

OpenStack環境でのオーケストレーション 定義を容易にするGUIエディタの実現

情報学群 コンピュータサイエンス専攻
(分散処理OS研究室)
1160304 川口 貴大
2016年2月15日

研究背景(1/4)

近年クラウドが普及し、今後IaaSの需要が増加

クラウドとは

- ◆ ネットワーク上の資源をネット経由で使用する

IaaSとは

- ◆ クラウドコンピューティングであり、CPUやメモリ、ストレージ等のITインフラをネットワーク経由で提供するサービス

現状

- ◆ 複数セグメントを持つシステム構成には時間がかかる

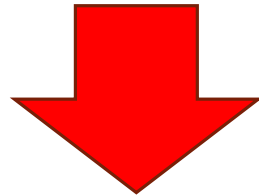


複数セグメントを持つシステム構成の構築には作業の効率化が求められている

研究背景(2/4)

OpenStack

- ◆ IaaS基盤構築ソフトウェア
 - インスタンス(仮想マシン)や仮想ネットワーク, 仮想ルーターを定義してシステムを構築
 - これらのリソースはOpenStack内のコンポーネント群により制御・管理されている
- ◆ Heatはそれらのコンポーネント群を制御, 一括で定義し構築する



オーケストレーション

研究背景(3/4)

Heat

- ◆仮想環境構築をオーケストレーション(自動化)
- ◆テンプレートファイルに構築内容を記述

問題点

- ◆テキスト記述量が膨大
- ◆構成情報を把握しづらい
- ◆書式が複雑

テンプレートファイル例

```
parameters:
  ext-net:
    type: string
    description: ext-netID
    default: ext-net

resources:
  private1:
    type: OS::Neutron::Net
    properties:
      name: demo-net1

  private1-sub:
    type: OS::Neutron::Subnet
    depends_on: private1
    properties:
      network_id: [get_resource: private1]
      cidr: 192.168.2.0/24

  ext-router1:
    type: OS::Neutron::Router
    properties:
      external_gateway_info:
        network: [get_param: ext-net]

  ext-router1-interface:
    type: OS::Neutron::RouterInterface
    depends_on: [ext-router1, private1-sub]
    properties:
      router_id: [get_resource: ext-router1]
      subnet_id: [get_resource: private1-sub]

  private2:
    type: OS::Neutron::Net
    properties:
      name: demo-net2

  private2-sub:
    type: OS::Neutron::Subnet
    depends_on: private2
    properties:
      network_id: [get_resource: private2]
      cidr: 192.168.3.0/24

  ext-router2:
    type: OS::Neutron::Router
    properties:
      external_gateway_info:
        network: [get_param: ext-net]

  ext-router2-interface:
    type: OS::Neutron::RouterInterface
    depends_on: [ext-router2, private2-sub]
    properties:
      router_id: [get_resource: ext-router2]

  private3:
    type: OS::Neutron::Net
    properties:
      name: demo-net3

  private3-sub:
    type: OS::Neutron::Subnet
    depends_on: private3
    properties:
      network_id: [get_resource: private3]
      cidr: 192.168.4.0/24

  ext-router3:
    type: OS::Neutron::Router
    properties:
      external_gateway_info:
        network: [get_param: ext-net]

  ext-router3-interface:
    type: OS::Neutron::RouterInterface
    depends_on: [ext-router3, private3-sub]
    properties:
      router_id: [get_resource: ext-router3]
      subnet_id: [get_resource: private3-sub]

  instance1:
    type: OS::Nova::Server
    depends_on: private1-sub
    properties:
      name: test1
      image: centos-test
      flavor: ml.small
      networks:
        - network: [get_resource: private1]

  instance2:
    type: OS::Nova::Server
    depends_on: private3-sub
    properties:
      name: test2
      image: cirros-0.3.3-x86_64
      flavor: ml.medium
      networks:
        - network: [get_resource: private3]

  instance3:
    type: OS::Nova::Server
    depends_on: private2-sub
    properties:
      name: test3
      image: cirros-0.3.3-i386
      flavor: ml.small
      networks:
        - network: [get_resource: private2]

  instance4:
    type: OS::Nova::Server
    depends_on: private2-sub
    properties:
      name: test4
      image: centos-test
      flavor: ml.large
      networks:
        - network: [get_resource: private2]

  instance5:
    type: OS::Nova::Server
    depends_on: private3-sub
    properties:
      name: test5
      image: cirros-0.3.3-i386
      flavor: ml.small
      networks:
        - network: [get_resource: private3]
```

そこで

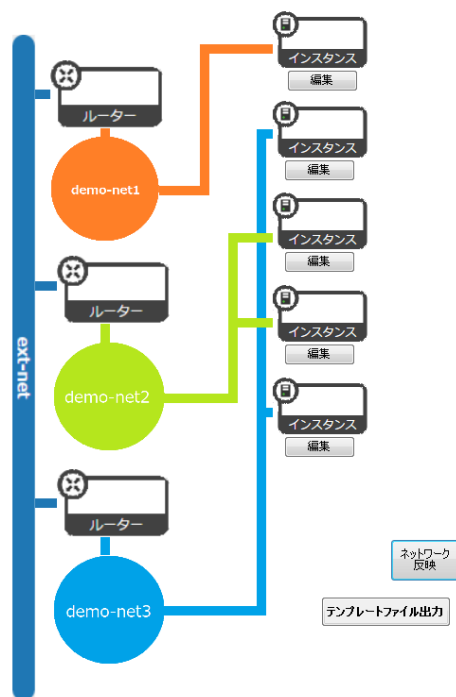
GUIベースで構成内容を容易に把握でき、
テキスト入力を極力撤廃した
オーケストレーション定義エディタを提案

オーケストレーション定義エディタの概要(1/3)

オーケストレーション定義エディタ

- ◆ 構成確認画面でシステム構成を可視化
- ◆ 詳細設定画面で入力項目をプルダウンメニューから選択する

構成確認画面



詳細設定画面

リソース
インスタンス設定

インスタンス名	<input type="text" value="test"/>
ディスクイメージ	<input type="text" value="centos-test"/>
フレーバー	<input type="text" value="m1.small"/>
接続先	<input type="text" value="demo-net1"/>

決定

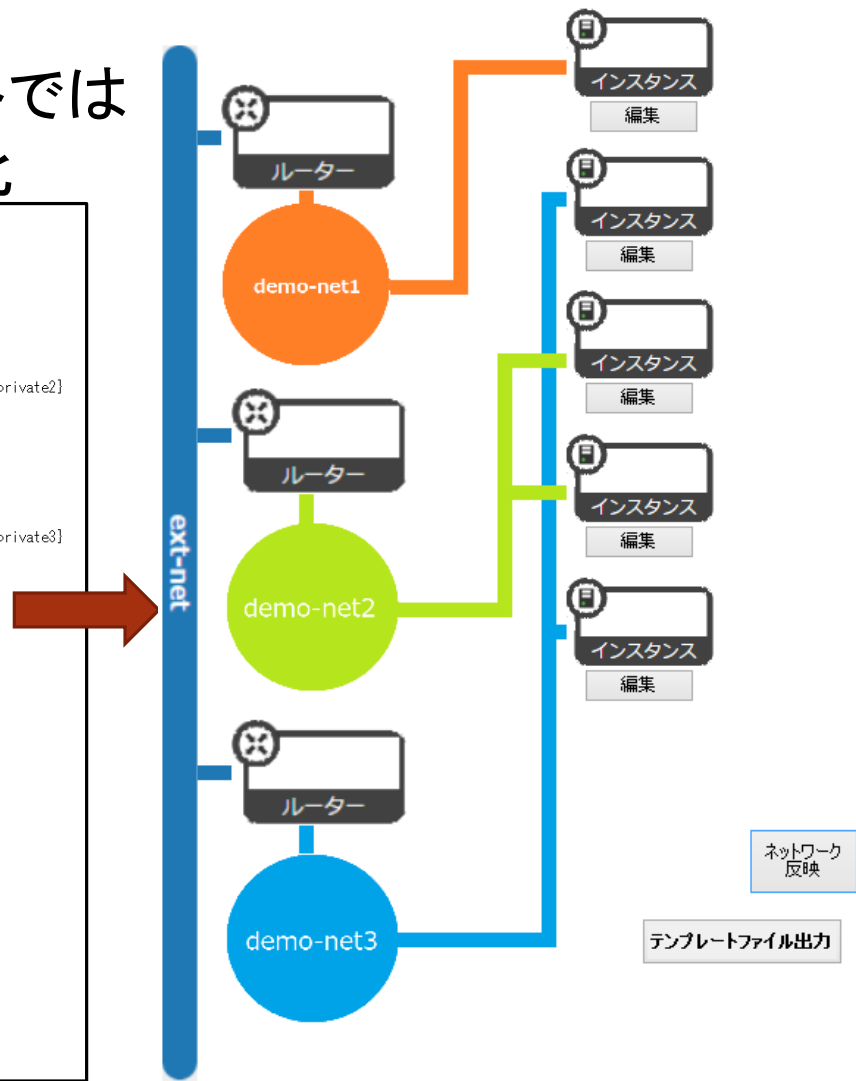
オーケストレーション定義エディタの概要(2/3)

◆システム構成の可視化

➤ GUIベースにすることでテキストではわかりづらかった構成を可視化

<pre>parameters: ext-net: type: string description: ext-netID default: ext-net resources: private1: type: OS::Neutron::Net properties: name: demo-net1 private1-sub: type: OS::Neutron::Subnet depends_on: private1 properties: network_id: [get_resource: private1] cidr: 192.168.2.0/24 ext-router1: type: OS::Neutron::Router properties: external_gateway_info: network: [get_param: ext-net] ext-router1-interface: type: OS::Neutron::RouterInterface depends_on: [ext-router1, private1-sub] properties: router_id: [get_resource: ext-router1] subnet_id: [get_resource: private1-sub] private2: type: OS::Neutron::Net properties: name: demo-net2 private2-sub: type: OS::Neutron::Subnet depends_on: private2 properties: network_id: [get_resource: private2] cidr: 192.168.3.0/24 ext-router2: type: OS::Neutron::Router properties: external_gateway_info: network: [get_param: ext-net] ext-router2-interface: type: OS::Neutron::RouterInterface depends_on: [ext-router2, private2-sub] properties: router_id: [get_resource: ext-router2]</pre>	<pre>private3: type: OS::Neutron::Net properties: name: demo-net3 private3-sub: type: OS::Neutron::Subnet depends_on: private3 properties: network_id: [get_resource: private3] cidr: 192.168.4.0/24 ext-router3: type: OS::Neutron::Router properties: external_gateway_info: network: [get_param: ext-net] ext-router3-interface: type: OS::Neutron::RouterInterface depends_on: [ext-router3, private3-sub] properties: router_id: [get_resource: ext-router3] subnet_id: [get_resource: private3-sub] instance1: type: OS::Nova::Server depends_on: private1-sub properties: name: test1 image: centos-test flavor: ml.small networks: - network: [get_resource: private1] instance2: type: OS::Nova::Server depends_on: private3-sub properties: name: test2 image: cirros-0.3.3-x86_64 flavor: ml.medium networks: - network: [get_resource: private3] instance3: type: OS::Nova::Server depends_on: private2-sub properties: name: test3 image: cirros-0.3.3-i386 flavor: ml.small networks: - network: [get_resource: private2]</pre>	<pre>instance4: type: OS::Nova::Server depends_on: private2-sub properties: name: test4 image: centos-test flavor: ml.large networks: - network: [get_resource: private2] instance5: type: OS::Nova::Server depends_on: private3-sub properties: name: test5 image: cirros-0.3.3-i386 flavor: ml.small networks: - network: [get_resource: private3]</pre>
---	---	---

テンプレートファイル



オーケストレーション定義エディタ

オーケストレーション定義エディタの概要(3/3)

◆テキスト手動入力を極力撤廃

- プルダウンメニューによりクリックだけで入力可能
- 入力内容が決まっているものは自動入力

テンプレートファイル

```
instance2:  
  type: OS::Nova::Server  
  depends_on: private3-sub  
  properties:  
    name: "test2"  
    image: cirros-0.3.3-x86_64  
    flavor: m1.medium  
  networks:  
    - network: [get_resource: private3]
```



オーケストレーション定義エディタ

リソース

インスタンス設定

インスタンス名	<input type="text"/>
ディスクイメージ	<input type="text" value="centos-test"/>
フレーバー	<input type="text" value="centos-test"/>
接続先	<input type="text" value="cirros-0.3.3-x86_64"/>

決定

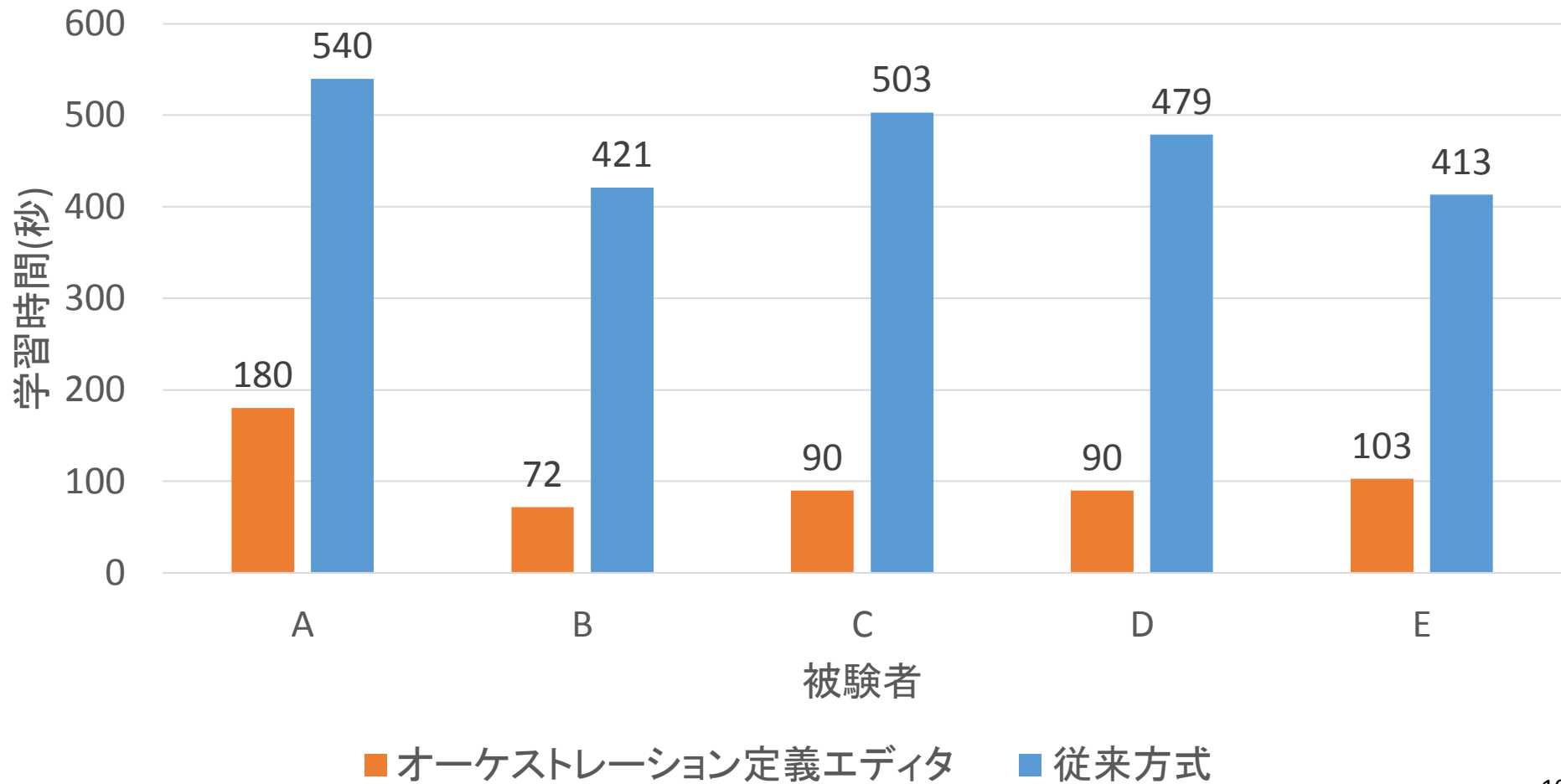
評価(1/3)

被験者の学習時間, テンプレートファイル作成所要時間を記録

実験内容	従来方式とオーケストレーション定義エディタそれぞれで同じシステム構成を構築
計測項目	学習時間, 作成所要時間
被験者	OpenStackに関する基本的な知識を有した学生5名
システム構成	(Ⅰ)1セグメント構成(インスタンス数5台) (Ⅱ)2セグメント構成(インスタンス数5台) (Ⅲ)3セグメント構成(インスタンス数5台)

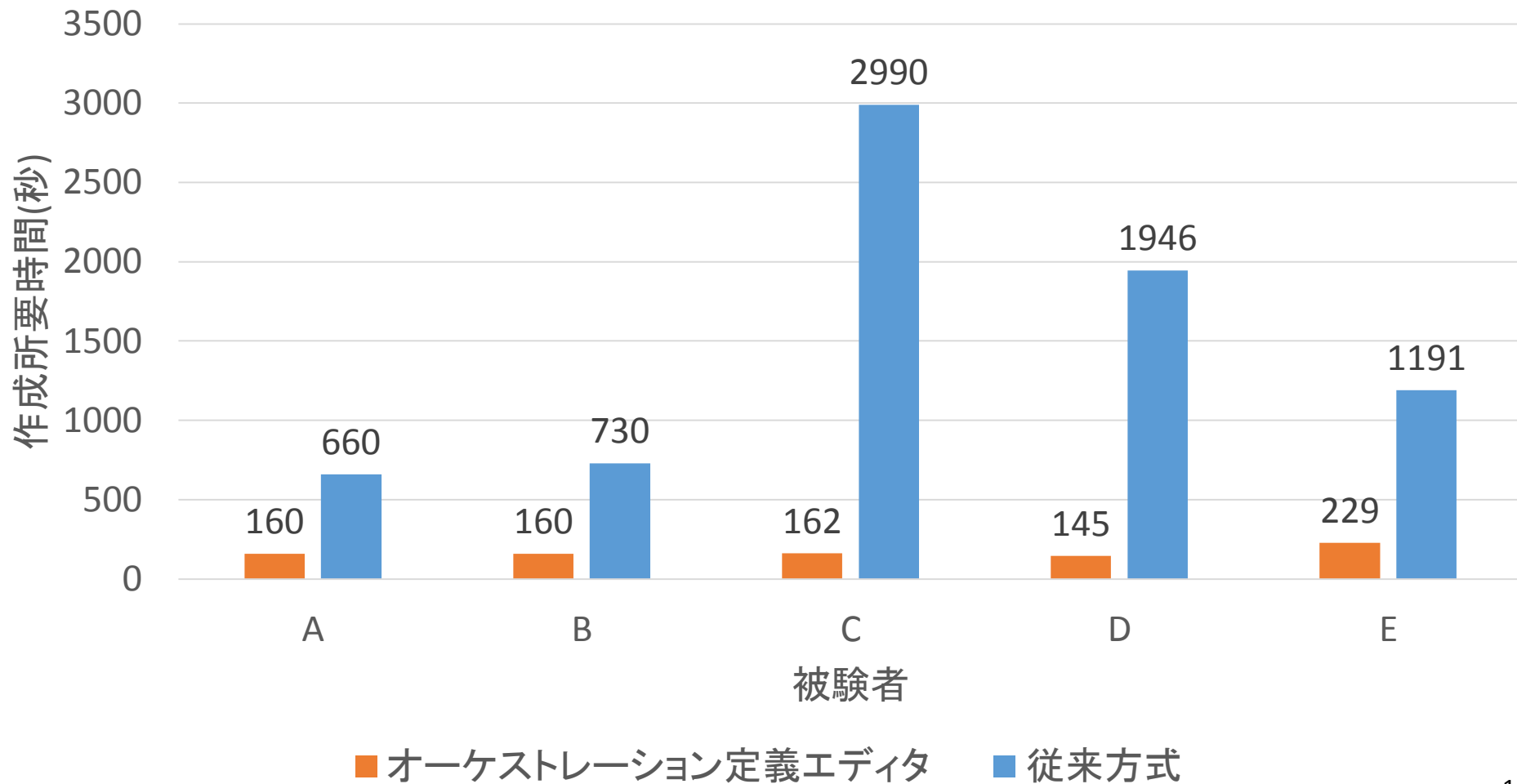
学習時間比較

各方式学習時間比較



作成所要時間比較

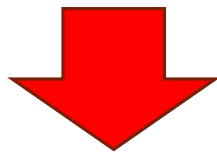
3セグメント構成作成所要時間比較



まとめ

オーケストレーション定義エディタ

- ◆ Heat専門知識を意識することなくテンプレートファイルを作成可能なオーケストレーション定義エディタを実現



- 従来の方式よりも少ない学習時間で容易に短時間かつ正確にテンプレートファイルを作成可能

ご静聴ありがとうございました。