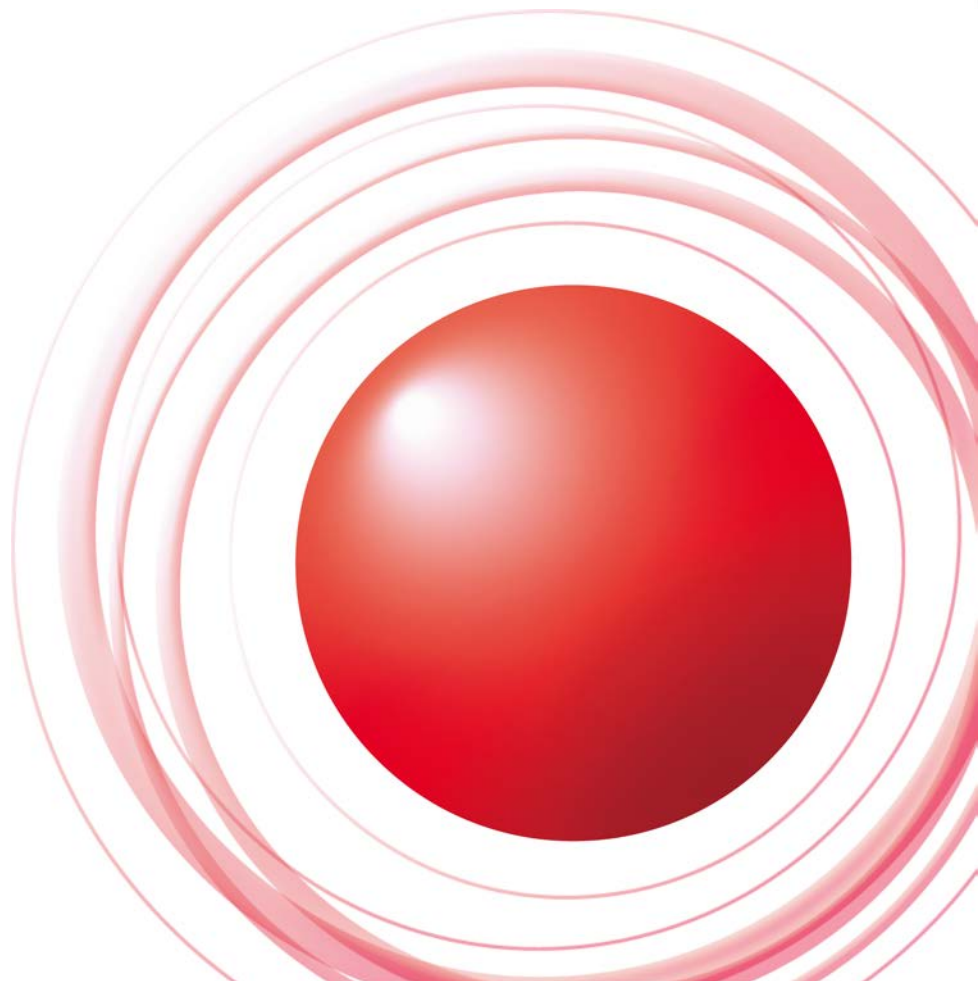


IIJ GIO ホスティングパッケージサービス API デモ

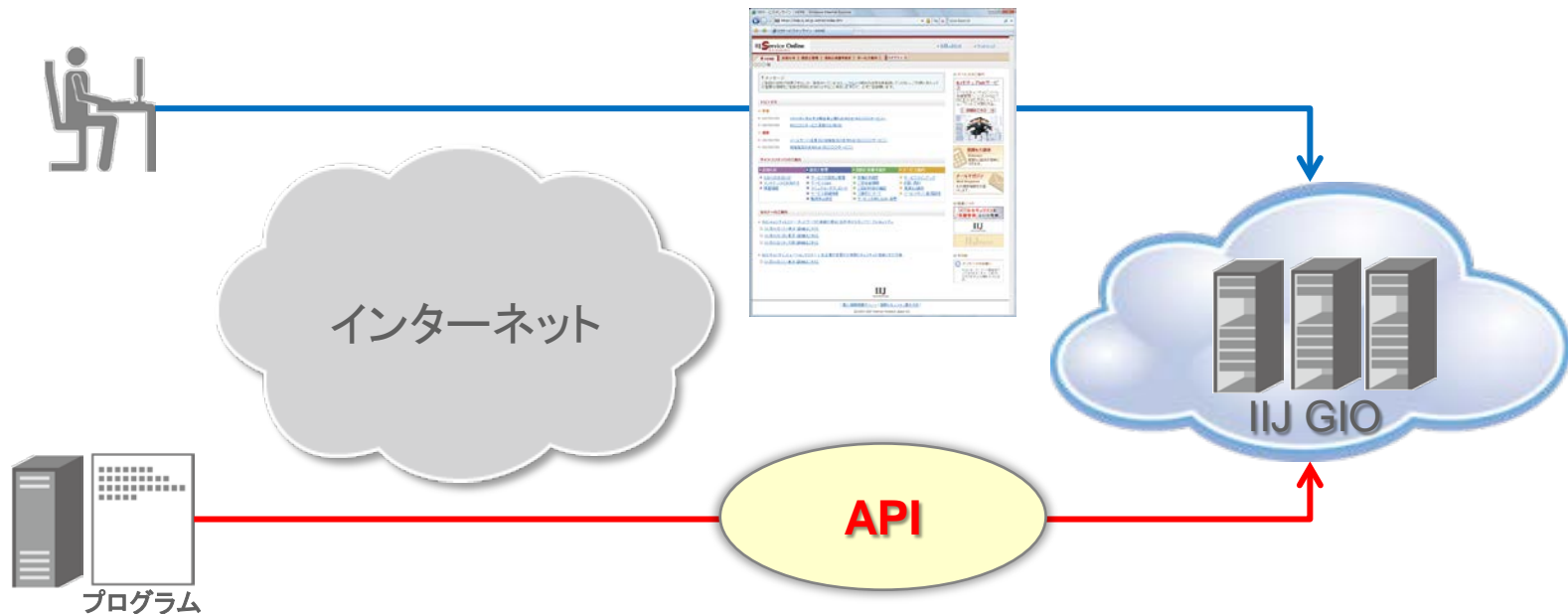


株式会社インターネットイニシアティブ
樋村 隆弘

Ongoing Innovation

API とは

- API (Application Programming Interface)
 - 提供されているサービスをプログラムを通じて呼び出すためのインターフェース
- API 公開の目的
 - 定型操作を**独自のプログラム**を作って自動化したいというニーズに対する**手段の提供**



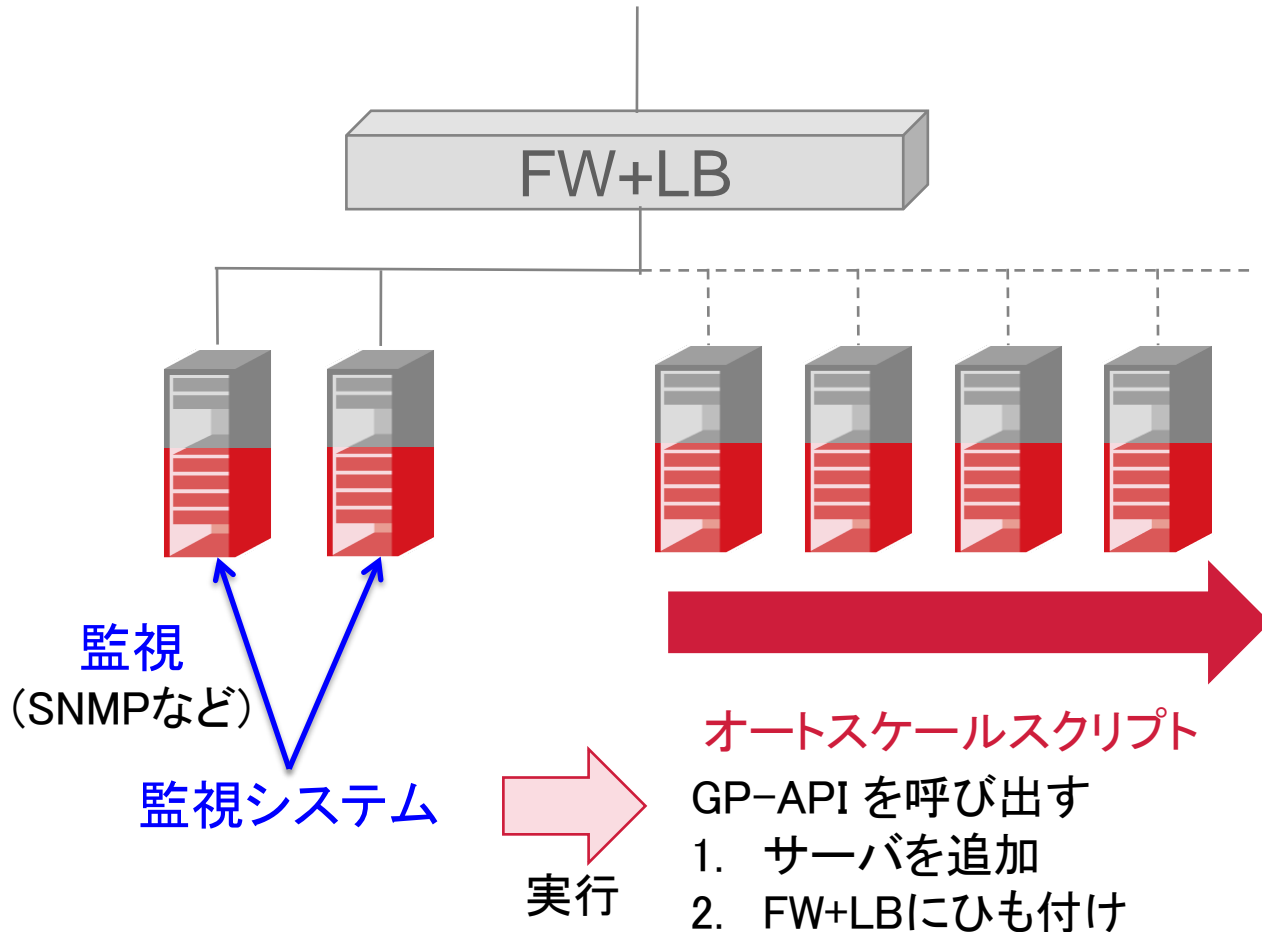
IIJ ホスティングパッケージサービス API (GP-API)

- API プラットフォーム利用の第一弾としてリリース
 - 2013年9月24日 リリース
 - 2013年11月7日 FW+LB オプション関連機能追加
- IIJ GIO ホスティングパッケージサービス (GP) の仮想サーバや FW+LB オプションなどの契約・操作が可能

- API プラットフォーム自体の仕組みは「GIO-API を支える API プラットフォーム」
川上芳尚, IIJ Technical WEEK 2013, 11/19

活用例1: GP-API を利用したオートスケール

仮想サーバの負荷を監視・高負荷の場合自動でサーバを追加



活用例1: GP-API を利用したオートスケール

仮想サーバの追加 (クローニング)

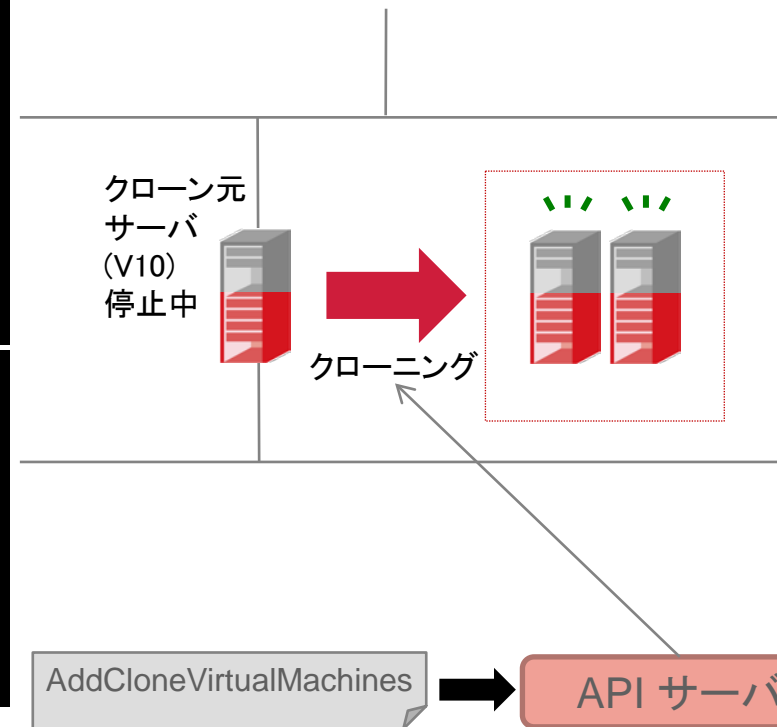
1. テンプレートサーバからクローニングにより
仮想サーバを作成 (AddCloneVirtualMachines)

```
https://gp.api.ij.jp/json?APIVersion=20130901
&AccessKeyId=#####
&Action=AddCloneVirtualMachines
&GpServiceCode=gp#####
&SrcGcServiceCode=gc#####
&ContractNum=2
&Expire=2013-10-30T06%3A11%3A00Z
&Signature=#####
&SignatureMethod=HmacSHA256
&SignatureVersion=2
```

リクエスト

```
{
  "AddCloneVirtualMachinesResponse": {
    "RequestId": "xxxxxxxx-xxxxxxxx-xxxxxxxx",
    "GpServiceCode": "gp#####",
    "SrcGcServiceCode": "gc#####",
    "GcServiceCodeList": [ "gc#####1", "gc#####2" ]
  }
}
```

レスポンス



2. 作成完了まち
契約状態取得 API でのポーリング
(GetContractStatus)

非同期処理

完了まで時間がかかる処理

- 仮想サーバ追加申込 (AddVirtualMachines)
- 仮想サーバ起動 (StartVirtualMachine)

- 契約追加などの API レスポンスは即座に返される
- 期待する状態になるまで
状態取得系の API を定期的に実行
 - 契約状態取得 (GetContractStatus)
 - 仮想サーバ状態取得 (GetVirtualMachineStatus)

= ポーリング

```
{  
  "GetContractStatusResponse": {  
    "RequestId": "xxxxxx-xxxxxx",  
    "GcServiceCode": "gcYYYYYYYYYY",  
    "Status": "InPreparation"  
  }  
}
```



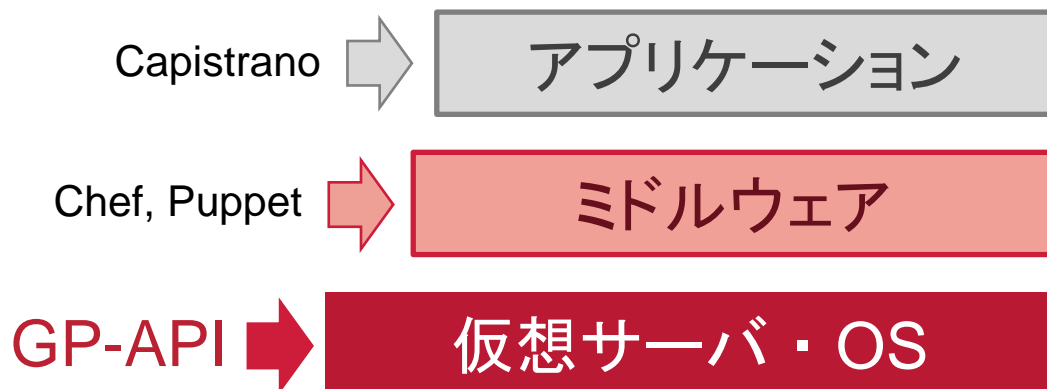
```
{  
  "GetContractStatusResponse": {  
    "RequestId": "xxxxxx-xxxxxx",  
    "GcServiceCode": "gcYYYYYYYYYY",  
    "Status": "InService"  
  }  
}
```


活用例2: 再現可能なテスト

Infrastructure as Code

「再現可能なテスト」のために
→ 再現可能な環境構築

- アプリケーションデプロイの自動化 (Capistrano など)
- 環境構築の自動化 ([Chef](#), Puppet, Ansible など)
 ↖ CentOS6 ベーシックプランに導入済み
- 仮想サーバ構築の自動化
 → GP-API により可能に



活用例2: 再現可能なテスト

Infrastructure as Code

