



LiON: ネットワークトポロジを考慮した 実験用仮想ネットワーク自動構築機構 (1/2)

慶應義塾大学 林 和輝, 渡邊 大記, 近藤 賢郎, 寺岡 文男 [gordon, nelio, latte, tera}@inl.ics.keio.ac.jp

❖ ヘテロジニアス・コンピューティングの需要

- ✓ 計算資源の多様化, 高速 IO 技術の多様化

CPU / GPU, DPDK / Netmap

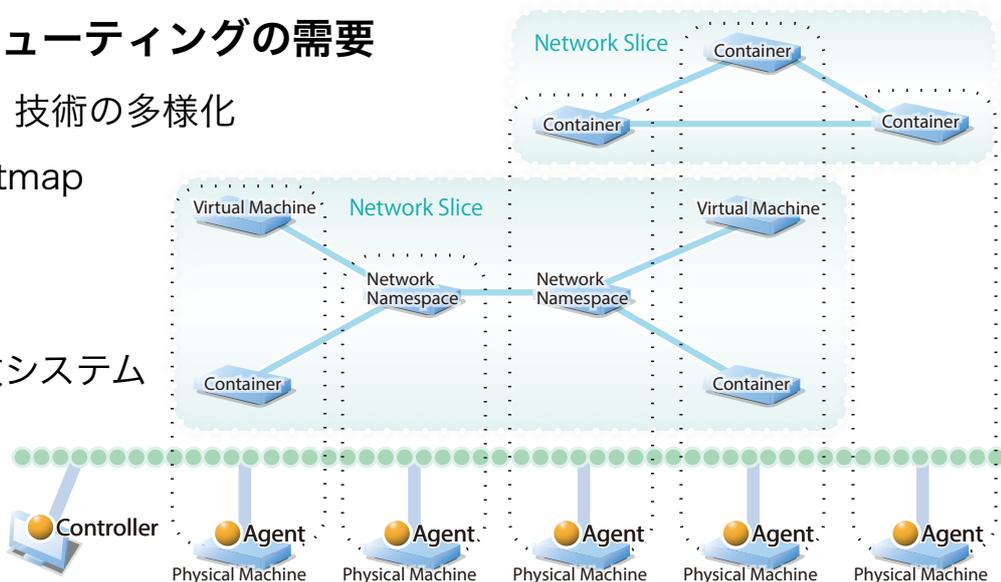
- ✓ 仮想化技術の多様化

KVM, Container, NetNS

- ✓ ネットワークを介した分散システム

遅延, ジッタ, 帯域など

多様な物理資源を統括する
ネットワーク管理機構が必要



❖ LiON: Lightweight On-demand Networking

- ✓ 軽量かつ手軽な仮想ネットワーク構築ツール

コマンド入力で簡単に実験環境を構築, 破棄可能

- ✓ 多様な仮想化技術に対応

KVM, Docker, NetNS による仮想ノードを生成

- ✓ **EVPN / VXLAN** によるオーバーレイネットワーク

ネットワーク機器を意識しない簡潔な設定

- ✓ **IaC (Infrastructure as Code)** に基づく環境構築

JSON ファイルにネットワークの構成情報を記述

```

"p_nodes": [
  {
    "pn_id": 1,
    "label": "gw01",
    "ip": "172.20.1.1"
  },
  {
    "pn_id": 2,
    "label": "gw02",
    "ip": "172.20.1.2"
  },
  ...
]

"v_nodes": [
  {
    "vn_id": 1,
    "label": "node-1",
    "pn_id": 1,
    "resource": [
      { "type": "vm" }
    ]
  },
  {
    "vn_id": 2,
    "label": "node-2",
    "pn_id": 2,
    "resource": [
      { "type": "docker" }
    ]
  },
  ...
]

```

実験者がネットワークトポロジから直感的に仮想環境を構築可能

```

"networks": [
  {
    "net_id": 1,
    "label": "vnet-1",
    "v_nodes": [
      {
        "vn_id": 1,
        "ip": "10.0.1.1/24"
      },
      {
        "vn_id": 2,
        "ip": "10.0.1.2/24"
      }
    ]
  },
  ...
]

```

❖ 関連研究・技術との比較

比較項目	LiON	DCE	Mininet	Kubernetes	Ansible	Chef
トポロジの考慮	◎	◎	◎			
アンダーレイの考慮	◎			○	◎	◎
低導入コスト	◎	○	◎		○	
スケーラビリティ	◎		○	◎	◎	◎
実アプリの動作	◎	○	○	○	◎	◎
仮想化の多様性	◎				○	○

Network Slice



LiON: ネットワークトポロジを考慮した 実験用仮想ネットワーク自動構築機構 (2/2)

慶應義塾大学 林 和輝, 渡邊 大記, 近藤 賢郎, 寺岡 文男 {gordon, nelio, latte, tera}@inl.ics.keio.ac.jp

❖ LiON による仮想ネットワーク構築の流れ

- ✓ 利用者が JSON 形式で構成定義ファイルを記述
物理 - 仮想ノードの対応関係, 仮想ノード間の接続情報

- ✓ Controller にアクセスしてコマンドを入力

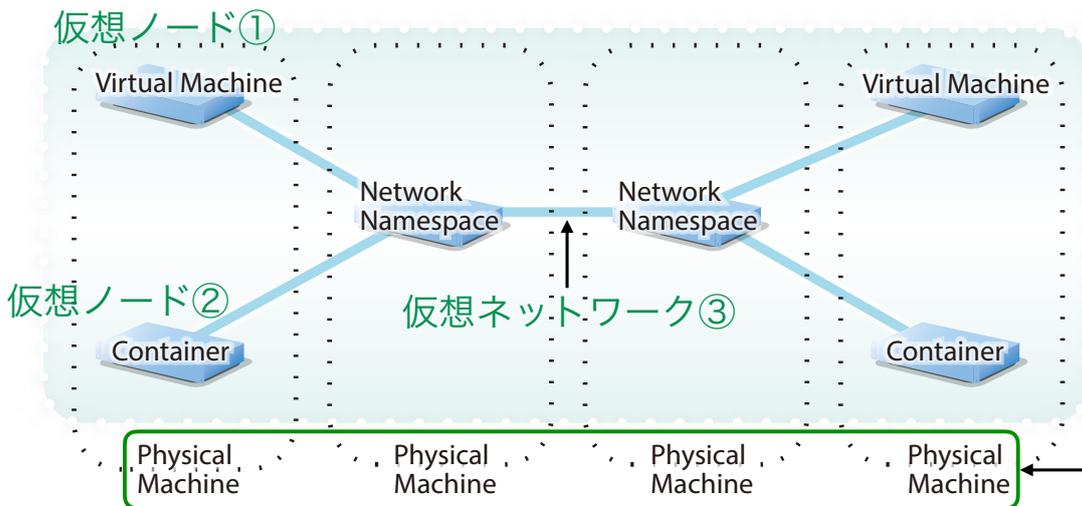
Controller: 各 Agent に必要な情報を送信

Agent: 仮想ノード (KVM, Docker, NetNS) 作成, VXLAN ネットワーク構築

```
controller# create vn1 setting.json
...
controller# list
vn1:
- node-1
  - id: 1
  - p_node: gw01 (172.20.1.1)
- node-2
  - id: 2
  - p_node: gw02 (172.20.1.2)
```

❖ デモ動画で構築した仮想ネットワーク

- ✓ 6 つの仮想ノードからなるダンベル型トポロジ



```
"p_nodes": [
  {
    "pn_id": 1,
    "label": "gw01",
    "ip": "172.20.1.1"
  },
  {
    "pn_id": 2,
    "label": "gw02",
    "ip": "172.20.1.2"
  },
  {
    "pn_id": 3,
    "label": "gw03",
    "ip": "172.20.1.3"
  },
  {
    "pn_id": 4,
    "label": "gw04",
    "ip": "172.20.1.4"
  }
]
```

← 4台の物理マシンの記述

```
"v_nodes": [
  {
    "vn_id": 1,
    "label": "node-1",
    "pn_id": 1,
    "resource": {
      "type": "vm"
    }
  },
  {
    "vn_id": 2,
    "label": "node-2",
    "pn_id": 1,
    "resource": {
      "type": "docker"
    }
  }
]
```

仮想ノードの記述 (①, ②)

仮想ノードを配置する
物理マシンのIDを記述

サブネットに属する
仮想ノード群を記述

仮想ノードの種類を選択
netns / docker / vm

```
"networks": [
  ...
  {
    "net_id": 3,
    "label": "vnet-3"
    "v_nodes": [
      {
        "vn_id": 3,
        "ip": "10.0.3.1/24"
      },
      {
        "vn_id": 4,
        "ip": "10.0.3.2/24"
      }
    ]
  },
  ...
]
```

仮想ネットワークの記述 (③)