても触れる。

Continuous Delivery of Cloud Software

SHIGETOSHI YOKOYAMA, NOBUKAZU YOSHIOKA

We introduce continuous delivery of cloud software as an example of software development and deployment trends. We also explain how our open source continuous delivery support tool *dodai* works in real environments.

1. はじめに

クラウドコンピューティング技術の導入がソフト開発および運用に変革を引き起こしている。本稿では、特に変化の激しいクラウド基盤とそれを前提に開発されたソフトウェアについて、開発・運用現場に起こっている継続的デリバリと呼ばれる変化について概観する。それと同時に、NII が中心となって開発しているオープンソースソフトウェア *dodai* [1] およびそれを活用したクラウド基盤の継続的インテグレーション環境について紹介する。

2. クラウドアプリケーションとクラウド基盤の開発と運用に関する課題

ビジネス変化に迅速に対応するため、クラウドソフトウェアの開発から運用までのサイクルが益々短縮されている。また、それを支えるクラウド基盤も技術変化が激しくその開発・運用周期も同様である。どちらも開発環境での開発・試験から本番環境へのデプロイメントまでの速度と信頼性を向上させざるを得ないという課題を持つ。クラウド基盤側はそれに加えて、実装される仮想化技術の種類が多様でデプロイされる可能性のある全ての技術組合せでの試験が、コード変更毎に必要となる。

これらの課題解決の一つのアプローチとして継続的インテグレーション/継続的デリバリに取り組む機関が増えている。(図1 参照)

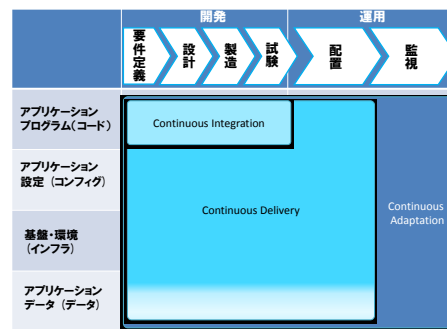


図 1 継続的インテグレーションと継続的デリバリ

3. クラウドソフトウェアの継続的デリバリ

前章で述べた継続的インテグレーションやデリバリという取り組みはアジャイルなソフトウェア開発一般に適用できるものである。しかしながら筆者らは特にその変化が顕著なクラウドソフトウェア(アプリケーションと基盤)に着目し、その開発手法および運用手法の変化を支える技術を研究している。

3.1. クラウドアプリケーションの継続的デリバリ

クラウドアプリケーションとは、その実行環境としてクラウド基盤を使うことをアプリケーション開発時から意識して書かれたものであり、例えば負荷増加に伴った自動スケールアウトの機構などクラウド基盤を活用するコードをアプリケーション内部に包含しているもののことを言う。

3.2. クラウド基盤の継続的デリバリ

クラウド基盤は、クラウドアプリケーションから呼び

[†] 国立情報学研究所
National Institute of Informatics

出されるマシン管理機能を実装していると同時に、クラウド基盤側からクラウドアプリケーションへの通知機構を備えている。

これらソフトウェアに関する継続的デリバリーを実現するためには、開発環境、試験環境、本番環境における構成情報(コード、アプリケーション、基盤、データ)をバージョン指定で、自動的に再現できる仕組みが必要となる。また、クラウド基盤のそれについては仮想化技術(コンピュート、ネットワーク)の様々な種類に対しての試験が必要であるため、ベアメタル環境上の環境構築も自動で実施できる必要がある。

4. 継続的デリバリー支援ツール dodai

4.1. Dodai

ベアメタルマシンもあたかも仮想マシンのように API 経由で制御し、さらにはネットワーク機器も同様に制御することで複数のベアマシンをクラスタとして一つの独立したプライベートクラウドとして自動構築するサービス(Cluster as a Service: CaaS と呼ぶ)をオープンソースソフトウェア dodai は実現し、継続的デリバリーを支援する。(図 2 参照)

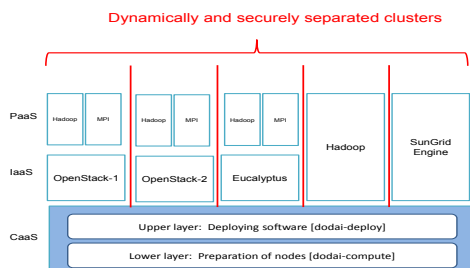


図 2 dodai の利用例

Dodai は、物理マシンで構成されるクラスタを管理する dodai-compute と、物理、および仮想マシン上のソフトウェア環境を管理する dodai-deploy で構成される。

(1) Dodai-compute

各クラスタ環境は、一つの仮想閉域ネットワーク上に構築されるため、他のクラスタ利用者と排他的なアクセス制御が可能である。

(2) Dodai-deploy

クラスタ中のどのマシンに対してどのソフトウェアを、といった設定で構築するかを Proposal と呼ぶ設定ファイルによって宣言できる。

Dodai-deploy の基本的な動作手順は次の通りである。Deploy サーバは、インストールの要求があると、ソフトウェア自動設定ツール(Puppet)用の設定ファイルを生成し、指定されたマシンにそれを送り込む。本方式

では、多数のソフトウェアを並行インストールし、環境構築時間を短縮するために、オペレーションを並行実行するフレームワーク(MCollective)を経由してインストールの指示を各マシンに出している。

Dodai-compute は EC2 互換の API を持っているため、dodai-compute で設定された物理マシンクラスタ、および仮想マシンのクラスタ環境に対して、必要な OS・ソフトウェア環境を dodai-deploy により動的に構築することができる。(図 3 参照)

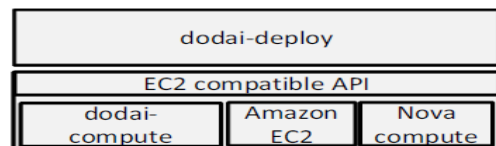


図 3 Dodai-compute API

この特徴により、物理マシンと仮想マシンをターゲットとした継続的デリバリーをサポートできる。

4.2. 継続的インテグレーションへの dodai の適用

クラウド基盤(OpenStack)の継続的インテグレーションのために dodai を使ったクラウド基盤の継続的デリバリーの例を図 4 に示す。

Jenkins のテストジョブを改修し以下を実行。

- dodai により Openstack テスト環境を構築
- Smoke Test の実行コマンド呼び出し

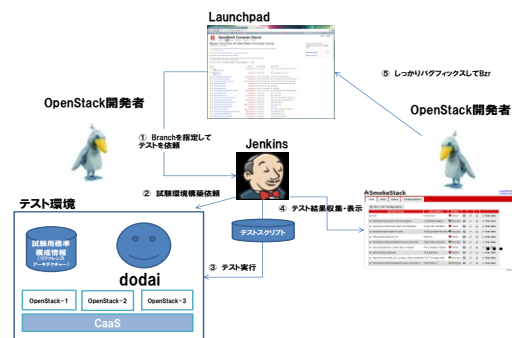


図 4 クラウド基盤の継続的インテグレーションへの dodai の適用

5. まとめ

ビジネス要求の変化への対応力を高めるために不可欠となった継続的インテグレーションと継続的デリバリーをオープンソースソフトウェア dodai で支援できる。今後は、監視フェーズへの適用である継続的アダプテーション(図 1 最右部)を視野に入れた dodai の拡張を実施して行きたい。

参考文献

- [1] dodai プロジェクト: <https://github.com/nii-cloud/dodai>