

動的配置法によるハイブリッドクラウドの 運用管理コスト最小化

江丸 裕教^{1,2,a)} 高井 昌彰^{3,b)}

受付日 2012年7月1日, 採録日 2013年1月11日

概要: インターネット経由の一般向けサービスであるパブリッククラウドと、組織内に閉じたサービスであるプライベートクラウドを組み合わせたハイブリッドクラウドが注目されている。ハイブリッドクラウドの効率的な運用には、ハイブリッドクラウドのシステム全体の運用管理コストを最小化するための手法が必要であり、マイグレーションのコスト評価に加え、アプリケーションの動的な挙動への適応が不可欠である。そこで本論文では、プライベートクラウドの使用率を一定に保つことにより、ハイブリッドクラウドの運用管理コストを最小化する動的配置法を提案する。本手法は、稼働中のアプリケーションの特性をふまえ、マイグレーションを必要最小限に抑えることを特徴とする。また、シミュレーション環境での評価実験により、提案手法の有効性を示す。

キーワード: ハイブリッドクラウド, 動的配置法, 最適化

Optimization of the Hybrid Cloud Operation by Dynamic Placement Method

HIRONORI EMARU^{1,2,a)} YOSHIAKI TAKAI^{3,b)}

Received: July 1, 2012, Accepted: January 11, 2013

Abstract: As economic pressure builds globally, enterprises, municipalities and universities are starting to look at hybrid cloud build by combining public cloud and private cloud as a potential choice to reduce the total cost of ownership for IT. To optimize the hybrid cloud, it is necessary to minimize the operational cost. That is, in addition to consideration of migration cost, adaptation to dynamic behavior of the application is necessary. In this paper, we propose a dynamic placement method which minimizes the operational cost of the hybrid cloud by keeping the utilization of the private cloud at the constant range. This technique is characterized by minimizing the migration based on a dynamic behavior of applications. We show the effectiveness of the proposed method by evaluating in a simulation environment.

Keywords: hybrid cloud, dynamic placement method, optimization

1. はじめに

必要ときに IT リソースを必要な分だけ使うことによ

¹ Hitachi America, Ltd.
2845 Lafayette Street, Santa Clara, CA, 95050, USA

² 北海道大学大学院情報科学研究科
Graduate School of Information Science and Technology,
Hokkaido University, Sapporo, Hokkaido 060-0814, Japan

³ 北海道大学情報基盤センター
Information Initiative Center, Hokkaido University,
Sapporo, Hokkaido 060-0811, Japan

a) hironori.emaru@wm.hitachi.com

b) takai@iic.hokudai.ac.jp

り「所有から利用へ」の転換を促進するクラウドコンピューティング（以下クラウド）が注目されている [1], [2]。インターネット経由の一般向けサービスであるパブリッククラウドと、組織内に閉じたサービスであるプライベートクラウドに加え、近年は両者を組み合わせたハイブリッドクラウドを構築するケースが出てきている。今後は、ハイブリッドクラウドに複数のアプリケーションをデプロイして連携させる利用形態が一般化していくものと考えられる [3]。

我々はハイブリッドクラウドの実現にあたり、クラウド

サービスの仲介、統合、調停などを行う存在であるクラウドブローカ [4] に着目している。クラウドブローカの 1 つの実現形態として、プライベートクラウドおよび複数の IaaS 型パブリッククラウドを束ねて管理する技術が確立しつつある [5]。本技術をベースに、各組織の IT 部門が、プライベートクラウドを含む複数の IaaS クラウドから調達した IT リソースを集約管理することにより、ユーザ部門には複数クラウドの違いを意識させない技術の確立を目指している。

ハイブリッドクラウド上にデプロイされたアプリケーションに対し、想定以上の負荷が発生したり、当初とは異なる利用方法が展開されたりするケースがある。一般にアプリケーションの挙動には動的な部分があり、デプロイ時にすべての挙動を予測できるわけではない。また、各々のアプリケーションのデプロイ要求がどのタイミングで発生するかを事前に知ることは困難である。したがって、ハイブリッドクラウドにアプリケーションをデプロイする処理には、事前に予想できない動的な挙動に対する適応性が必要である。

そこで本論文では、ハイブリッドクラウド上でアプリケーションを運用する場合に、運用コストを最小化することを目的としたアプリケーションのデプロイおよびマイグレーション計画を決定する動的配置法を提案し、その有効性をシミュレーション環境で評価する。

2. 本研究の位置づけ

2.1 従来研究とその問題点

地理的に分散した複数のシステムを利用する仕組みとして、文献 [6], [7] に示されるように 1990 年代半ばからグリッドコンピューティング（以下グリッド）の研究が行われてきた。グリッドはインターネットなどの広域のネットワーク上にある計算資源を結びつけ、1 つの複合したコンピュータシステムとしてサービスを提供する仕組みである。提供されるサービスは主に計算処理とデータの保存/利用に大別される。一般にグリッドはヘテロ環境を対象としている。文献 [8] はハードウェアや OS の差異を吸収する Globus と呼ばれるミドルウェアを用いて環境を構築している。グリッドにアプリケーションをデプロイする方法には文献 [9] をはじめとした多くの研究があるが、経済性に着目するブローカモデルを導入したものに文献 [10], [11] がある。ブローカモデルでは、個々のシステムがリソース空き状況をマーケットに登録するとともに、利用者はタスク種別と上限金額、制限時間などの利用条件に登録し、これらの情報からブローカがタスクの最適な割当てを行う。

一方、仮想化技術の進展、ネットワーク環境の改善、モバイルを含むアクセス端末の多様化などを背景にクラウドが注目を集めている。クラウドはグリッドの延長線上に位置づけられる技術であるが、文献 [12] ではその相違とし

て、仮想化技術によるコンピューティングリソースの抽象化、使用者に上限を意識させないスケーラビリティといった点をあげている。

本論文におけるデプロイとは、アプリケーションが動作可能な状態のサーバを仮想化技術により作成することを意味する。文献 [13], [14] では負荷に応じて動的に仮想サーバを作成・消去する仕組みが述べられている。また、ハイブリッドクラウドに対するアプリケーションのデプロイに関する研究例として、複数のアプリケーションをプライベートクラウドまたはパブリッククラウドのどちらに配置すべきかを混合整数計画問題（以下 MIP）として定式化し、線形回帰モデルにより評価した文献 [15] がある。この研究では、クラウドのコスト構造とアプリケーションの挙動をすべて MIP ソルバに入力し、最適配置解を得る手法を採用している。そのため、アプリケーションの挙動が事前にすべて把握できる場合にのみ適用できる。また、線形回帰モデルによりパブリッククラウドの使用量とアプリケーションの関係を分析している。その結果、アプリケーション数（正の相関）、計算量（負の相関）、ストレージ量（正の相関）の順で関連が高いことを指摘している。すなわち、計算量が少なく、ストレージ量が多いアプリケーションが大量にある場合に、パブリッククラウド活用の効果が最も高いと結論付けている。しかし、この分析はパブリッククラウドとプライベートクラウドのコスト構造に依存するため、汎用的な知見とはいえない。

また、文献 [16] では、計算と通信のコストを加味したハイブリッドクラウドのコストモデルを構築し、プライベートクラウドは使用の有無にかかわらず維持コストがかかること、パブリッククラウドは従量課金だがプライベートクラウドよりも単価が高いこと、クラウドで扱う負荷は時間変動することを前提として、すべてをプライベートクラウドもしくはパブリッククラウドで処理するよりも、ハイブリッドクラウドで扱った方が低コストになることを導いている。この研究では、アプリケーションの負荷が既知であることを前提にプライベートクラウドに保持すべきリソース量を決定し、プライベートクラウドのリソースはつねに最大限活用されることを仮定している。しかし、マイグレーションの方法やコストに関しては言及されていない。

また、文献 [17] ではハイブリッドクラウドを実現するうえでの重要な要素技術の 1 つであるクラウド間のライブマイグレーション技術確立について議論している。

ある組織がハイブリッドクラウドを活用しようとする際に、これまでの研究では解決されていない課題が 2 つある。

第 1 の課題はアプリケーションの動的な挙動への適応である。ハイブリッドクラウド上にデプロイされたアプリケーションが、見込み以上のサービス要求や前提とは違った形態での利用により、デプロイ時の想定と異なる振舞いを示すケースがある。また、いつでもどれくらいの頻度でデプ

表 1 各パブリッククラウドの認証取得状況
Table 1 Security certifications of major public cloud providers.

	SAS 70	SSAE	ISO/IEC27001	PCI-DSS
Amazon Web Services	○	○	○	○
Rackspace	○	○	○	○
Savvis	○	○	○	×
Bluelock	○	○	×	×
GoGrid	○	○	×	×
SoftLayer	○	○	×	×
Google Compute Engine	×	×	×	×

ロイ要求が発生するのかをあらかじめ知ることも困難である。クラウドは一般的に大容量のリソースプールを持っているため、想定以上の負荷や負荷変動を吸収することが可能である。しかしコストの観点から、動的なアプリケーションに対してプライベートクラウドとパブリッククラウドを最適に使い分ける手法は知られていない。

第2の課題は文献[18]にもあげられているセキュリティである。アプリケーションやデータごとに対応しなければならない様々なセキュリティ基準がある。たとえば、情報セキュリティマネジメントシステムに求められる規格を定めた ISO/IEC 27000 シリーズや、米国公認会計士協会が監査用に定めた SAS 70 などが知られている。このように様々なセキュリティ基準が存在しているが、クラウドごとに取得されている認証基準は異なっている。これら基準を考慮しつつアプリケーションを適切にデプロイする手法は知られていない。

2.2 対象のモデル化

本研究の目的は、ハイブリッドクラウドにアプリケーションをデプロイする際に、アプリケーション相互の関係を考慮したうえで、セキュリティ要件を満たしつつ全体の運用管理コストを最小化するデプロイ方法を構築することである。本節では、ハイブリッドクラウドおよびアプリケーションのモデル化を行う。

(1) 対象時間

対象となる時間を $t = \{0, 1, 2 \dots T\}$ と離散時間で表す。

(2) セキュリティ

クラウドサービスプロバイダは、様々な認証を取得している。それぞれの認証の間には厳密な順序関係はなく、独立に取得が可能だが、Amazon Web Services (AWS), Rackspace, Savvis, Bluelock, GoGrid, SoftLayer, Google Compute Engine など主要 IaaS クラウドの認証取得状況を調査したところ、SAS 70, 内部統制保証報告に関する基準である Statement on Standards for Attestation Engagements (SSAE), ISO/IEC 27000 シリーズ, クレジット業界におけるグローバルセキュリティ基準である PCI-DDS について、表 1 に示すように各クラウド間で認証取得状

況に応じた順列をつけることが可能であることが確認できた。そこで、想定するすべてのクラウドがそれぞれセキュリティレベルを持つと仮定し、各アプリケーションの要求セキュリティレベル以上のセキュリティレベルを持つクラウドにのみ、そのアプリケーションを配置可能とする。

(3) アプリケーション

一般に、ある組織が利用する IT システムは複数のアプリケーションから構成される。この IT システムのうち、ハイブリッドクラウドに配置されている全アプリケーションを $A = \{a_1, a_2, a_3 \dots a_N\}$ で表す。アプリケーション a_j は、任意の時間 t における計算量 p_{jt} とストレージに格納されるデータ量 (ストレージ量) d_{jt} のほか、ライフタイムを通して一定のセキュリティレベル s_j が定義されているものとする。アプリケーション a_j は、セキュリティレベル s_j 以上のセキュリティレベルを持つクラウドに配置可能とする。

アプリケーションデプロイ時には、使用期間を指定する。使用期間が事前に予測できない場合には無期限とし、不要になった時点でアンデプロイする。

(4) ハイブリッドクラウド

ハイブリッドクラウド H は複数のクラウドから構成され、これを $H = \{h_1, h_2, h_3 \dots, h_I, \dots, h_M\}$ と表す。このうち、 $I = \{h_1, h_2, h_3 \dots h_I\}$ はプライベートクラウド、 $E = \{h_{I+1}, h_{I+2}, h_{I+3} \dots h_M\}$ はパブリッククラウドを表すものとする。各クラウドは単位時間あたりの最大計算量 P_i と、単位時間あたりの最大ストレージ量 D_i 、およびセキュリティレベル S_i を持つ。

IT リソースを所有せずにサービスを利用だけのパブリッククラウドと、IT リソースを所有しているプライベートクラウドとでは、課金モデルが異なる。

パブリッククラウドでは、単位時間の使用計算量および使用ストレージ量に応じた従量課金が行われる。各パブリッククラウド h_i ($I+1 \leq i \leq M$) の単位時間、単位計算量あたりの課金額を Π_i 、単位時間、単位容量あたりのストレージ課金額を Δ_i とする。パブリッククラウドは利用者が単位時間ごとに利用するリソース量を増減することができる。ストレージに関しては、リクレーンと呼ばれる不

要ページ解放処理により使用ストレージ量を減らす技術もあるが、クラウドでは一般的でないため、本論文では使用ストレージ量はアプリケーションが使用している期間中に単調増加するものとする。

一方、プライベートクラウドでは、アプリケーションを稼働させるための運用管理コスト (Operational Expense: OPEX) に加え、利用の有無にかかわらず IT リソースを所有するための設備投資費 (Capital Expense: CAPEX) が必要である。一般に設備投資費は、減価償却という形で一定の期間で費用化される。OPEX と CAPEX は互いに素であり、この和をプライベートクラウドのコストと定義する。プライベートクラウド h_i ($1 \leq i \leq I$) の単位時間、単位計算量あたりの OPEX を Π_i 、単位時間、単位ストレージ量あたりの OPEX を Δ_i とし、CAPEX を C_i とする。

(5) 目的関数

ハイブリッドクラウドにデプロイするアプリケーションの運用管理コストは、複数のアプリケーションを実行するために必要なプライベートクラウドの OPEX および CAPEX と、パブリッククラウドの利用に応じた従量課金からなる。したがって、目的関数は以下のように表すことができる。

目的関数：

$$\min \sum_{t=0}^T \left(\sum_{i=1}^I C_i + \sum_{i=1}^M \sum_{j=1}^N x_{ijt} (\Pi_i p_{jt} + \Delta_i d_{jt}) \right) \quad (1)$$

制約条件：

$$\forall a_j \in A, a_j \text{ is deployed on } h_i \in H \quad (2)$$

$$\forall a_j \in A, s_j \leq S_i \quad (3)$$

$$\forall h_i \in H, \forall t \in T, \sum_{j=1}^N x_{ijt} p_{jt} \leq P_i \quad (4)$$

$$\forall h_i \in H, \forall t \in T, \sum_{j=1}^N x_{ijt} d_{jt} \leq D_i \quad (5)$$

目的関数 (1) の第 1 項目はプライベートクラウドを所有していることにともない発生する CAPEX、第 2 項目はプライベートクラウドにおける OPEX またはパブリッククラウドの使用分に対して発生する従量課金のコストである。

制約条件式 (2) はすべてのアプリケーションがいずれかのクラウドに配置されることを表し、式 (3) は、すべてのアプリケーションはそのアプリケーションが求める水準以上のセキュリティレベルを持つクラウドに配置されることを表す。式 (4) は、デプロイされたアプリケーションの計算量が、各クラウドの最大計算量をつねに下回っている必要があることを、式 (5) はデプロイされたアプリケーションが必要とするストレージ量が各クラウドの最大ストレージ量をつねに下回っている必要があることを表す。ここで x_{ijt} は時刻 t において、クラウド h_i にアプリケーション a_j が配置されているときに 1、配置されていないときに 0 と

なる 0-1 変数とする。

3. 動的配置法

3.1 目的関数の最小化に求められる要件

ハイブリッドクラウドにデプロイするアプリケーション全体での運用コストを最小化するための手法は、以下に示す 3 つの要件を満たすことが必要である。

第 1 は複数アプリケーションの動的な挙動への適応性である。一般にアプリケーションには負荷や使用データ量の変動という動的な性質がある。稼働期間も様々であり、デプロイされるタイミングも事前には分からないことが多い。これらの動的な変化に適応できる必要がある。

第 2 はスケーラビリティである。数千ものクラウドを使いこなすことは現実的ではないが、ある組織がクラウドにデプロイしている全アプリケーション数が 10^4 といった規模になることは十分考えられる。そこで最小化手法のスケーラビリティの観点から、 10^2 のオーダのクラウドと 10^4 のオーダのアプリケーションを扱えることを要件とする。

第 3 は手法の汎用性である。プライベートクラウドだけでは、プロプライエタリな仕組みを実装することが可能だが、パブリッククラウドも活用しようとした場合、汎用的な機能のみを用いる必要がある。本論文で対象とする全クラウドは、汎用的な機能として、負荷に応じた計算資源量のスケールアウトおよびダウン機能、ストレージのオンデマンド容量拡張機能、性能情報取得機能および他クラウドとのサーバイメージのマイグレーション機能を備えるものと仮定する。取得可能な性能情報は、レスポンスタイムに加え、使用計算資源量と使用ストレージ量とする。

3.2 問題解決のアプローチ

ハイブリッドクラウドの運用管理コストを最小化する第 1 の基本方針として、プライベートクラウドのリソースの使用率を指標とし、これを一定の範囲内に維持することを考える。プライベートクラウドのリソース利用率を高く保つことにより、従量課金で利用するパブリッククラウドの利用を極力抑えるためである。なおプライベートクラウドの使用率の目標を 100% に設定しないのは、プライベートクラウドがリソース不足になるような事態を避けるためである。アプリケーションの不測の動的な挙動からリソース不足が発生すると、アプリケーションが停止したり、新規業務をデプロイできなかつたりするケースが起こりうる。維持すべき一定範囲を表すために、 $T_b < T_c < T_u$ を満たす目標値 T_c 、下限閾値 T_b 、上限閾値 T_u を導入する。パブリッククラウドは使用リソース量に応じてのみ課金されるが、プライベートクラウドでは使用の有無にかかわらず OPEX と CAPEX が必要となる。文献 [16] にも示されているとおり、単位リソースあたりの利用コストを比べると、パブリッククラウドの従量課金額よりもプライベートク

ウドの OPEX の方が一般的に低い。そのため、同じ処理をハイブリッドクラウド環境で行う場合、プライベートクラウドの使用率が高いほどハイブリッドクラウド全体の運用管理コスト最小化に寄与すると考えられる。

また、運用管理コスト最小化における第2の基本方針として、デプロイとマイグレーションを組み合わせることを考える。これまでデプロイ先の選択手法は複数提案されているが、コスト最小化の観点からマイグレーションを活用する手法は知られていない。たとえば文献[16]では、アプリケーション間の通信量を加味し、オンプレミスから動かさないアプリケーションとの通信量が多いアプリケーションはプライベートに、それ以外のアプリケーションはパブリックに配置する手法を提案している。また文献[15]は、各アプリケーションの使用リソース量と各クラウドのリソース量を MIP として定式化し、この解に基づきデプロイを行う。これらの手法は通信量と負荷が既知であることが前提であり、アプリケーションが次々とデプロイとアンデプロイを行う動的な環境には対応できない。そこで本論文では、マイグレーションとデプロイを組み合わせることによって動的な環境への適応性とコスト最小化の両立を狙う。

以上の基本方針に基づいて、プライベートクラウドの使用率を $T_b \sim T_u$ の範囲に保つため、新規アプリケーションのデプロイ要求発生時にプライベートクラウドの使用率が T_c 未満であればプライベートクラウドにデプロイし、プライベートクラウドの使用率を向上させる。プライベートクラウドの使用率が T_c 以上の場合、アプリケーションのセキュリティレベルの制約からプライベートクラウドしか使用できない場合を除き、パブリッククラウドにデプロイする。

またマイグレーションには、文献[19]で示されているように時間および実行中の性能劣化のコストがあるため、本論文では、プライベートクラウドとパブリッククラウドの間のマイグレーションに限定し、パブリッククラウド間のマイグレーションは行わないこととする。これは、パブリッククラウド間のマイグレーションが技術的に成熟していないことに加え、一般に似た特性を持つパブリッククラウド間でマイグレーションを行うよりも、特性が異なるプライベートクラウドとパブリッククラウドの間でマイグレーションを行う方が効果的であると考えられるためである。

3.3 動的配置法のデプロイとマイグレーション

3.3.1 上限閾値を超えた場合

プライベートクラウドの使用率が上限閾値 T_u を超えた場合には、プライベートクラウドからパブリッククラウドにマイグレーションを行う。アプリケーションごとに評価値を算出し、評価値の大きなものから順にアプリケーションを選択し、それらをマイグレーションの対象とする。ただし、デプロイ時に使用期間が指定されたアプリケーション

の中に期間 p_b 以内に終了予定のアプリケーションがあり、そのアプリケーションの終了によって上限閾値 T_u を下回る見込みがあれば、 T_u よりも高い値に設定された第2上限閾値 T_{u_2} を超えるまではマイグレーションを行わない。

上限閾値を超えたのが計算資源使用率か、あるいはストレージ使用率かに依存して、評価値の算出方法は異なる。

(1) 計算資源使用率

計算資源使用率が上限閾値を超えた場合は、セキュリティレベルを満たしてパブリッククラウドに配置可能なアプリケーションを対象に、下記の2つのパラメータから評価値を算出する。第1は、計算量に対するストレージ量の割合である。マイグレーションの負荷はストレージ量にその多くを依存するため、計算量に対するストレージ量の割合が小さいアプリケーションほどマイグレーションの効果は高い。したがって、計算量に対するストレージ量の割合が小さいアプリケーションを優先して選択する。第2は、期間 p_s における計算量の標準偏差である。プライベートクラウドのリソース使用率を一定に保つためには、負荷変動の大きなアプリケーションよりも、負荷変動が小さいアプリケーションをプライベートクラウドに配置しておくのが望ましい。

これら2つの点に基づき、プライベートクラウドにおける負荷の安定性の維持を目的にプライベートクラウドにデプロイされている各アプリケーション a_j の評価値 $a_j^{m_1}$ を式(6)で算出する。

$$a_j^{m_1} = \frac{p_j^{p_s}}{d_j^{p_s}} \times \alpha_1 + \sigma(p_j^{p_s}) \times \beta_1 \quad (6)$$

ここで、 $p_j^{p_s}$ はアプリケーション a_j の期間 p_s の最大計算量、 $d_j^{p_s}$ はアプリケーション a_j の期間 p_s の最大ストレージ量、 $\sigma(p_j^{p_s})$ はアプリケーション a_j の期間 p_s の計算量の標準偏差である。 α_1 および β_1 は係数である。

得られた各アプリケーションの評価値 $a_j^{m_1}$ を降順にソートし、マイグレーションによってプライベートクラウドの計算資源使用率が T_c を下回るまでアプリケーションを順に選択し、それらをマイグレーション対象とする。

(2) ストレージ使用率

ストレージ使用率が上限閾値を超えた場合は下記の2つの観点からマイグレーション対象を選択する。第1はストレージ量の期間 p_s における増加率である。マイグレーションの効果を高めるためには、単にストレージ量の多いアプリケーションではなく、ストレージ量の増加率が高いアプリケーションをマイグレーションすべきである。第2は、期間 p_s の計算量の標準偏差である。プライベートクラウドには負荷変動の小さいアプリケーションを配置しておくことが望ましい。

これら2つの点から、プライベートクラウドにデプロイされている各アプリケーション a_j の評価値 $a_j^{m_2}$ を式(7)

表 2 アプリケーション

Table 2 Applications.

#	ID	Application	Lifetime	Security Level	Pattern		Deploy Frequency (Number of Apps per day)
					Compute	Storage	
0	OLTP1	OLTP	2-5 years	4-5	Constant	Increase	0.1
1	OLTP2		2-5 years	4-5	Periodic	Increase	0.1
2	OLAP1	OLAP	1 month- 1 year	2-4	Burst	Constant	0.25
3	OLAP2		3 days - 1 week	2-4	Constant	Constant	1
4	DevTest	Test / Development	3 days - 1 month	1-3	Random	Increase	2.5
5	Web1	Web Server	6 months - 2 years	2-4	Constant	Constant	0.1
6	Web2		6 months - 2 years	2-4	Increase	Constant	0.6
7	Web3		6 months - 2 years	2-4	Periodic	Constant	0.25
8	Mail	Mail Server	5 years	3	Constant	Increase	0.05
9	File	File Server	2 -5 years	2-4	Constant	Increase	0.05

で算出する.

$$a_j^{m2} = d_j^{gp_s} \times \alpha_2 + \sigma(p_j^{p_s}) \times \beta_2 \quad (7)$$

ここで, $d_j^{gp_s}$ はアプリケーション a_j の期間 p_s におけるストレージ量の増加率である. α_2 および β_2 は係数である.

各アプリケーションの評価値 a_j^{m2} を降順にソートし, マイグレーションによってプライベートクラウドのストレージ使用率が T_c を下回るまでアプリケーションを順に選択し, それらをマイグレーション対象とする.

3.3.2 下限閾値を下回った場合

プライベートクラウドの使用率が低くなり, 下限閾値 T_b を下回った場合, T_c 以上にすべく, パブリッククラウドからプライベートクラウドへのマイグレーションを行う. プライベートクラウドには計算量およびストレージ量の安定したアプリケーションを持ってくるべきである. そこで以下の3つの観点からマイグレーション対象のアプリケーションを選択する. すなわち第1は期間 p_s におけるストレージ量の増加率, 第2は期間 p_s における計算量の標準偏差である. 第3は残り期間の長さであり, 残っている実行期間の長いアプリケーションが望ましい.

これら3つの点から, パブリッククラウドにデプロイされている各アプリケーション a_j の評価値 a_j^{m3} を式 (8) で算出する.

$$a_j^{m3} = \frac{1}{d_j^{gp_s}} \times \alpha_3 + \frac{1}{\sigma(p_j^{p_s})} \times \beta_3 + \frac{1}{a_j^{t_r}} \times \gamma_3 \quad (8)$$

ここで $a_j^{t_r}$ はアプリケーション a_j の残り実行時間である. a_j の使用期間が無期限に設定されている場合, 第3項目は0となる. α_3 , β_3 および γ_3 は係数である.

各アプリケーションの評価値 a_j^{m3} を降順にソートし, マイグレーションによってプライベートクラウドの計算資源使用率もしくはストレージ使用率のいずれかが T_c を上回るまでアプリケーションを順に選択し, それらをマイグレーション対象とする.

4. 実験

4.1 シミュレーション環境

(1) セキュリティ

各アプリケーションおよびクラウドに5段階のセキュリティレベルを設定する. 各アプリケーションは設定されたセキュリティレベル以上の値を持つクラウドにしかデプロイできないものとする.

(2) アプリケーション

表2に示すアプリケーションを用意する. Application はアプリケーションの種別を表し, OLTP (Online transaction processing) アプリケーション, OLAP (Online analytical processing) アプリケーション, Dev/Test (開発・テスト環境), Web サーバ, メールサーバ, ファイルサーバから構成される. OLTP, OLAP, Web サーバに関しては異なるワークロードを持つ複数のバリエーションを用意したため, アプリケーションは計10種類である. Lifetime はアプリケーションの運用期間である. 実運用ではデプロイ時に既知のケースと未知のケースがあるが, 本シミュレーションではすべて既知であると仮定したうえで, 各アプリケーションの運用期間は本欄に示された範囲で一様分布しているものとし, この範囲でランダムに決定した. Pattern はアプリケーションの所要計算量, ストレージ量の時間推移の概要である. 実際の計算量, ストレージ量の時間推移は, 文献 [20], [21], [22] をベースに, トレースデータから1日ごとのリソースの最大使用量を抽出し, それを1日のリソース使用量とする形で作成した. Deploy Frequency はデプロイ要求頻度である. 本欄に示す値は1日あたりの発生頻度を表し, この値のポアソン分布に基づきデプロイ要求の発生回数を求めた. 以降は1日の最大計算量とストレージ量をその日の使用リソース量として更新を行う. 運用期間が0になったアプリケーションはアンデプロイされ, 使用していたリソースを解放する.

(3) ハイブリッドクラウド

シミュレーション対象のハイブリッドクラウドは、セキュリティレベル5のプライベートクラウド1個と、セキュリティレベルが相互に異なる5個のパブリッククラウドから構成される。プライベートクラウドの計算量、ストレージ量はシミュレーション開始時に決定し、シミュレーション中の拡張は行わず、パブリッククラウドの計算量、ストレージ量は無限とする。

(4) シミュレーションの実施方法

プライベートクラウドの計算資源使用率が70%、パブリッククラウドはいっさい使用していない状態からシミュレーションを開始する。

クラウドの運用管理コストは式(1)により決定される。パブリッククラウドの運用管理コストは従量課金される利用料のみから構成され、前述したように課金単位を1日としたうえで、使用した計算量およびストレージ量に応じて決定される。この式に適用するパラメータは文献[23], [24]を参考にして設定した。マイグレーションは、文献[19]から1アプリケーションあたり、20 MB/secの速度でマイグレーションが行われ、その間スループットが8%低下するものとする。実際にはクラウド間の有効帯域は任意のクラウド間で異なるが、本シミュレーションではこれらのパラメータは固定とする。これらの値からマイグレーションの所要時間、その間の8%のスループット低下を補うために必要となる計算資源量、マイグレーション中に使用されるマイグレーション元/先のストレージ量を求め、追加リソース量に基づくマイグレーションコストを計算して全体のコストに加味する。シミュレーション期間はサーバやストレージといったハードウェアの通常の耐用年数である5年間とする。また、実際には技術の進歩や運用管理スキルの成熟にともない、プライベートクラウド、パブリッククラウドともに、ITリソースの単価は時間の経過とともに下がることが期待される。この恩恵は両クラウドに同等にもたらされる。たとえば、文献[25]は、プライベートクラウドとしてデータセンタを保有するケースと、パブリッククラウドとしてAWSを使用するケースを3年の期間で比較し、各クラウドのコスト構造が長期間にわたり維持されることを示している。単価の低減の度合いが両クラウドで同一であるとするシミュレーション結果に影響を及ぼさないため、今回のシミュレーションでは単価は変動しないものとした。

(5) 比較対象の実装

比較対象を文献[15]に基づき、以下のように定める。最初に手法適用間隔を決定する。従来手法の特徴はプライベートクラウドのリソース使用率を高く維持できることである。したがって、手法適用間隔を、従来手法の特徴が最も活きる課金単位と同一の1日と設定した。そのうえで、アプリケーションの最大計算量およびストレージ量が1日

表 3 閾値の設定

Table 3 Parameters of threshold.

	T_b	T_c	T_u	T_{u_2}
Value	70%	80%	90%	95%

表 4 係数の設定

Table 4 Parameters of coefficient.

	α_1	β_1	α_2	β_2	α_3	β_3	γ_3
Value	1	2	1	2	2	4	1

の最初に分かると仮定し、MIPを解くことによりプライベートクラウドの計算資源およびストレージのリソース使用率を100%に近づけるようにアプリケーションの配置を1日ごとに更新し、それに必要なマイグレーションを行う。これにより、マイグレーション以外のコスト、すなわちプライベートクラウドのOPEXとCAPEX、ならびにパブリッククラウドの従量課金のコストの和は最小になると期待される。以下、本論文ではこれを従来手法と呼ぶ。

(6) 提案手法の実装と予備実験

閾値 T_u , T_c , T_{u_2} を低く設定しすぎるとプライベートクラウドに余剰リソースをかかえることになり、逆に高く設定しすぎるとプライベートクラウドにおいてバースト的な負荷が発生した場合に、マイグレーションが間に合わず、プライベートクラウド全体のシステムダウンのリスクがある。同様に T_b を低く設定しすぎるとプライベートクラウドに余剰リソースをかかえることになり、高く設定しすぎるとマイグレーションが頻発する。そこでこれらのパラメータの設定に関して予備実験を行い、表3に示す値に決定した。同様に予備実験により、係数 α , β , γ を表4に示す値に設定し、期間 p_b , p_s は3日、7日、14日、30日のパターンを試行し、ともに目的関数である運用管理コストに対して最も良い値であった7日間に定めた。

また、予備実験として提案手法のスケラビリティの確認も行った。その結果、 10^2 オーダーのクラウドと 10^4 オーダーのアプリケーションの規模に対しても、一般的なPCを用いて数秒以下の処理時間でマイグレーション対象のアプリケーションを決定できることを確認した。したがって、提案手法による処理がスケラビリティに影響を及ぼすことはない。

4.2 評価

4.2.1 コストの推移

従来手法と提案手法の5年間のコストの推移を、図1と図2にそれぞれ示す。左縦軸はプライベートクラウドのストレージと計算資源のリソース使用率、右縦軸はハイブリッドクラウドのコストの累積値を表している。提案手法の5年間の全コストは、従来手法の56.1%である。最初の1年間だけ見ても、提案手法は従来手法の55.4%であり、

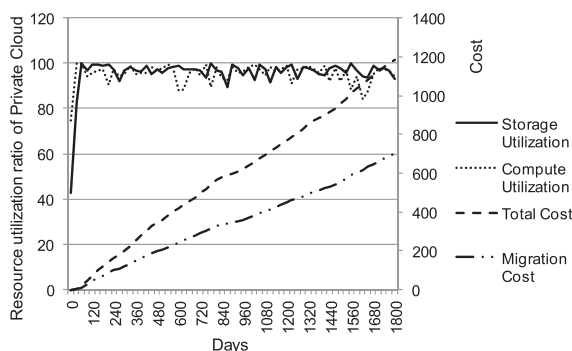


図 1 従来手法

Fig. 1 Conventional method.

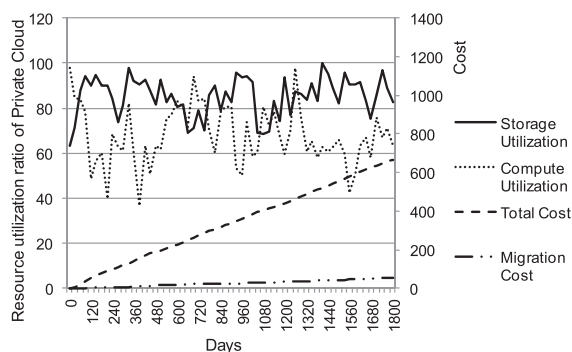


図 2 提案手法

Fig. 2 Proposal method.

表 5 従来手法との比較

Table 5 Comparison with conventional method.

	Conventional Method	Proposal Method
Total Cost	1186.0	665.9
Migration Cost	699.0	54.5
Cost w/o Migration	487.0	611.4
Number of Migrations	1825	360
Storage Utilization	95.7%	82.1%
Compute Utilization	95.2%	66.8%

短期間でも効果が得られている。

表 5 に示すように、マイグレーションコストを除いた残りのコストは従来手法の方が提案手法に比べて 20.3% 少ない。従来手法はプライベートクラウドのリソース使用率を高めることを目的に運用が行われるため、実際、5 年間の平均リソース使用率は 95% を超えている。一方、提案手法の平均リソース使用率は、ストレージ 82.1%、計算資源 66.8% であり、従来方式を大きく下回っている。すなわち、提案手法の効果はマイグレーションコストの少なさに起因している。従来手法の 5 年間のマイグレーション量は提案手法の 12.8 倍、マイグレーション発生回数は 5.1 倍ある。図 3 は 1 日あたりのマイグレーション量の推移を示したものである。提案手法のマイグレーション量がつねに低く抑えられている。

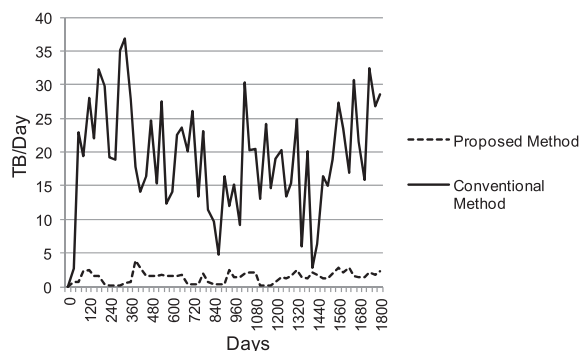


図 3 1 日あたりのマイグレーション量

Fig. 3 Migration amount per day.

表 6 ランダム法との比較

Table 6 Comparison with random method.

	Proposal Method	Random Method
Total Cost	665.9	705.1
Migration Cost	54.5	93.0
Cost w/o Migration	611.4	612.1
Number of Migrations	360	386
Storage Utilization	82.1%	85.9%
Compute Utilization	66.8%	64.8%

4.2.2 アプリケーション選択アルゴリズムの影響

提案手法は、アプリケーションの特性を解析し、リソースの使用量が安定しているアプリケーションはプライベートクラウドで、リソースの使用量が動的に変動しているアプリケーションはパブリッククラウドで稼働させることにより、マイグレーションの発生を抑止しつつプライベートクラウドのリソースの使用率を一定に保つ点を特徴としている。そこで、閾値監視する仕組みは提案手法と同様とし、閾値を超えた際にアプリケーションの特性を解析するのではなく、ランダムにアプリケーションを選択する手法（以下ランダム法と呼ぶ）を比較対象に用いて、提案手法の妥当性をアプリケーション選択アルゴリズムの観点から検証する。

比較実験の結果を表 6 と図 4 に示す。ストレージおよび計算資源のリソース使用率は、提案手法とランダム法に大きな差異はなく、マイグレーションコストを除いたコストもほぼ同じである。しかし、図 4 に示すようにランダム法のマイグレーションコストは提案手法の 1.7 倍と高く、これが全コストの差に結びついている。このように、アプリケーションの特性を解析することでプライベートクラウドとパブリッククラウドに配置するアプリケーションを決定することは有効である。

4.2.3 大きな環境変化への適応

提案手法が環境の変化に動的に適応することを確認するため、シミュレーション開始から 3 年後に、デプロイされているアプリケーションの種類をすべてランダムに変更す

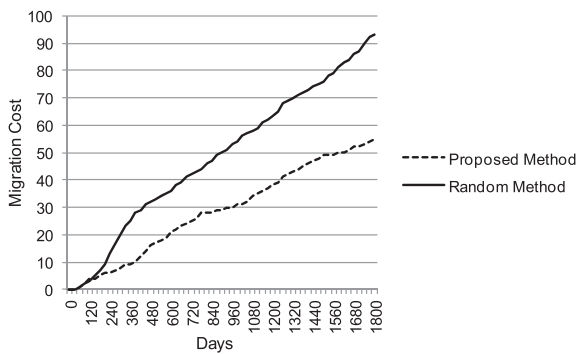


図 4 マイグレーションコスト

Fig. 4 Migration cost.

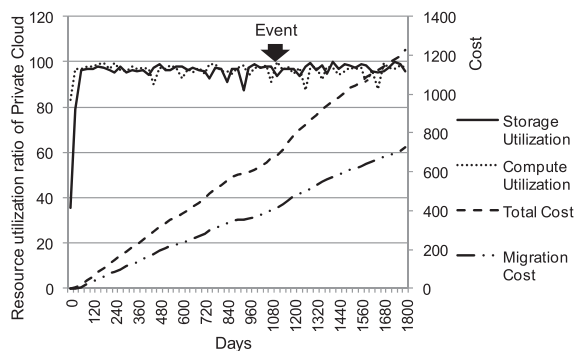


図 5 従来手法

Fig. 5 Conventional method.

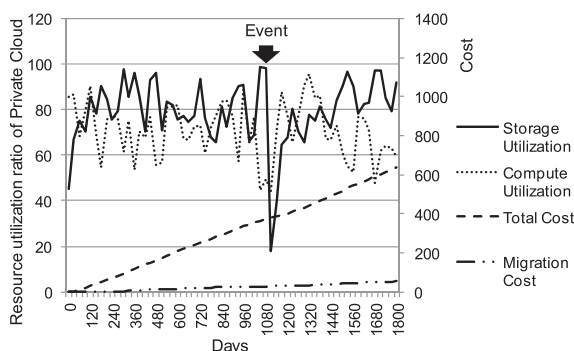


図 6 提案手法

Fig. 6 Proposal method.

るイベントを発生させる。この実験結果を図 5 と図 6 に示す。

従来手法では、当日のアプリケーションの挙動が既知であることを前提にアプリケーションの再配置を毎日行っているため、イベント発生直後にマイグレーション量の増大がみられるものの、プライベートクラウドの使用率は高い値に保たれている。これに対し提案手法では、当日のアプリケーションの挙動が未知であるため、プライベートクラウドにおけるイベント発生直後のストレージ使用率が 18%、計算資源使用率が 44%とそれぞれ落ち込んでいる。今回のケースでは、計算資源使用率が先に最適化されることにより、ストレージ使用率の回復に時間がかかっているが、イベント発生後 60~90 日程度で環境変化に適応し、イ

ベント発生前の状態にはほぼ戻っていることが確認できる。結果として、提案手法の 5 年間の全コストは従来手法に比べて 52.1%であり、動的な環境に対する適応能力も高いことが示された。

5. おわりに

本論文では、ハイブリッドクラウドにおいてアプリケーションを運用する場合に、デプロイとマイグレーションを組み合わせることによってマイグレーションの発生を抑えつつプライベートクラウドの使用率を一定に保つ動的配置法を提案し、その有効性をシミュレーション環境で評価した。その結果、アプリケーションのデプロイ要求の発生頻度や、各々のアプリケーションの負荷特性が分からないハイブリッドクラウドにおいて、従来手法と比較して 5 年間で 43.9%の運用管理コスト削減効果があることを示した。

今後の課題の 1 点目はプライベートクラウドの計算資源使用率の改善である。アプリケーションには、一般にストレージに比べて計算量の時間変動が激しいという特徴がある。時間課金を行うパブリッククラウドにおいてこの傾向は顕著である。結果として 70%~90%のリソース使用率の維持を目的としたものの、提案手法におけるプライベートクラウドの計算資源使用率は 66.8%にとどまっている。本論文のシミュレーションでは係数 α , β , γ の値を固定としたが、プライベートクラウドにおけるストレージ使用率と計算資源使用率の差に応じてチューニングするなどの手法により、マイグレーション量の抑止と計算資源使用率の向上を両立させる改善を検討する。あわせてプライベートクラウドの計算資源使用率とストレージ使用率が乖離してしまった場合には、従来手法である MIP 法に基づく再配置の適用も検討する。

2 点目の課題は、複数クラウドの使いこなしである。本論文ではセキュリティレベルが相互に異なる 5 個のパブリッククラウドを前提とした。そのため、パブリッククラウド間のマイグレーションは実施しないものとしたが、実際には計算量とストレージ量に対する課金割合が異なるクラウドが複数存在する。またそれらクラウド間の有効帯域も異なる。契約期間によって課金額が異なるクラウドも存在する。これらの条件をモデル化し、パブリッククラウド間のマイグレーションの効果もあわせて検討する。

これら課題を解決し、実環境への適用および評価を行っていきたい。

参考文献

- [1] Hayes, B.: Cloud computing, *Comm. ACM*, Vol.51, No.7, pp.9-11 (2008).
- [2] Armbrust, M., Fox, A., Griffith, R., et al.: Above the Clouds: A Berkeley View of Cloud Computing, *Electrical Engineering and Computer Sciences University of California at Berkeley*, No.UCB/EECS-2009-28 (2009).

- [3] Bicer, T., Chiu, D. and Agrawal, G.: Time and Cost Sensitive Data-Intensive Computing on Hybrid Clouds, *Proc. 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid 2012)*, pp.636–643 (2012).
- [4] Gartner Says Cloud Consumers Need Brokerages to Unlock the Potential of Cloud Services, Gartner, Inc. (online), available from <http://www.gartner.com/it/page.jsp?id=1064712> (accessed 2012-10-20).
- [5] myCloud Private and Hybrid Cloud, RightScale, Inc. (online), available from <http://www.rightscale.com/products/mycloud.php> (accessed 2012-10-20).
- [6] Baker, M., Buyya, R. and Laforenza, D.: Grids and Grid technologies for wide-area distributed computing, *Software: Practice and Experience*, Vol.32, No.15, pp.1437–1466 (2002).
- [7] Foster, I. and Kesselman, C.: *The Grid: Blueprint for a New Computing Infrastructure*, Morgan Kaufmann (1998).
- [8] Fitzgerald, S., Foster, I. and Kesselman, C.: Grid information services for distributed resource sharing, *Proc. 10th International Symposium on High-Performance Parallel and Distributed Computing (HPDC-10)*, pp.181–194 (2001).
- [9] Lacour, S., Perez, C. and Priol, T.: Generic Application Description Model: Toward Automatic Deployment of Applications on Computational Grids, *Proc. 6th IEEE/ACM International Workshop on Grid Computing (GRID 05)*, pp.284–287 (2005).
- [10] Abramson, D., Buyya, R. and Giddy, J.: A computational economy for grid computing and its implementation in the Nimrod-G resource broker, *Future Generation Computer Systems*, Vol.18, No.8, pp.1061–1074 (2002).
- [11] Aloisio, G. and Cafaro, M.: Web-based access to the Grid using the Grid Resource Broker Portal, *Concurrency and Computation: Practice and Experience*, Grid Computing environments Special, Vol.14, No.13-15, pp.1145–1160 (2002).
- [12] Foster, I., Zhao, Y., Raicu, I. and Lu, S.: Cloud Computing and Grid Computing 360-Degree Compared, *Proc. Grid Computing Environments Workshop 2008 (GCE 08)*, pp.1–10 (2008).
- [13] Auto Scaling, Amazon Web Services LLC (online), available from <http://aws.amazon.com/autoscaling/> (accessed 2012-10-20).
- [14] Eicken, T.: RightScale Server Orchestration and Amazon SWF launch, RightScale, Inc. (online), available from <http://blog.rightscale.com/2012/02/22/rightscale-server-orchestration-and-amazon-swf-launch/> (accessed 2012-10-20).
- [15] Strebel, J. and Stage, A.: An economic decision model for business software application deployment on hybrid Cloud environments, *Proc. Multikonferenz Wirtschaftsinformatik 2010 (MKWI 2010)*, pp.195–206 (2010).
- [16] Mazhelis, O. and Tyrvaïnen, P.: Role of Data Communications in Hybrid Cloud Costs, *Proc. 37th EUROMICRO Conference on Software Engineering and Advanced Applications (SEAA 2011)*, pp.138–145 (2011).
- [17] Wood, T., Shenoy, P., Ramakrishnan, K.K. and Merwe, J.: CloudNet: Dynamic pooling of cloud resources by live WAN migration of virtual machines, *Proc. 7th ACM SIGPLAN/SIGOPS International Conference on Virtual Execution Environments (VEE '11)*, pp.121–132 (2011).
- [18] Subashini, S. and Kavitha, V.: A survey on security issues in service delivery models of cloud computing, *Journal of Network and Computer Applications*, Vol.34, No.1, pp.1–11 (2011).
- [19] Singh, A., Korupolu, M. and Mohapatra, D.: Server-Storage Virtualization: Integration and Load Balancing in Data Centers, *Proc. ACM/IEEE Conference on Supercomputing (SC08)*, pp.1–12 (2008).
- [20] Ra, E.: Operating Systems and Benchmarks, StorageReview.com (online), available from <http://www.storage-review.com/articles/200003/20000313OSandBM.5.html> (accessed 2012-05-19).
- [21] Zhang, H., Jiang, G., Yoshihira, K., Chen, H. and Saxena, A.: Intelligent Workload Factoring for A Hybrid Cloud Computing Model, *Proc. 2009 IEEE Congress on Services (SERVICES 2009)*, pp.701–708 (2009).
- [22] Gong, Z., Gu, X. and Wilkes, J.: PRESS – PRedictive Elastic ReSource Scaling for cloud systems, *Proc. 6th International Conference on Network and Service Management (CNSM 2010)*, pp.9–16 (2010).
- [23] Kondo, D., Javadi, B., Malecot, P., Cappello, F. and Anderson, P.D.: Cost-benefit analysis of Cloud Computing versus desktop grids, *Proc. 23rd IEEE International Symposium on Parallel and Distributed Processing (IPDPS 2009)*, pp.1–12 (2009).
- [24] Martens, B., Walterbusch, M. and Teuteberg, F.: Costing of Cloud Computing Services: A Total Cost of Ownership Approach, *Proc. 45th Hawaii International Conference on System Sciences (HICSS 45)*, pp.1563–1572 (2012).
- [25] Public vs Private Cloud Cost Comparison, Xuropa Inc. (online), available from <http://www.xuropa.com/blog/2011/03/21/public-vs-cloud-costs/> (accessed 2012-10-20).



江丸 裕教 (正会員)

平成8年北海道大学工学部情報工学科卒業。平成10年同大学大学院工学研究科修士課程修了。同年株式会社日立製作所システム開発研究所(現、横浜研究所)入社。平成22年10月日立アメリカに出向。現在に至る。システム

運用管理の研究に従事。



高井 昌彰 (正会員)

昭和 58 年東北大学工学部電子工学科卒業。昭和 63 年同大学大学院工学研究科博士課程修了。工学博士。同年東京大学理学部助手。平成元年北海道大学工学部講師。平成 4 年同助教授。

平成 7 年同大学大型計算機センター助教授。平成 15 年同大学情報基盤センター教授。平成 16 年同大学情報基盤センター副センター長。平成 18 年同大学 CIO 補佐官（役員補佐相当職）。平成 23 年同大学情報基盤センター長。現在に至る。超並列・分散処理システム，コンピュータグラフィックス，コンピュータネットワークの研究に従事。平成 17 年 NPO 法人北海道地域ネットワーク協議会理事・副会長。平成 23 年情報処理学会北海道支部長。電子情報通信学会，IEEE，国際 CIO 学会各会員。