クラウドのこころ

神戸大学

中村　匡秀

1.1 身の回りのクラウドサービス

■ 身近なところにある様々なサービス

■エンドユーザ向け

- Gmail
- Dropbox
- Google Drive
- Evernote
- Twitter
- Facebook
- YouTube
- Flickr
- はてなブックマーク

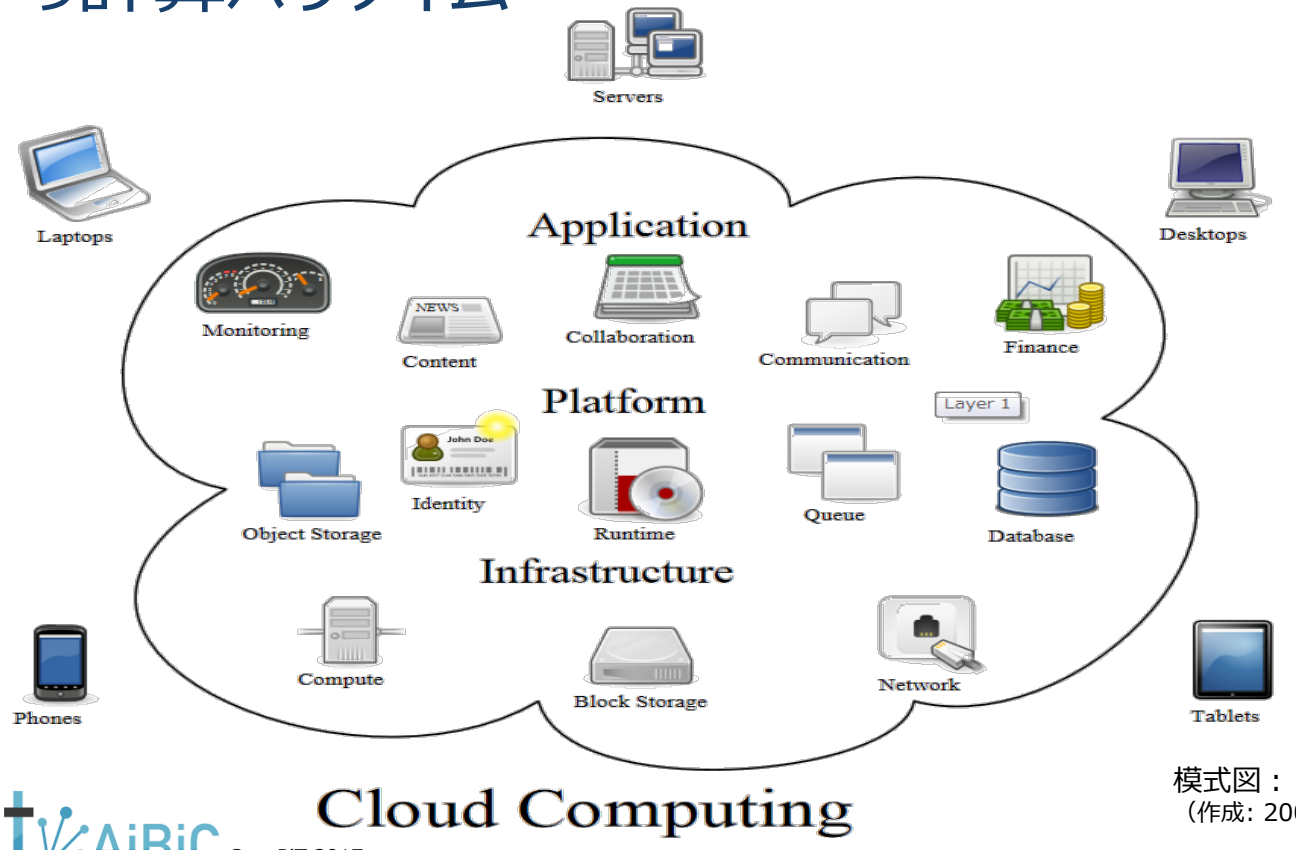
■ビジネス・開発向け

- MS-Office 365
- Wolfram Alpha
- Salesforce CRM
- Amazon EC2
- Amazon S3
- Google App Engine
- Google Compute Engine
- Heroku
- Orion Hub

1.2 クラウドコンピューティング

- ネットワークのどこかで管理され，必要に応じて利用する

- 様々な計算資源がネットワーク（のどこか）で管理され，ユーザは必要に応じてそれらを「サービス」として利用するという計算パラダイム



模式図： Wikipediaより
（作成: 2009年3月3日）

1.3 クラウドにおける計算資源

ICTにおいて利用される様々なモノ

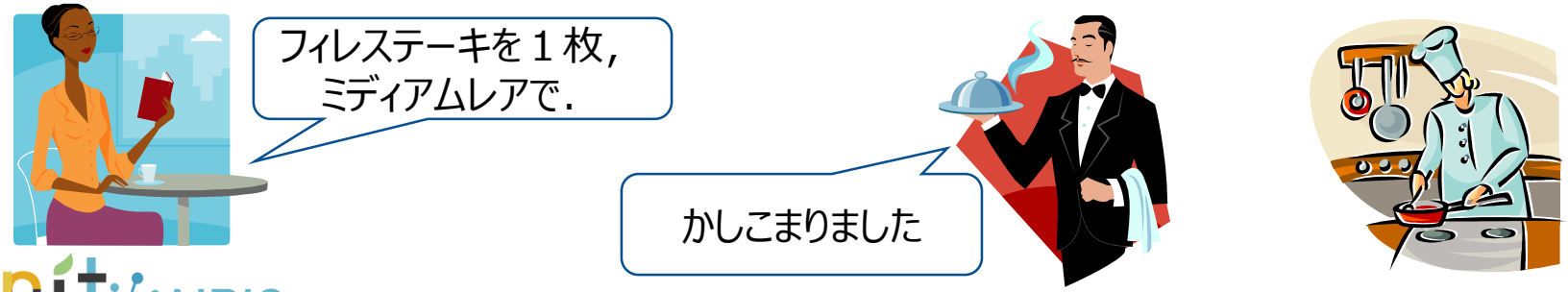
- 計算資源 (Computing Resources)
  - ICTにおける計算や処理，仕事に利用される様々なモノ
  - コンピュータ，ソフトウェア，デバイスなど広範囲に及ぶ
    - メール，カレンダー，スケジュール，営業ツール，オフィス，開発環境ソフト
    - データベース，Webサーバ，アプリケーションサーバ，ライブラリ
    - HDD, サーバ, デスクトップ，ルータ，スイッチ，ネットワーク，プリンタ

1.4 サービスとは

■ 提供者によって行われる，利用者にとって価値をもたらす処理

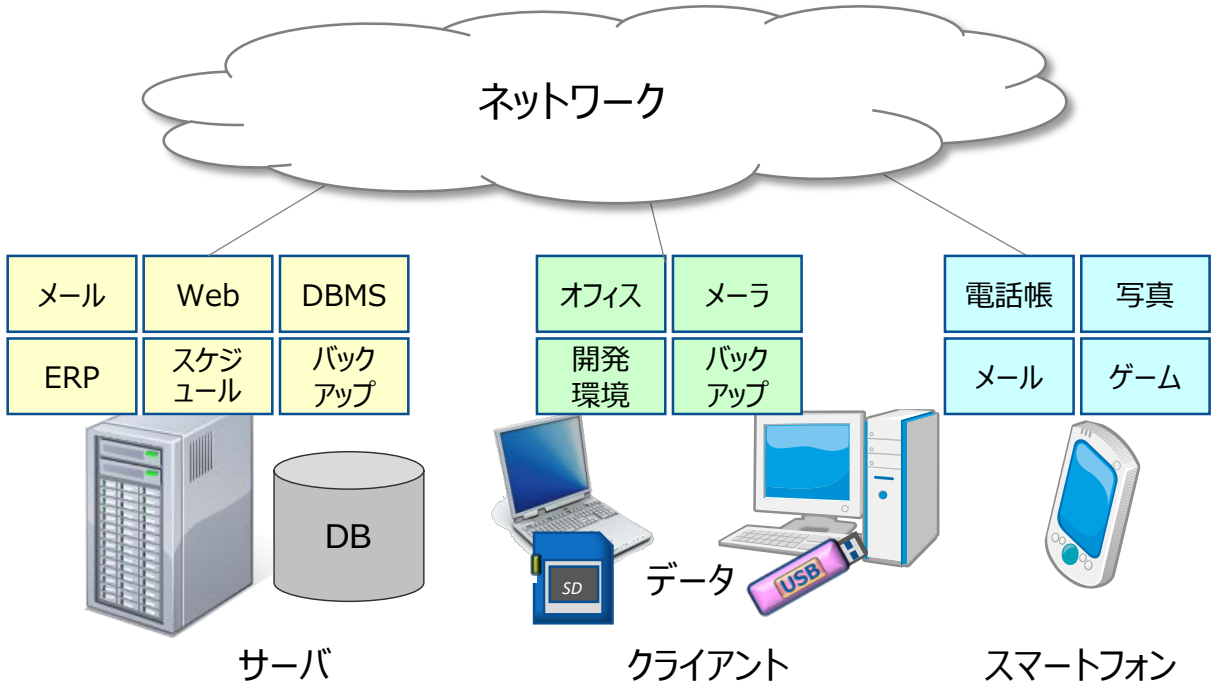
- 利用者 (Consumer)にとって価値をもたらす処理，計算，業務，仕事など．提供者 (Provider)によって行われる

- サービス (Service) の特徴
  - 利用方法が決まっており，それに従えば誰でも利用できる
  - 利用者は提供者が実施する仕事をあれこれ指示しないし手段を問わない
  - 良いサービス＝結果が求めたものに合っているか，それ以上であること
  - 対価を払って利用する．無料のものもある（≠ 購入して保有）
  - 買ってきて自分でやるのではなく，お金を払って人にやってもらう



1.5 従来のICT環境

従来のネットワークは情報を伝達するためのメディア

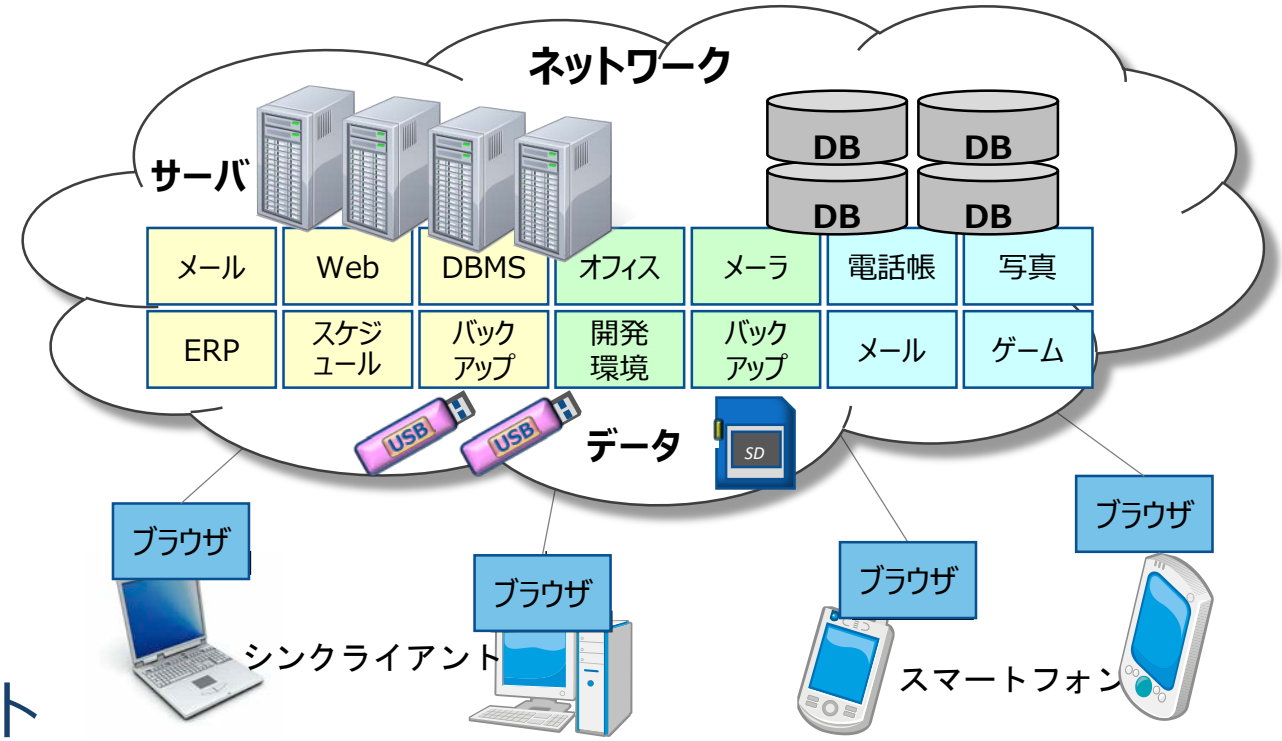


ポイント

- ネットワークは情報を伝達するためのメディア
- 業務や目的に応じて，ハードウェアを調達
- ソフトウェアは各ハードウェアにインストールされる
- データは各ハードウェアに分散して保存される

1.6 クラウドコンピューティング環境

■ ネットワークは多様な計算資源を収容する仮想的な巨大コンピュータ



ポイント

- ネットワークは多様な計算資源を収容する仮想的な巨大コンピュータ
- ハードウェア，ソフトウェア，データはクラウド内のサービスとして存在する
- 業務や目的に応じて，サービスを選択・利用．不要になればやめる
- 手元のコンピュータは巨大コンピュータにアクセスするための軽量な窓口

1.7 クラウドの中身は

- 汎用的なコンピュータを数千～数十万収容するデータセンターで実現

- 実際のデータセンター
  - さくらインターネット　石狩データセンター

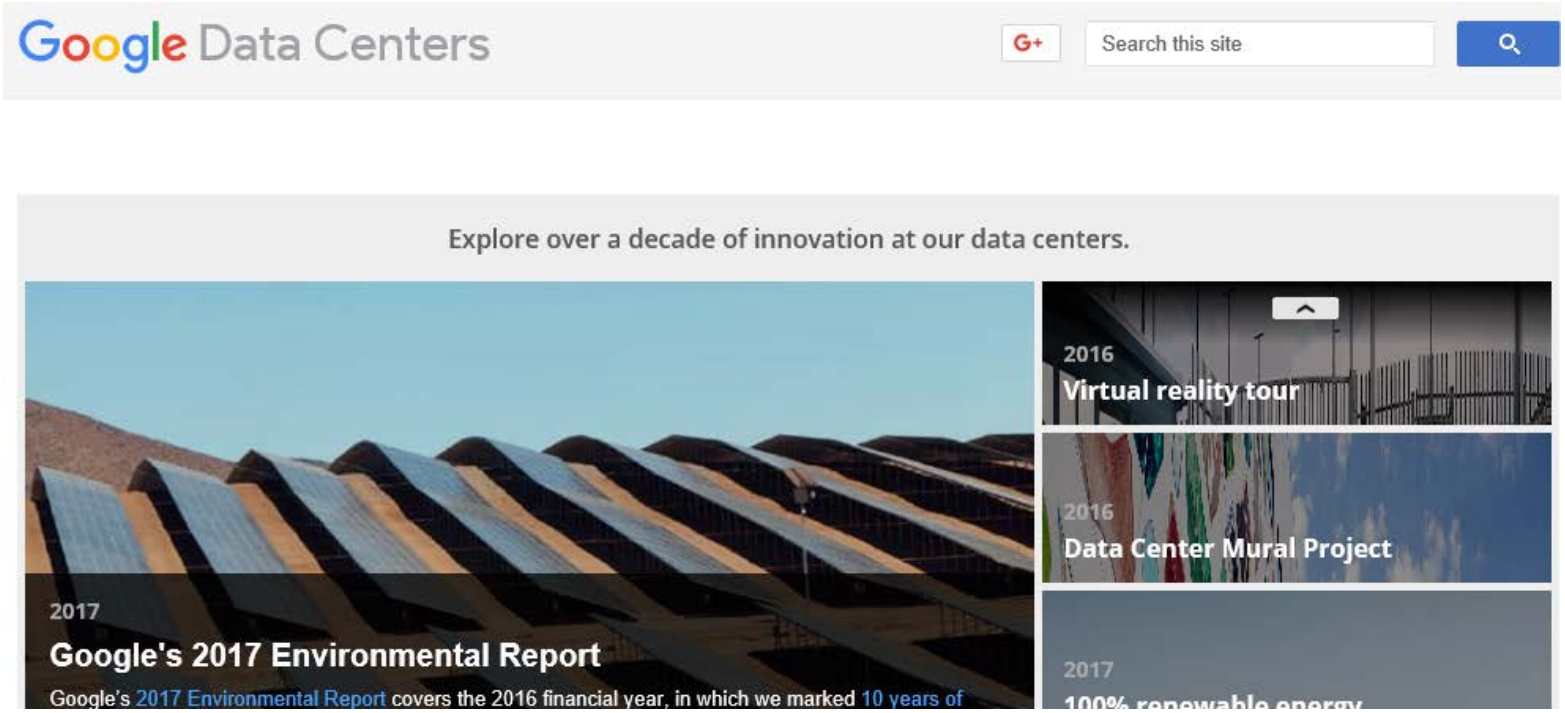
    http://ishikari.sakura.ad.jp/

1.7 クラウドの中身は

- 汎用的なコンピュータを数千～数十万収容するデータセンターで実現

- 実際のデータセンター
  - Google Data Center

    http://www.google.com/about/datacenters/



2.1 クラウドの語源

ネットワークこそが巨大なコンピュータ

- Eric Emerson Schmidt (Google CEO)の言葉
  [2006/11/16 The Economist より]
  ....(前略) Today we live in the clouds. We're moving into the era of "cloud" computing, with information and applications hosted in the diffuse atmosphere of cyberspace rather than on specific processors and silicon racks. The network will truly be the computer. ....(続く)

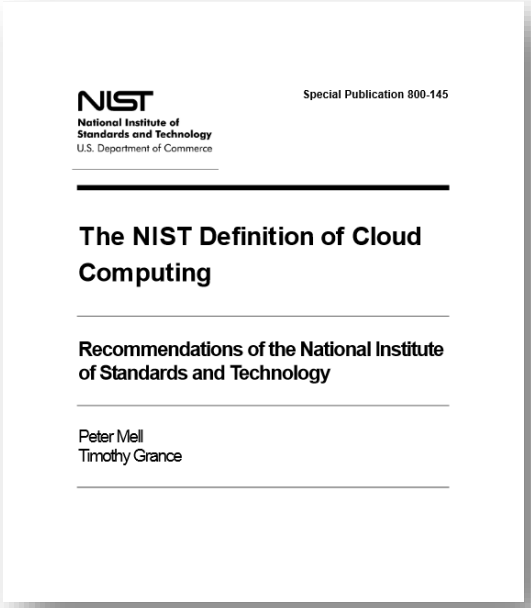
- ポイント
  - コンピュータ処理は，目の前の機器ではなく，雲 （＝ネットワーク．雲型のアイコン) のかなたで行われる.
  - ネットワークこそが巨大なコンピュータになる.

2.2 NISTによるクラウドコンピューティングの定義

クラウド業界でのコンセンサスが取れたリファレンス

- The NIST Definition of Cloud Computing
  - NIST: 米国国立標準技術研究所 (National Institute of Standards and Technology)
  - クラウドの様々な解釈を整理
  - クラウド業界でのコンセンサスが取れたリファレンス

- 3種類の概念でクラウドを性質づけ
  - 5つの本質的性質
  - 3つのサービスモデル
  - 4つの配備モデル


NIST
National Institute of Standards and Technology
U.S. Department of Commerce

Special Publication 800-145

The NIST Definition of Cloud Computing

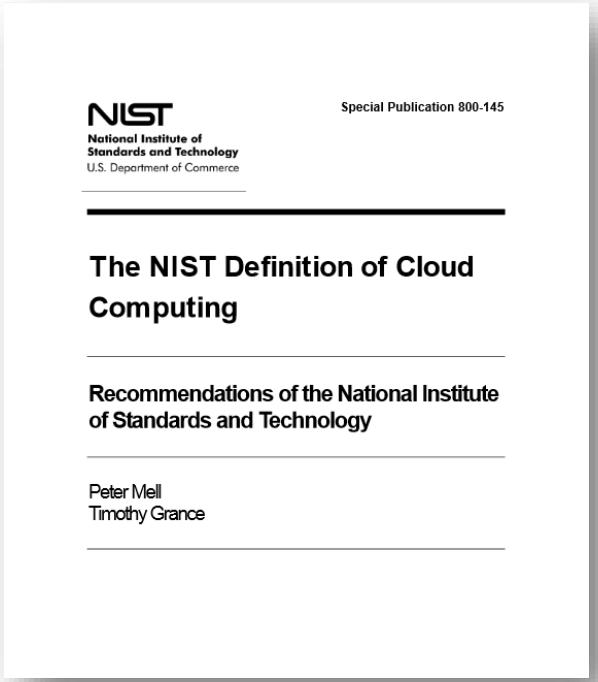
Recommendations of the National Institute of Standards and Technology

Peter Mell
Timothy Grance

2.3 クラウドを性質づける３つの側面

5つの本質的性質，3つのサービスモデル ， 4つの配備モデル

- 5つの本質的性質（Essential Characteristics）
  1. On-demand self-service
  2. Broad network access
  3. Resource pooling
  4. Rapid elasticity
  5. Measured service
- 3つのサービスモデル (Service Models)
  1. Software as a Service (SaaS)
  2. Platform as a Service (PaaS)
  3. Infrastructure as a Service (IaaS)
- 4つの配備モデル (Deployment Models)
  1. Public cloud
  2. Private cloud
  3. Community cloud
  4. Hybrid cloud



2.4.1. On-Demand Self-Service

■ 利用者は，いつでも好きなときに，自分で資源を利用可能

■ オンデマンド・セルフサービス

- ◆ 利用者は，いつでも好きなときに，自分で資源を利用できる
- ◆ 提供者の人手の介入は最小限にセルフサービスで始められる

2.4.2. Broad Network Access

多様なクライアントから標準的な方法で利用可能

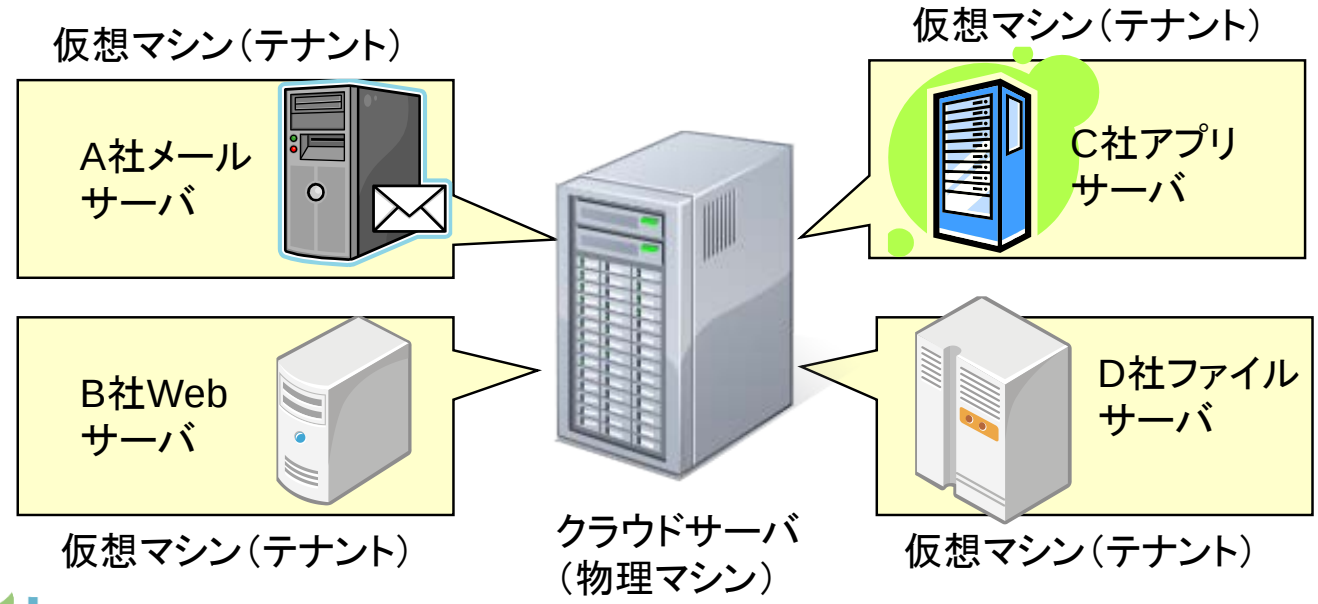
- 広いネットワークアクセス
  - 多様なクライアントから標準的な方法で利用可能である
  - シッククライアント (thick client)
    - Windows PC, Mac ノート，Linuxワークステーションなど
  - シンクライアント (thick client)
    - スマートフォン，タブレット，PDAなど

2.4.3. Resource Pooling

■ 計算資源は，物理的な実装や設置場所が隠蔽された形でプールされている

■ プールされた計算資源

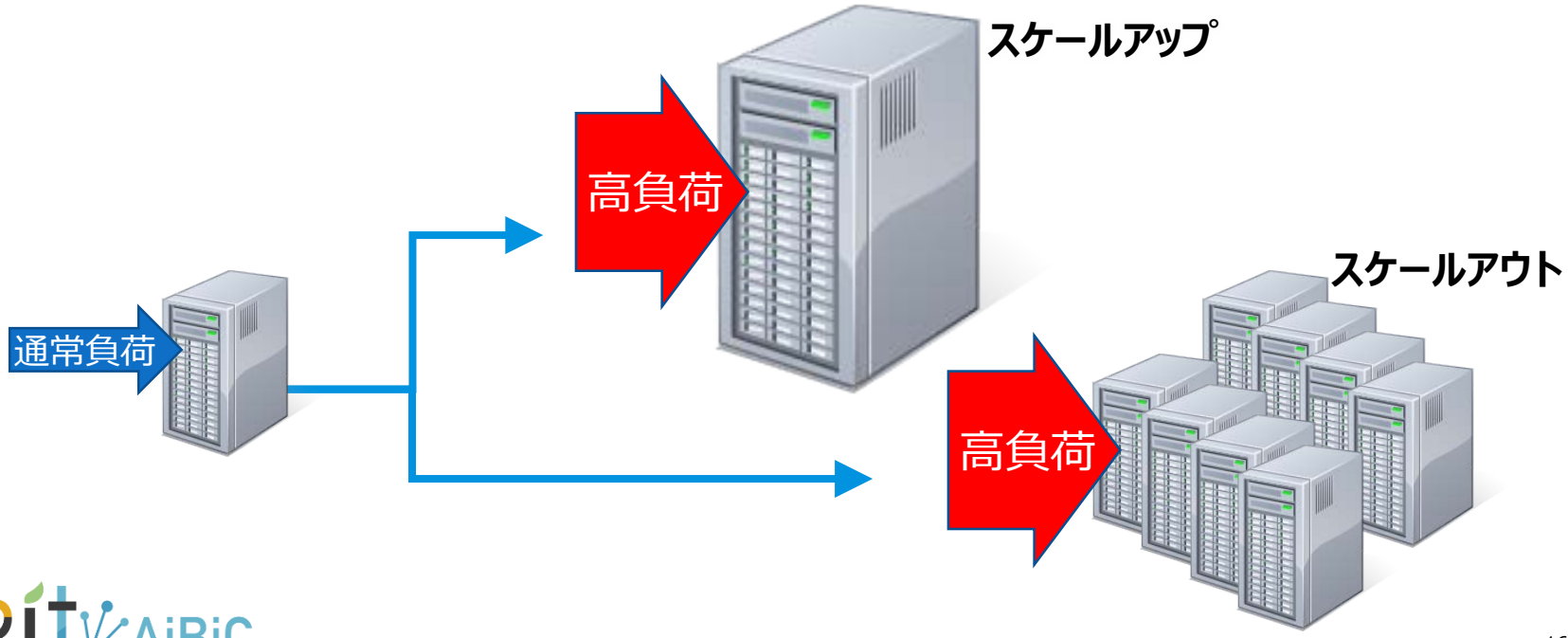
- ◆ 計算資源は，物理的な実装や設置場所が隠蔽された形でプールされている.
- ◆ 利用者からのリクエストに応じて，資源が貸し出される.
- ◆ 資源は複数の利用者間で共用される （マルチテナンシー）.



2.4.4. Rapid Elasticity

■ 必要なだけいくらでも使え，いらなくなったら返し，使った分だけ料金を支払う

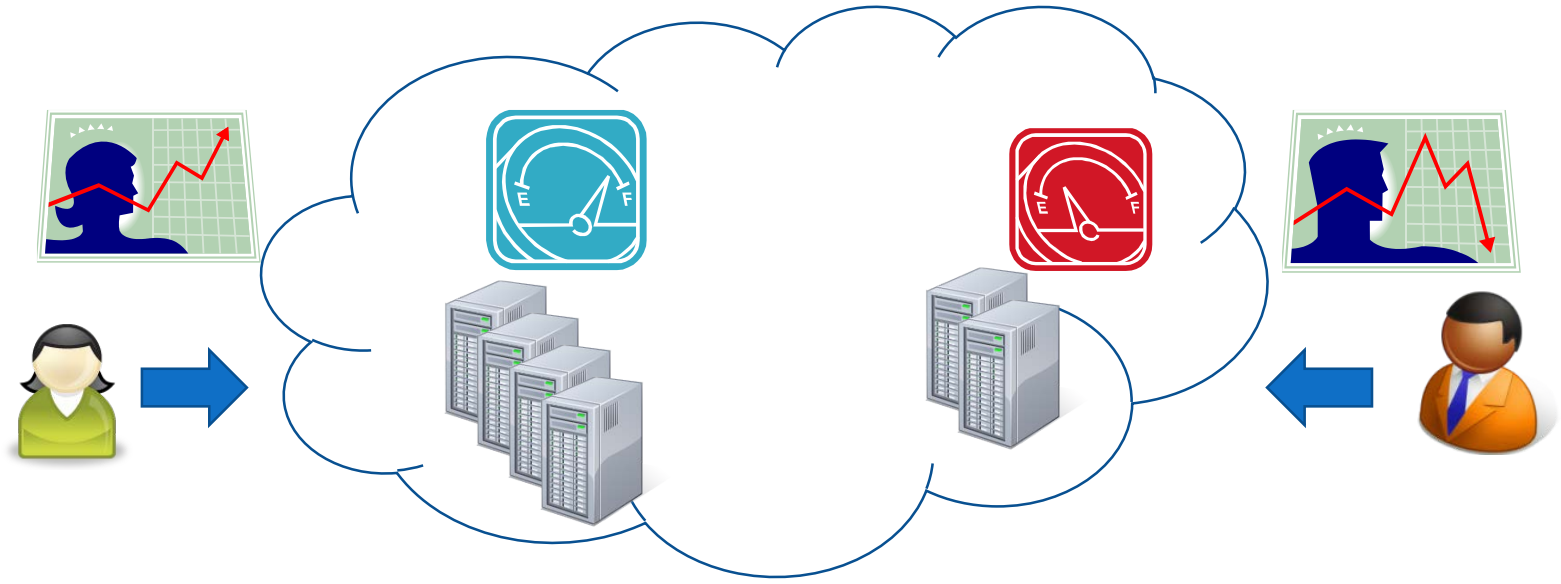
- 迅速な伸び縮み
  - 迅速かつ伸縮可能な計算資源の貸し出し
  - 動的なスケールアップ，スケールアウト
  - 必要なだけいくらでも使える．いらなくなったら返す
  - 使っただけの料金を支払う



2.4.5. Measured Service

■ どんな資源をいつどれだけ利用したかを計測可能

- 計測されたサービス
  - どんな資源をいつどれだけ利用したかをきちんと計測する
  - 計測に基づき自動的に最適な資源をサービスとして供給する



2.5 3つのサービスモデル

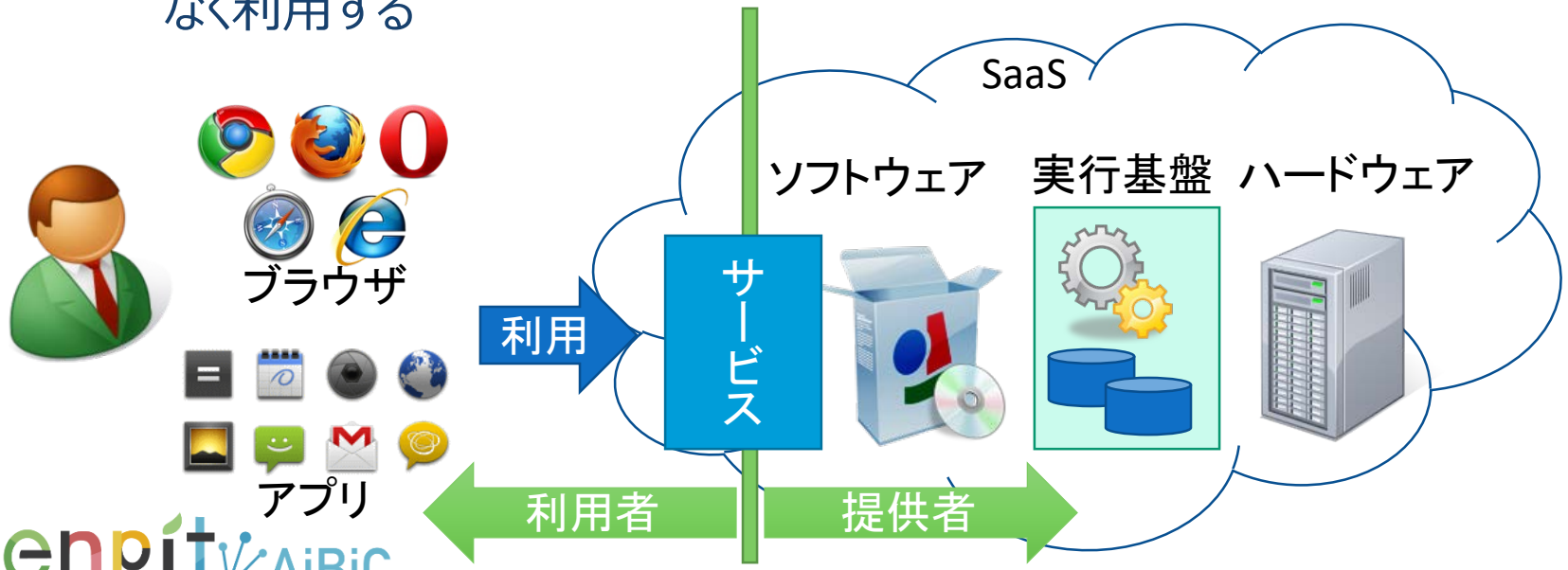
Service Models

- 3つのサービスモデル (Service Models)
  1. Software as a Service (SaaS)
  2. Platform as a Service (PaaS)
  3. Infrastructure as a Service (IaaS)

2.5.1 Software as a Service (SaaS)

■ ソフトウェア・アプリケーションをクラウド内のサービスとして提供する

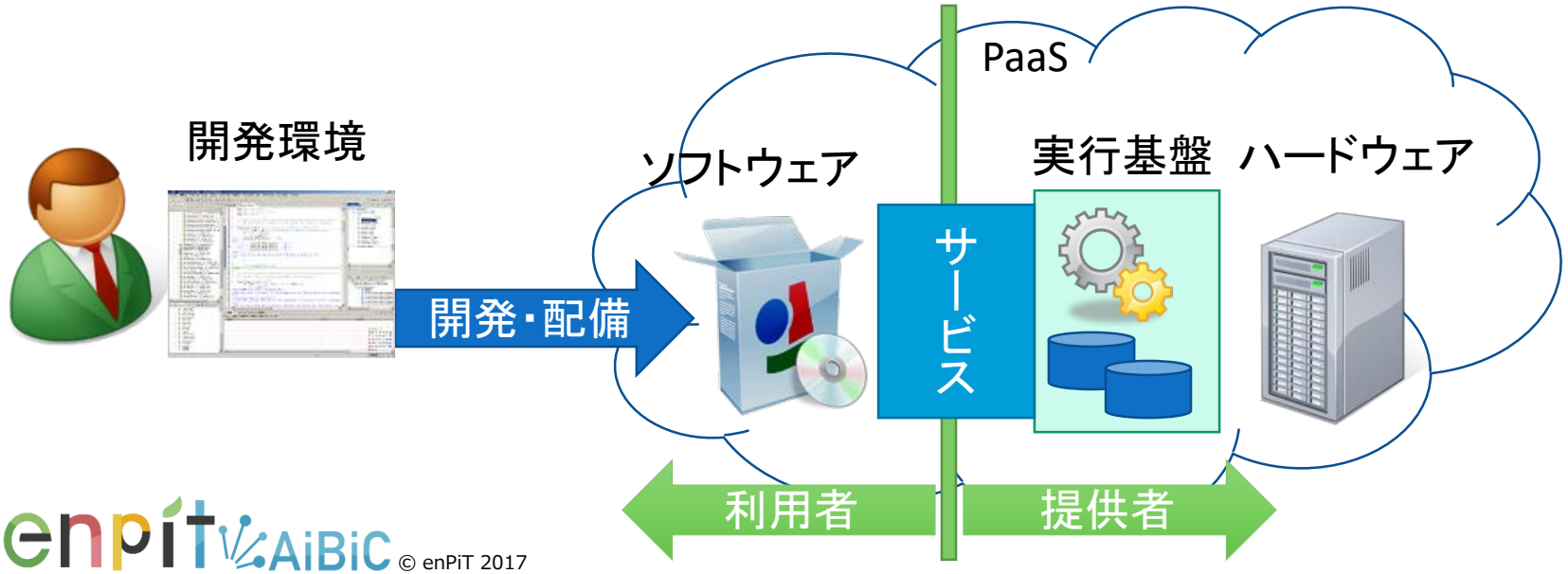
- ソフトウェア・アズ・ア・サービス，サース
  - 出来合いのソフトウェア・アプリケーションをクラウド内のサービスとして提供するもの
  - 利用者は様々なクライアントデバイスから利用．多くのSaaSではブラウザさえあれば利用可能であるが，よりリッチなクライアントプログラム（アプリ）が用意される
  - 利用者は，クラウド内部のソフトウェア実装やOS，ハードウェアを気にすることなく利用する



2.5.2 Platform as a Service (PaaS)

■ サーバソフト　等をサービスとして利用できるため，自前のSaaSを効率的に開発可能

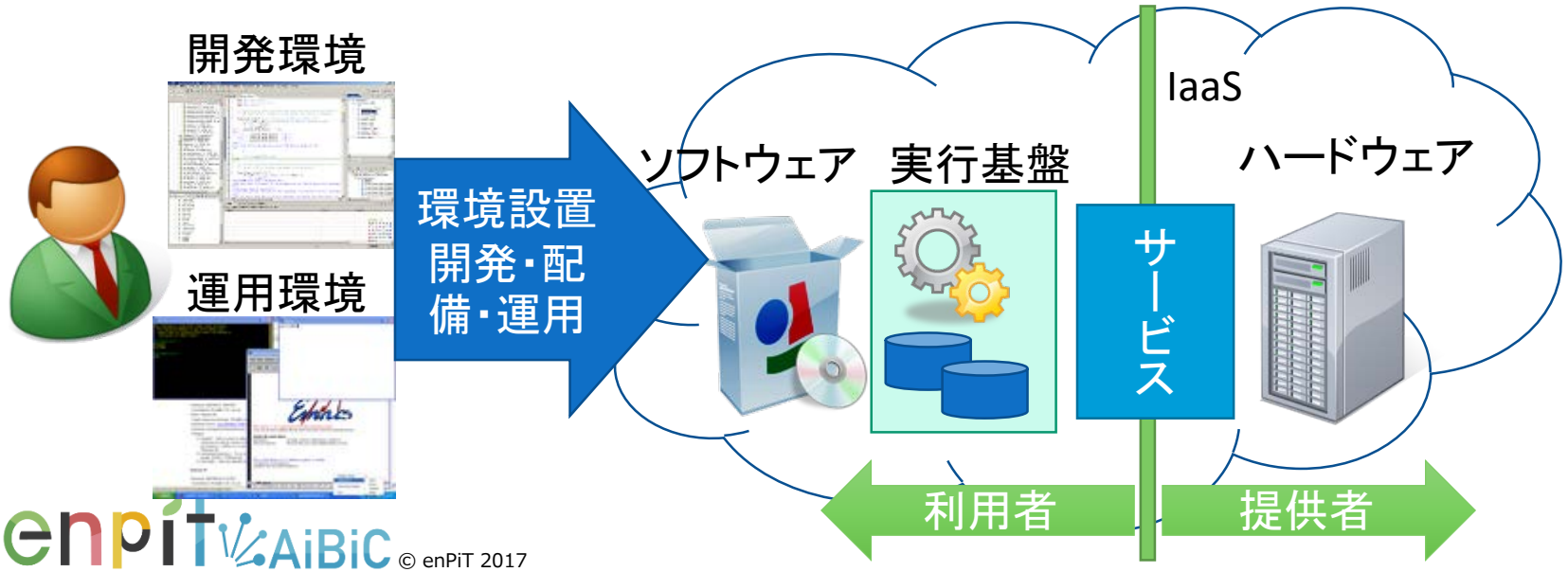
- プラットフォーム・アズ・ア・サービス，パース
  - 利用者自らが作成したソフトウェアを，クラウド内のサービスとして動作させるための実行基盤（プラットフォーム）を提供するもの.
  - PaaSが提供するサーバソフトやデータベースシステム，認証機構等をサービスとして利用できるため，自前のSaaSを効率的に開発できる.
  - 利用者は，自ら配備するソフトウェアを管理するが，PaaSが載っているサーバマシンやOS，実行環境の内部には立ち入らない.



2.5.3 Infrastructure as a Service (IaaS)

■ サーバマシンやデスクトップPC等，物理的なハードウェアを仮想マシンとして貸し出す

- インフラストラクチャ・アズ・ア・サービス，(ア)イアース
  - 利用者が任意のソフトウェアを実行するために必要となる，計算基盤(インフラ)をサービスとして提供するもの．
  - 多くの場合，サーバマシンやデスクトップPC等，物理的なハードウェアを仮想マシンとして貸し出すことを指す．
  - 利用者は，貸し出された仮想マシン，OS，配備するソフトウェアをすべて管理するが，最下層のハードウェア基盤には立ち入らない．



2.6 4つの配備モデル

Deployment Models

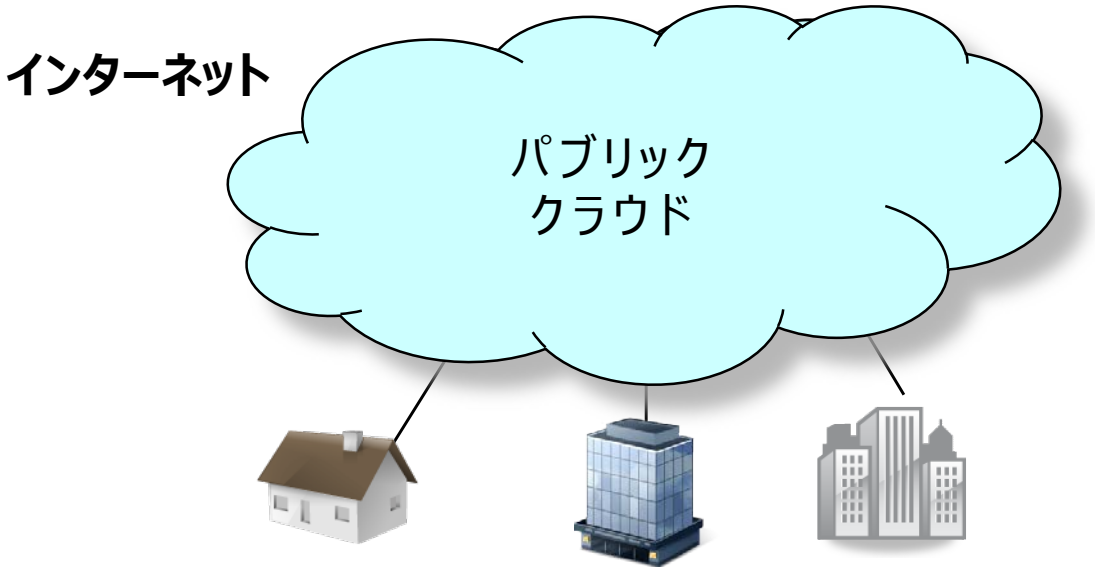
- 4つの配備モデル (Deployment Models)

1. Public cloud
2. Private cloud
3. Community cloud
4. Hybrid cloud

2.6.1 Public Cloud

■ 不特定多数の個人ユーザや組織にサービスを提供するクラウド

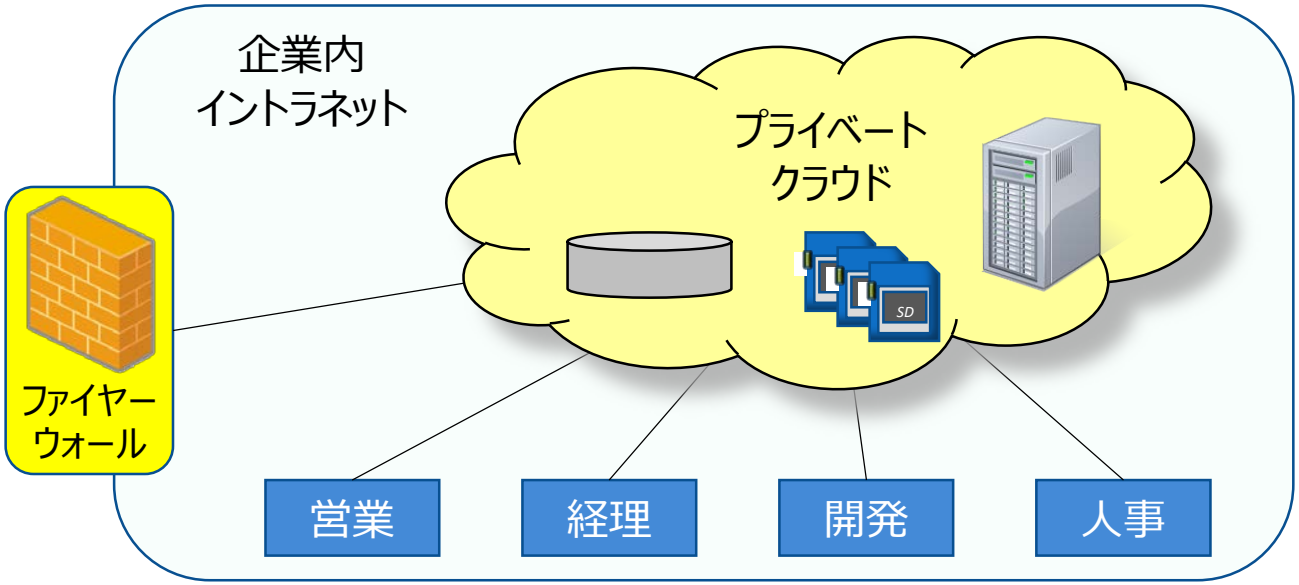
- パブリッククラウド
  - 不特定多数の個人ユーザや組織にサービスを提供するクラウド
  - 公衆のインターネット網を介してアクセスされる
  - クラウドのインフラは，サービス提供者によって所有・管理される



2.6.2 Private Cloud

特定の組織や企業のためだけに運用されるクラウド

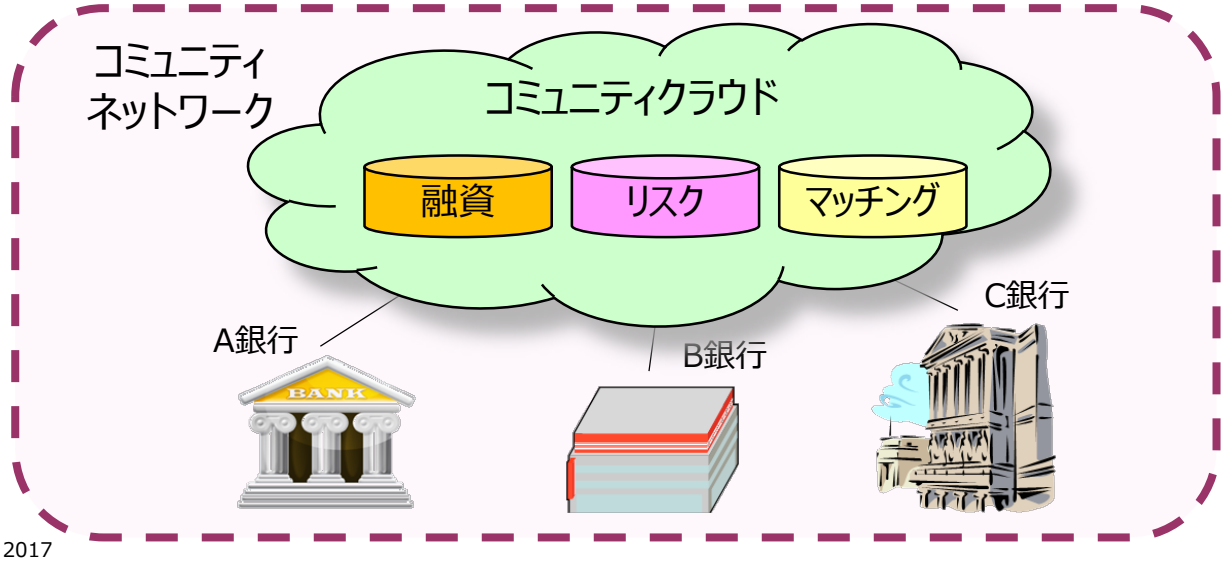
- プライベートクラウド
  - 特定の組織や企業のためだけに運用されるクラウド．部門間で資源を共用し，効率化する
  - ファイヤーウォール内のイントラネットや企業専用網を介してアクセスされる
  - クラウドのインフラは，その組織によって所有・管理される．



2.6.3 Community Cloud

■ 共通の関心事を持つ特定のコミュニティのためのクラウド

- コミュニティ クラウド
  - いくつかの組織やグループ企業で共有され，共通の関心事を持つ特定のコミュニティのためのクラウド．publicとprivateの中間のようなニュアンス
    - 共通の関心事：業務の共用可能な情報や，セキュリティやポリシー，法令，コンプライアンスなど
  - クラウドのインフラは，組織の集合によって所有・管理される．



2.6.4 Hybrid Cloud

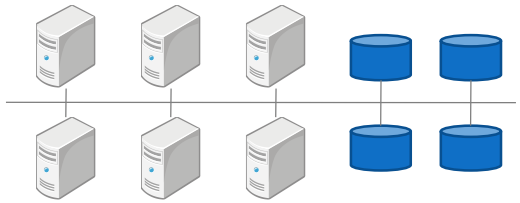
■ 上記のクラウドを2つ以上，統合・連携したクラウド

- ハイブリッド クラウド
  - 上記のクラウドを2つ以上，統合・連携したクラウド
  - 連携においては，標準化された，あるいは，固有の技術を用いて，データやアプリケーションの互換性を実現する.

3.1 クラウドを支えるキー技術 I

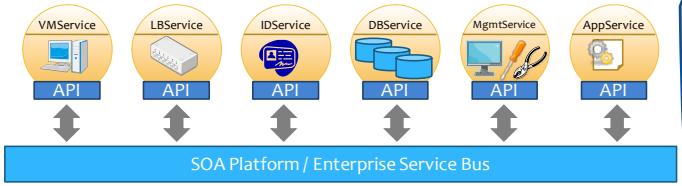
仮想化

仮想化



大規模分散処理

クラウド



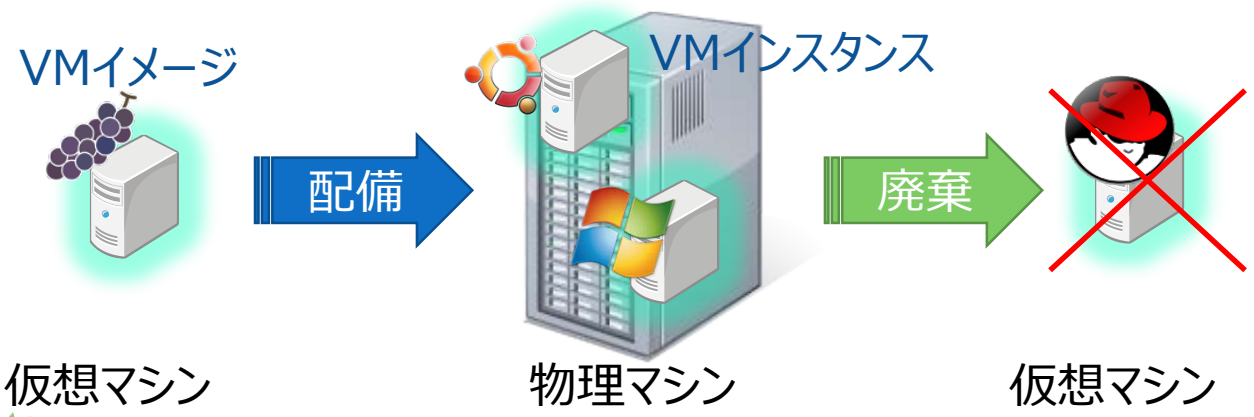
サービス指向アーキテクチャ

モバイルアプリケーション／
プラットフォーム

3.1.1 仮想化技術

エミュレーションと仮想マシン

- エミュレーション(emulation)によって計算機上に別の計算機を仮想的に再現する技術
  - クラウド上の計算資源の迅速な配備・廃棄を実現する
  - エミュレーション：あるシステムの上で，別のシステムの環境を模倣すること
- 仮想マシン (Virtual Machine, VM)
  - エミュレーションによって再現された仮想的な計算機



3.1.2 仮想化方式

ホストOS型とハイパーバイザー型

■ホストOS型

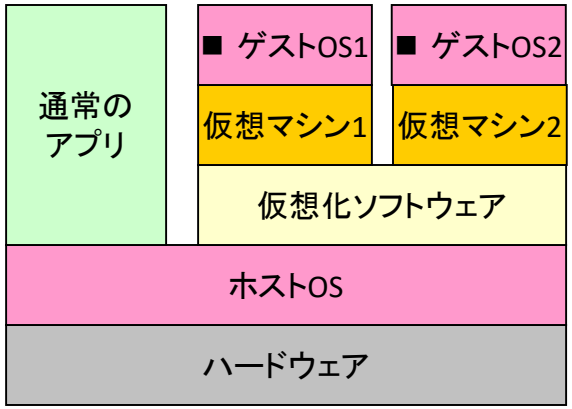
物理マシンのOS(ホストOS)上に仮想化SWをインストール．その上にVMとゲストOSをインストールする方法

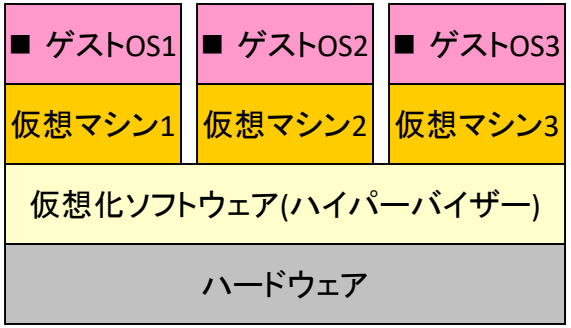
QEMU, VMware Server / Player, VirtualBox, etc.

■ハイパーバイザー型

物理マシンのHWの上に直接仮想化SW　（ハイパーバイザ）をかませ，その上にVMをインストールする方法

VMware ESXi, Xen, Hyper-V, etc.





3.1.4 クラウドを支える仮想化技術

仮想マシンとネットワーク仮想化

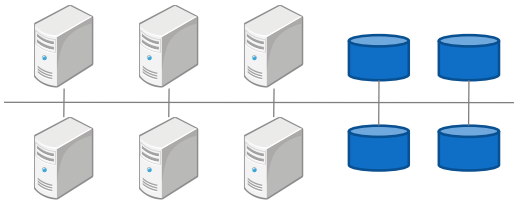
- 仮想マシン
  - IaaSの実現技術として用いられる
  - リクエストが来ると，物理マシン上に動的にマシンイメージを展開し，仮想マシンを起動する.
  - 物理資源に余裕がある限り，いくらでも仮想マシンを簡単に増やせる. ⇒ Rapid Elasticity, Resource Pooling

- ネットワーク仮想化
  - 従来HWで行ってきた経路制御やフィルタリング等をSWで行う
  - ネットワーク機器自体を仮想化する場合もある (e.g., vyatta)
  - 仮想マシンの生成・消滅に合わせて，迅速にネットワークを設定

3.2 クラウドを支えるキー技術 II

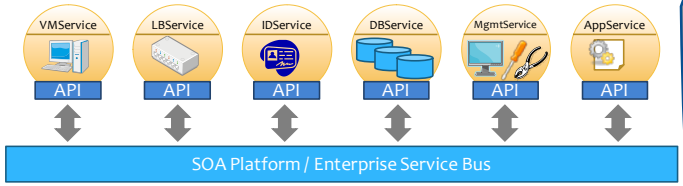
大規模分散処理

仮想化



大規模分散処理

クラウド



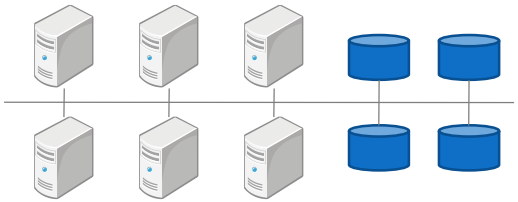
サービス指向アーキテクチャ

モバイルアプリケーション／
プラットフォーム

3.2.1 大規模分散処理

■ 大規模なデータを複数のサーバ，ストレージに分散配置し，並列・高速に処理

- 大規模なデータを複数のサーバ，ストレージに分散配置し，並列・高速に処理する技術
  - クラウド上のデータを効率よく裁く
  - クラウドを利用して大量のリソースを持つ処理基盤を調達



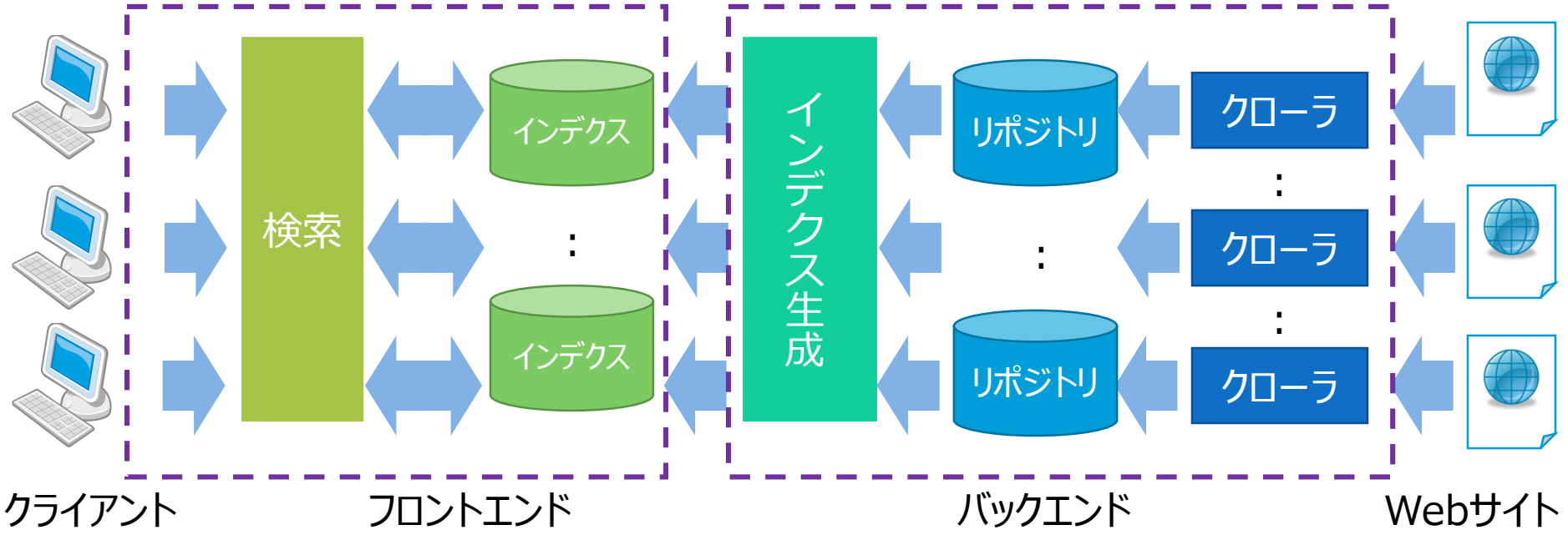
- ビッグデータ (Big Data)
  - 従来のデータベースやアプリケーションでは扱いが困難な巨大で複雑なデータの集合
  - ３V モデル　（Gartner）
    - Volume: サイズが非常に大きいデータであること
    - Variety:　ソースや形式が非常に多様なデータであること
    - Velocity: 発生から判断までの速度が非常に速いデータであること

3.2.2 Googleの分散処理基盤

フロントエンドとバックエンド

検索サービスのために開発されたデータ処理基盤

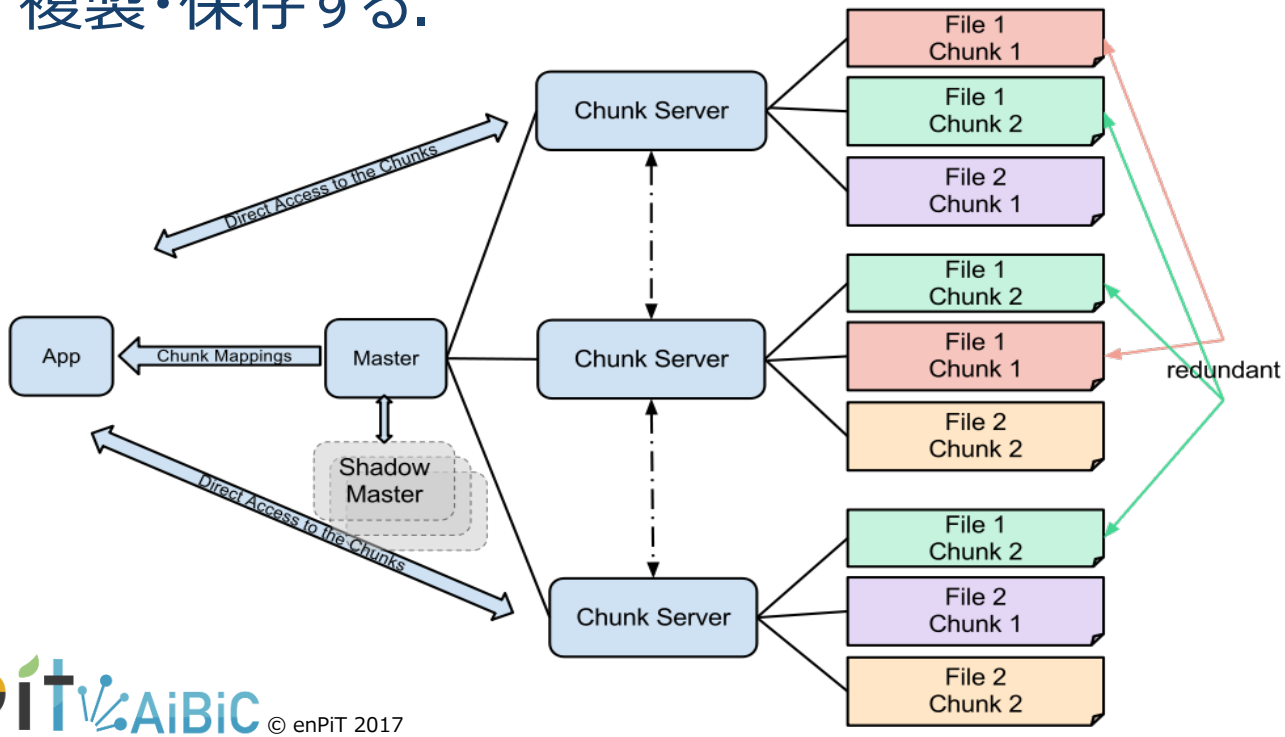
- 世界中からの検索クエリを瞬時に裁くフロントエンド
- 世界中の膨大なWebページ情報のクローリング，インデクス化を行うバックエンド



3.2.3 Google File System (GFS)

■ 同じデータを複製し，異なるマシンに持たせる（レプリケーション）

- 複数マシンのストレージをネットワーク越しに組み合わせ，ひとつのファイルシステムに見せかける技術
  - 同じデータを複製し，異なるマシンに持たせる（レプリケーション）
  - 大規模なデータは64MBのチャンクに分割し，3つ以上のサーバに複製・保存する.



出展：wikipedia
Created:18 September 2011

3.2.4 MapReduce

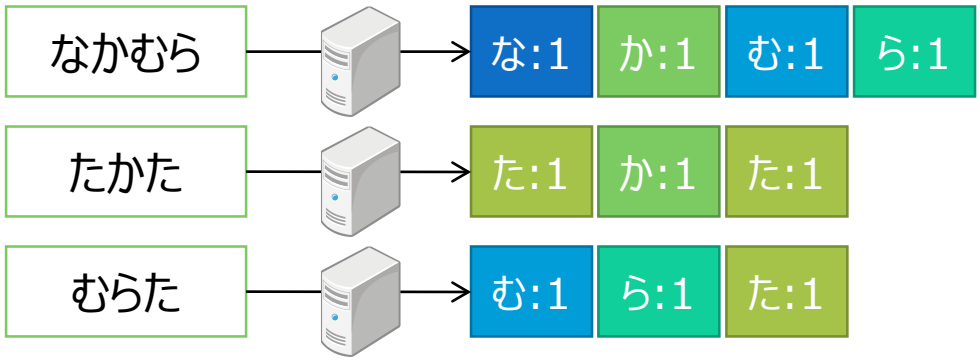
Map処理とReduce処理

- 大規模データを並列処理する仕組み．大量のデータやファイルを，多数のマシンで分担して処理する
  - Map処理：大規模データを分解し，各データをキーと値に変換
  - Reduce処理：データを同じキーで集計して結果を得る

3.2.4 MapReduce

Map処理とReduce処理

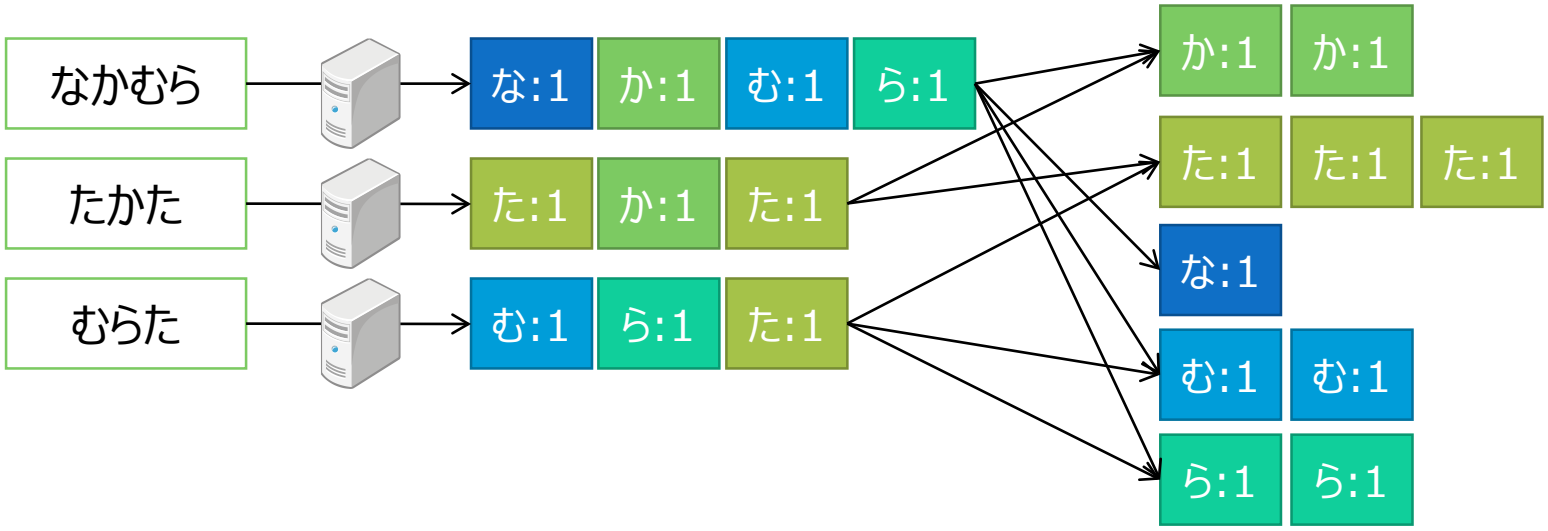
- 例題：受講生の名前に含まれる５０音の出現頻度を数えなさい



3.2.4 MapReduce

Map処理とReduce処理

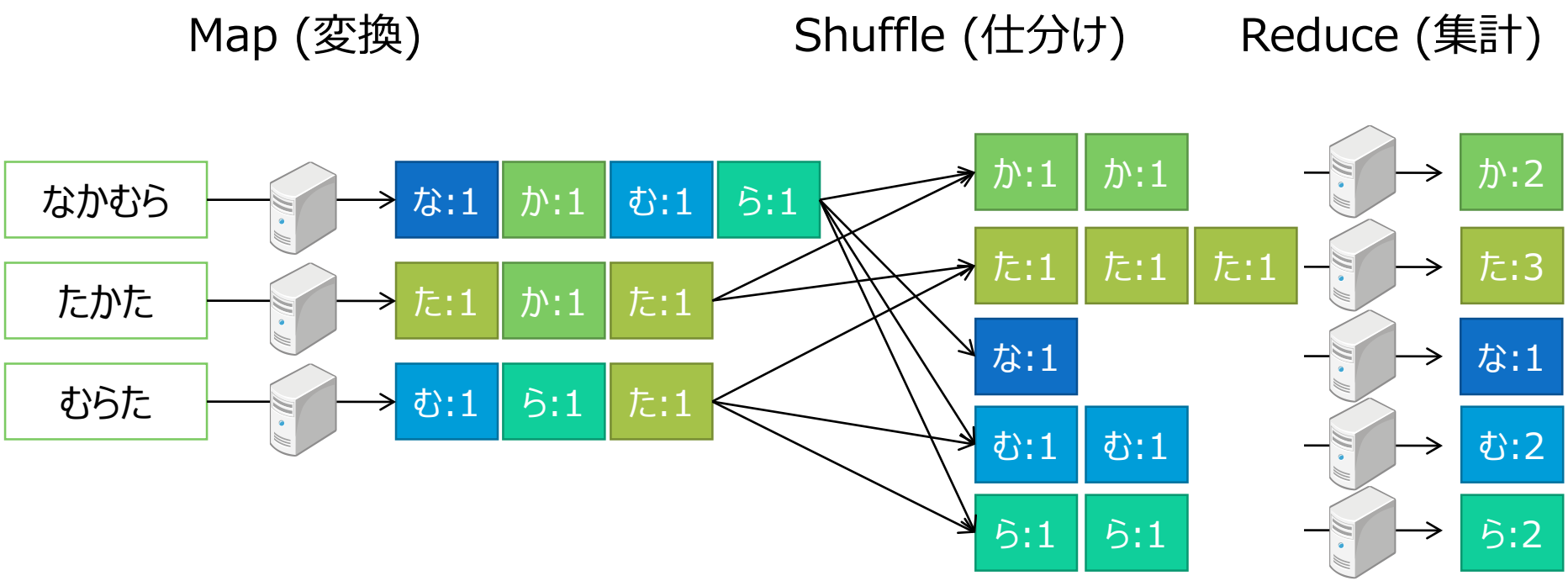
- 例題：受講生の名前に含まれる５０音の出現頻度を数えなさい



3.2.4 MapReduce

Map処理とReduce処理

- 例題：受講生の名前に含まれる５０音の出現頻度を数えなさい



3.2.5 Key-Value Store (KVS)

RDBが不得意とする，高スケーラビリティ・伸縮性をねらうクラウド時代のデータベース

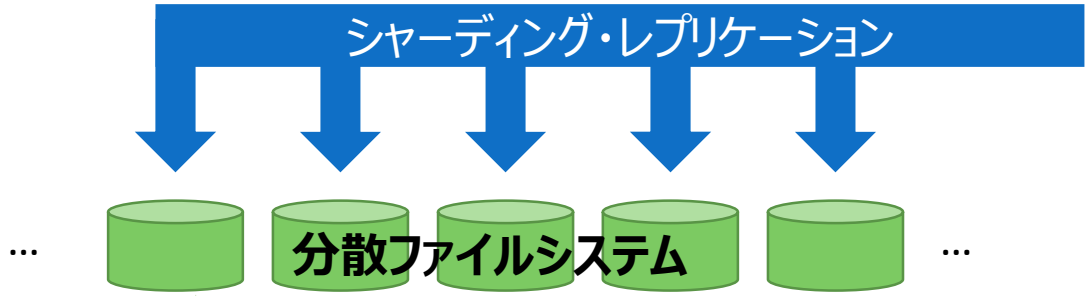
- データを識別子(key)と値(value)で保存するデータベース
  - 従来の関係データベース(RDB)が不得意とする，高スケーラビリティ・伸縮性をねらうクラウド時代のデータベース．NoSQL
  - Google BigTable, HBase, Cassandra, MongoDB, CouchDB, etc.

RDBによる名簿データベース

ID	名前	性別	Email
2013901	中村匡秀	男	masa-n@cs.kobe-u.ac.jp
2013902	まつ本真佑	男	shinsuke@cs.kobe-u.ac.jp
:	:	:	:

KVSによる名簿データベース

Key	Value
2013901.名前	中村匡秀
2013901.性別	男
2013901.Email	masa-n@cs.kobe-u.ac.jp
2013902.名前	まつ本真佑
2013902.性別	男
2013902.Email	shinsuke@cs.kobe-u.ac.jp
:	:



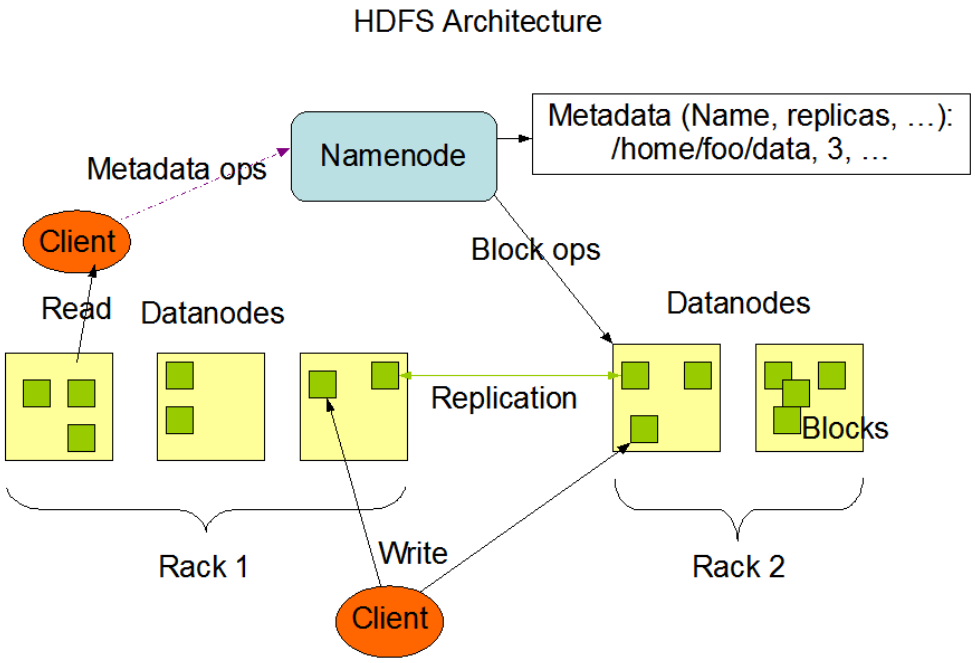
3.2.6 Hadoop

Google分散処理基盤のオープンソース・クローン

- Google分散処理基盤のオープンソース・クローン
  - Apache Foundationのトッププロジェクト
  - Javaで実装されている

Hadoop	Google基盤
HDFS	GFS
Hadoop MapReduce	MapReduce
HBase	Google Bigtable

HadoopとGoogle基盤の対応

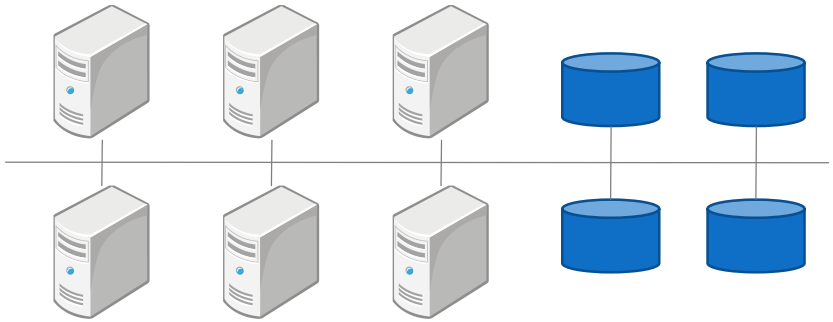


HDFSのアーキテクチャ

3.2.7 クラウドを活用した大規模分散処理

大規模分散処理に見られる３つのサービス形態

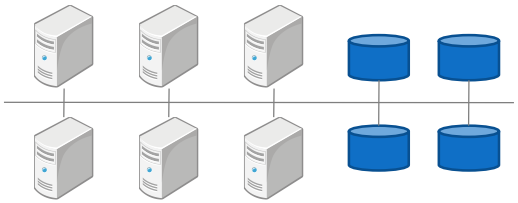
- 手元の環境では処理しきれない大規模データの処理をクラウドに外注
  - オフローディング (Offloading)
- 必要に応じて計算リソースを追加・削除
  - クラスタ as a Service
  - Amazonの計算クラスタ
- データ処理サービスの活用
  - 計算 as a Service
  - HPC as a Service



3.3 クラウドを支えるキー技術 III

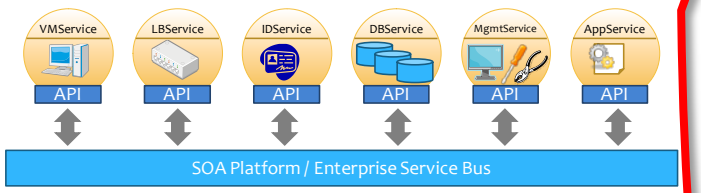
サービス指向アーキテクチャ

仮想化



大規模分散処理

クラウド



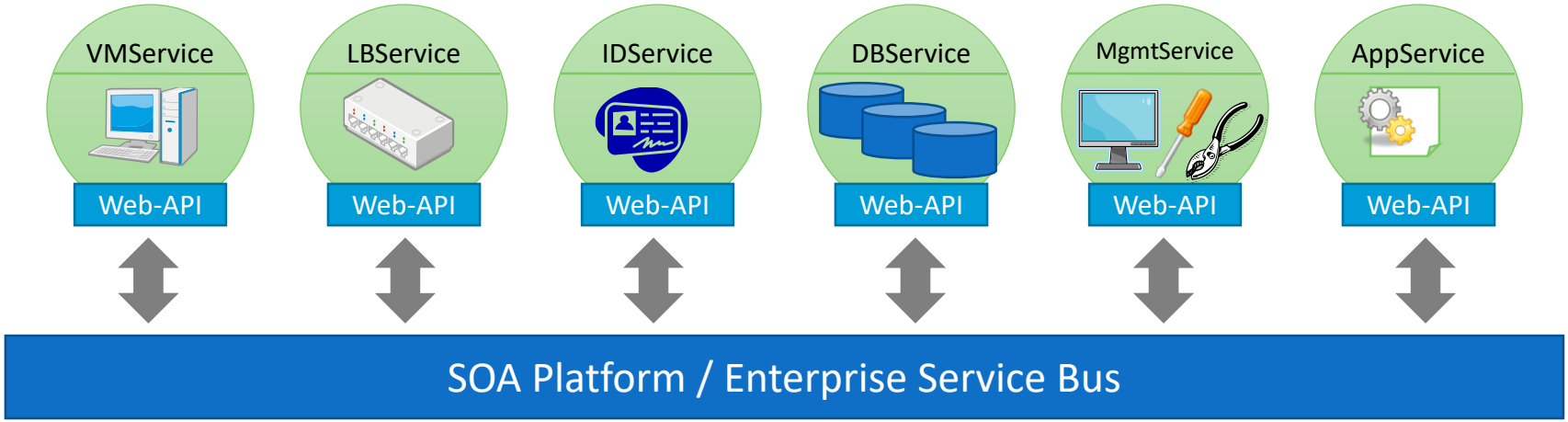
サービス指向アーキテクチャ

モバイルアプリケーション／
プラットフォーム

3.3.1 サービス指向アーキテクチャ(SOA)

■ システムや機器が提供する機能をサービスという単位で捉える

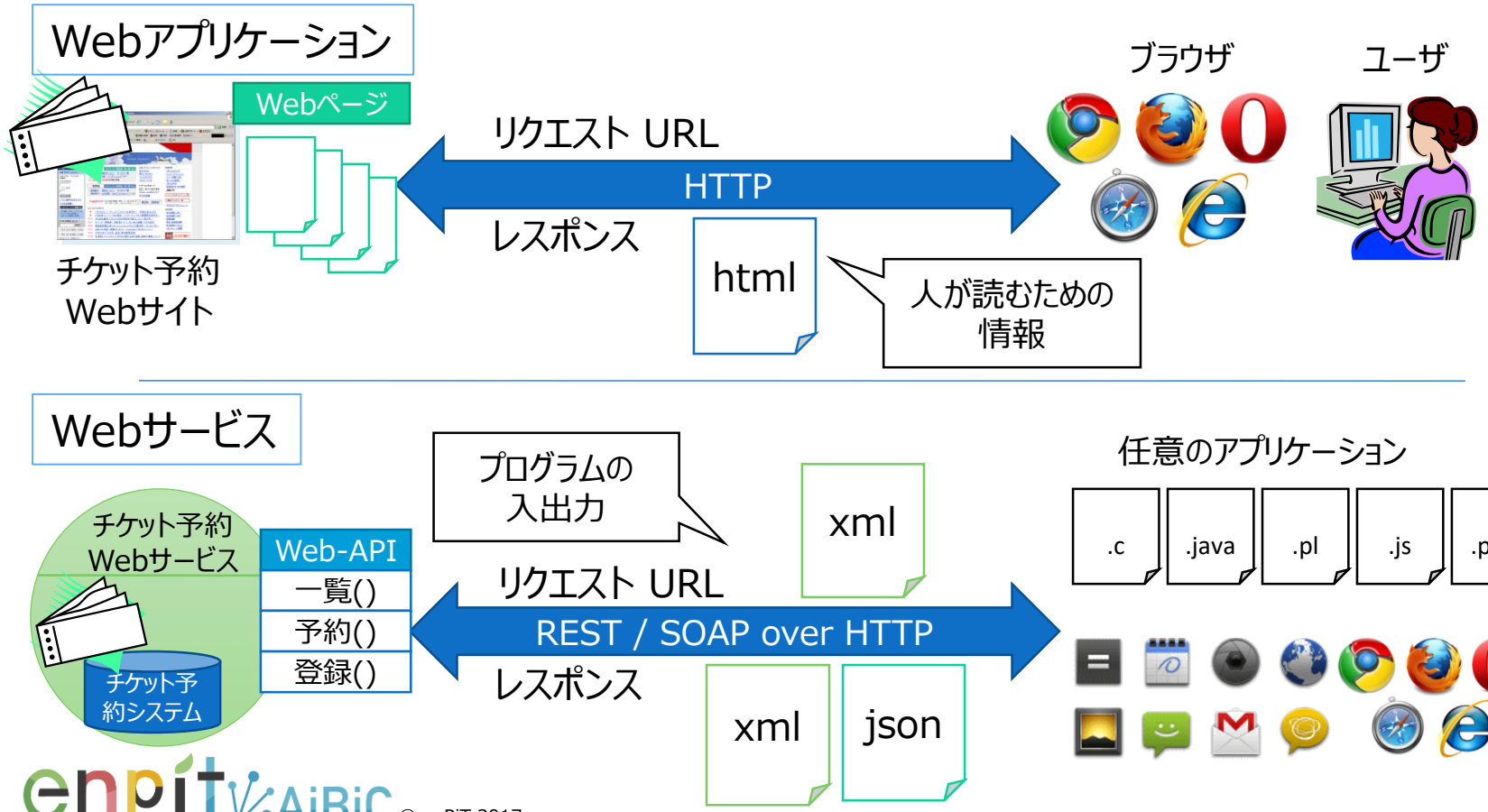
- システムや機器が提供する機能をサービスという単位で捉えるシステムアーキテクチャ (Service-Oriented Architecture)
  - 既存のサービス同士をゆるく連携（疎結合）することで，新たなサービスを構築する
  - クラウド内の計算資源をサービス化する技術として，SaaS, PaaS, IaaSの各レベルで適用される



3.3.2 Webサービス

SOAの実装技術

- Webの仕組みを用いて，プログラムから他のシステムの機能を呼出し・実行する仕組み （SOAの実装技術）



3.3.3 HTML, XML, JSON

チケット予約の事例

■チケットの情報

Cloudy Jazz Orchestra

・神戸みなとホール
・2013年7月10日　18:30開演
・S席　　¥12,000円（税込）
ID: 130124-0010

■HTML (読み手は人間)

```
<html>
  <body>
  <H1>Cloudy Jazz Orchestra</H1>
  <hr>
    <ul>
      <li>神戸港ホール </li>
      <li>2013年7月10日 </li>
      <li>¥12,000円 </li>
    </ul>
    <p> <a href=”t/130124-0010.html”>
    ID:130124-0010</a> </p>
  </body>
</html>
```

■XML (読み手はプログラム)

```
<ticket>
  <event>Cloudy Jazz Orchestra</event>
  <place>神戸港ホール </place>
  <date>2013-07-10</date>
  <seat>S</seat>
  <price>12000</price>
  <refId>130124-0010</refId>
</ticket>
```

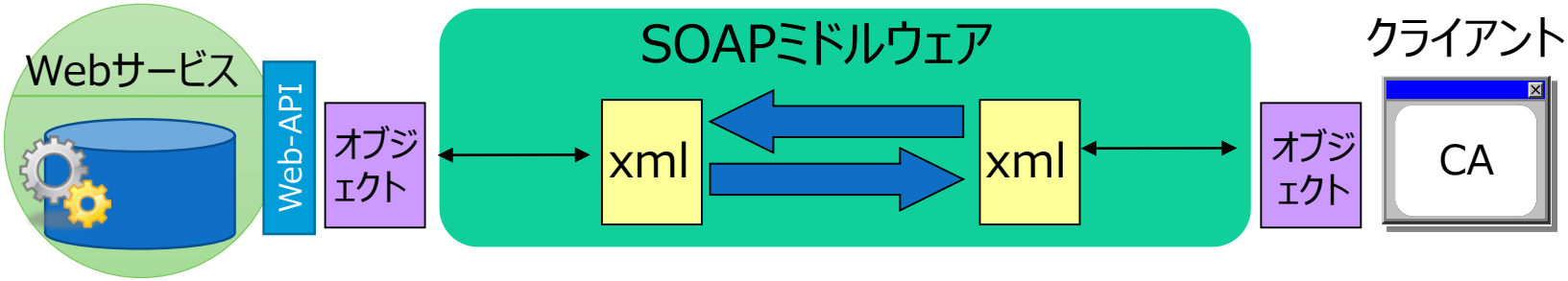
■JSON (読み手はプログラム)

```
{
  ”ticket”: {
    ”event”: ”Cloudy Jazz Orchestra”,
    ”place”: ”神戸港ホール”,
    ”date”:  ”2013-07-10”,
    ”seat”: ”S”,
    ”price”: 12000,
    ”refId”: ”130124-0010”,
  }
}
```

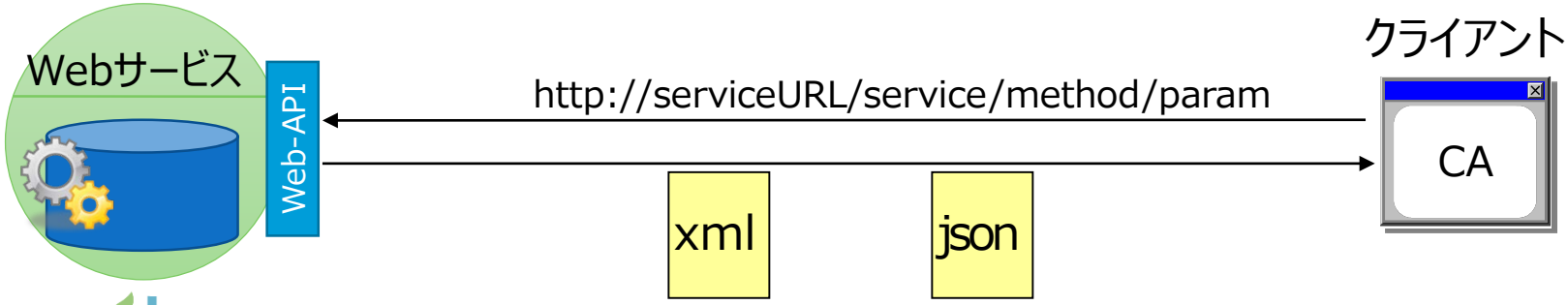
3.3.4 Webサービスのプロトコル

SOAPとREST

- SOAP: 入出力にXMLをやり取りして，Webサービスを呼び出す方法．XML／オブジェクトの変換はSOAPミドルウェアが行う．



- REST: 入力はHTTPメソッド，出力にXMLを使って，簡易的にWebサービスを呼び出すプロトコル．XML／オブジェクトの変換は手動



3.3.5 Web-API

Webサービスを利用するためのAPI

- Webサービスを利用するためのAPI
  - API: Application Program Interface (特定のシステムを利用するための命令や関数の集合)
  - ブラウザからも呼び出せるが，通常はプログラムから呼び出して戻り値を処理する

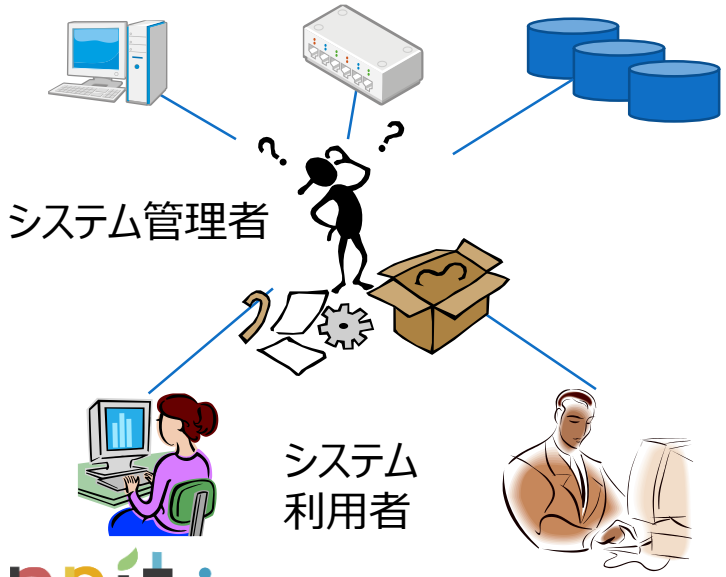
- Web-APIの呼び出し例
  - お天気Webサービス (REST-JSON) [JSON整形サービスで確認]
    - 神戸の天気予報 http://weather.livedoor.com/forecast/webservice/json/v1?city=280010
  - 路線／駅名 Webサービス (REST-XML)
    - 近畿地方の全路線
      http://express.heartrails.com/api/xml?method=getLines&area=%E8%BF%91%E7%95%BF
    - 阪急六甲駅の情報
      http://express.heartrails.com/api/xml?method=getStations&name=%E5%85%AD%E7%94%B2

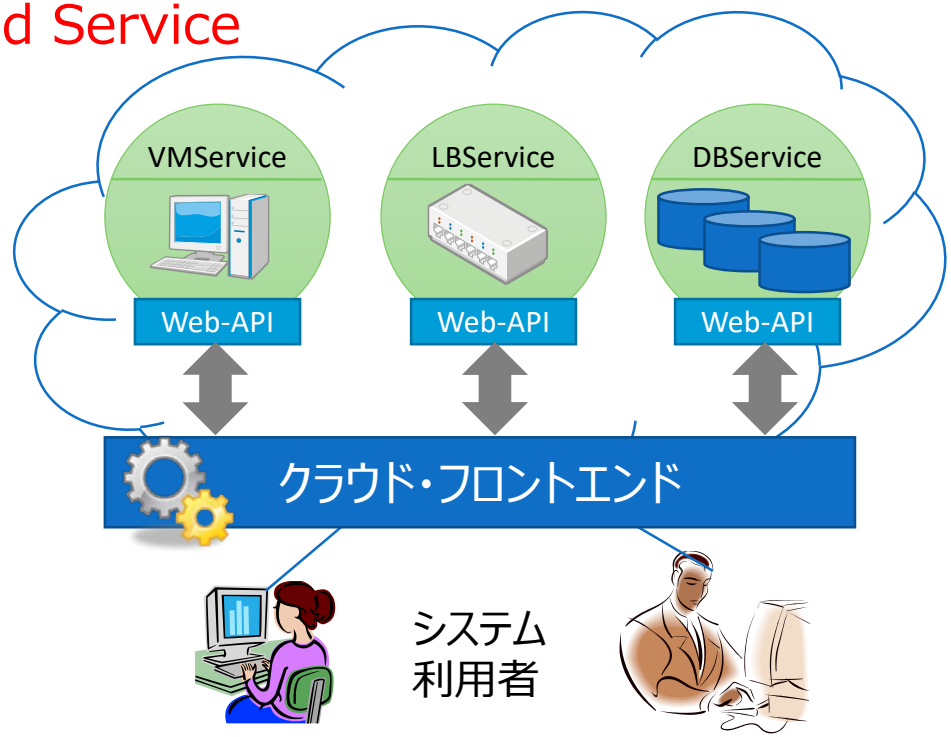
3.3.6 クラウドを支えるSOA

■ 様々なプログラムから人手を介さずに利用可能

- クラウド内の様々な資源をサービス化
- サービスを組み合わせてより高度なサービス（マッシュアップ）
- 様々なプログラムから人手を介さずに利用可能

⇒ On-demand Self-service, Broad Network Access, Rapid Elasticity, Resource Pooling, Measured Service

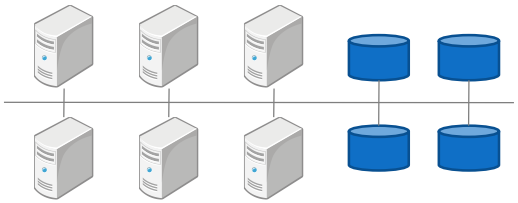




3.4 クラウドを支えるキー技術Ⅳ

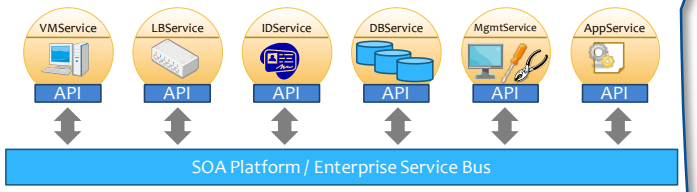
モバイルアプリケーション／プラットフォーム

仮想化



大規模分散処理

クラウド



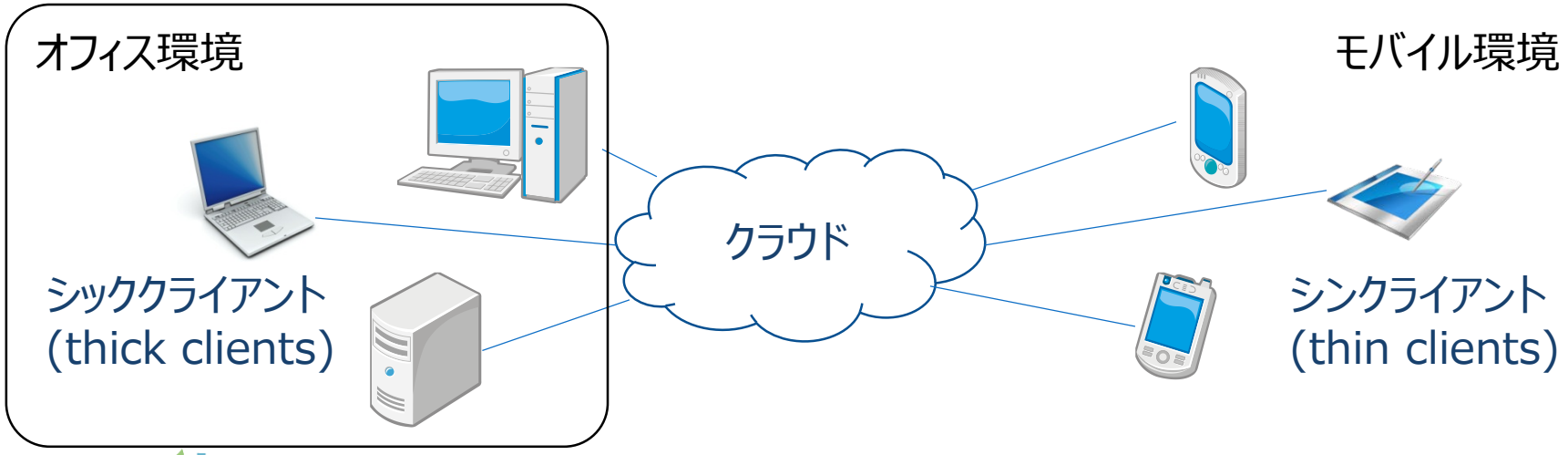
サービス指向アーキテクチャ

モバイルアプリケーション／
プラットフォーム

3.4.1 モバイルクラウドの潮流

■ 非力なCPU，小さい画面でも使いやすくする工夫が必要

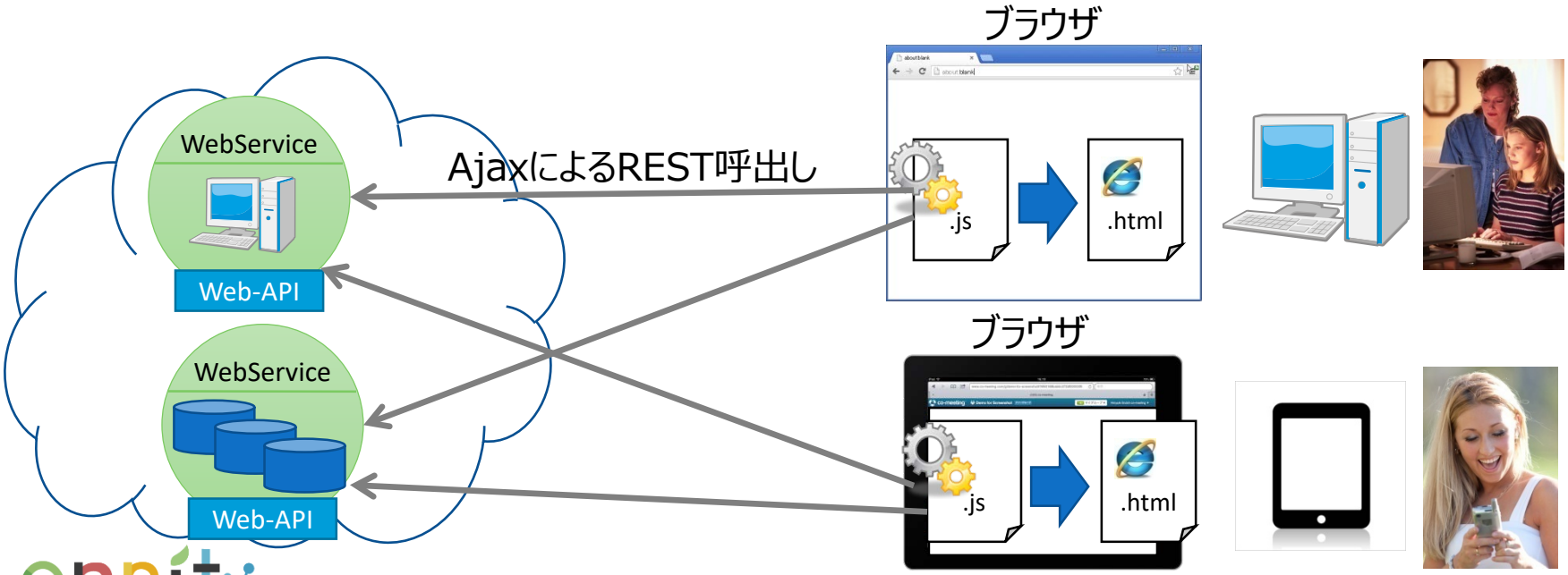
- クラウドサービスをモバイル端末で利用する
  - スマートフォン，タブレットなどのシンクライアントから利用
  - 従来のオフィスでのPC環境だけでなく，いつでもどこでも同じサービスが受けられる ⇒ Broad Network Access
    - c.f. BYOD (Bring Your Own Device)
  - 非力なCPU，小さい画面でも使いやすくする工夫が必要



3.4.2 アプリケーションプラットフォームとしてのブラウザ

ブラウザをWebアプリの実行基盤の一部

- ブラウザをWebアプリの実行基盤の一部として使う
  - ブラウザは様々なクライアント端末に標準装備されている
  - JavaScriptプログラムの実行が可能
  - JSからクラウドのWebサービスを呼び出し，結果のXMLを解析して，ブラウザにレンダリング(Ajax)
    ⇒ どんな端末でもいつでもサービスを利用可能.



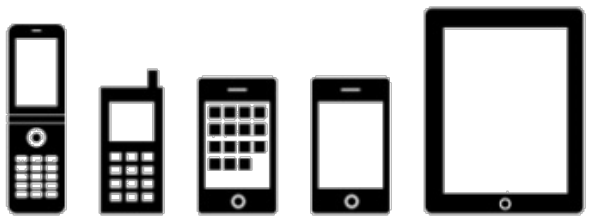
3.4.3 Webアプリケーションの標準技術

HTML5，CSS3，JavaScript

- HTML5 (Hyper Text Markup Language Ver.5)
  - ブラウザに表示する文書の構造を記述

- CSS3 (Cascading Style Sheets Ver.3)
  - ブラウザに表示する文書の体裁，デザインを指定

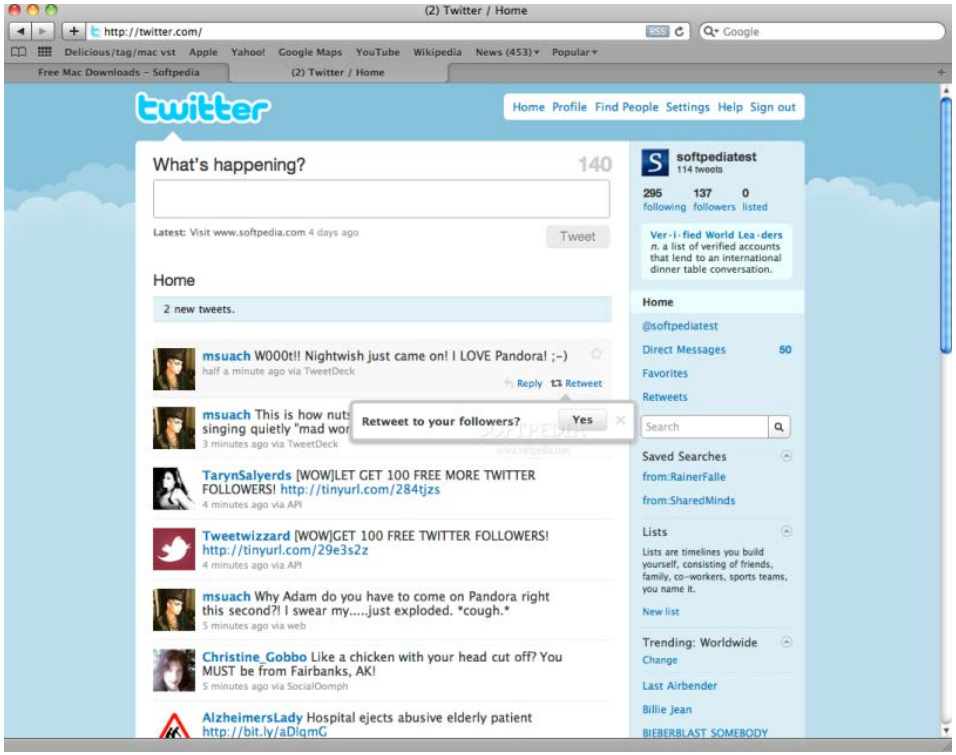
- JavaScript
  - ブラウザで実行する処理（振る舞い）を記述



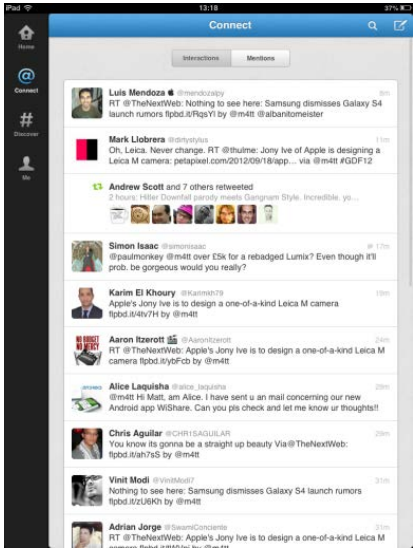
3.4.4 PCアプリとモバイルアプリの比較

■ 中核を担保するPCアプリ，実用性重視のモバイルアプリ

Twitter



PCアプリ

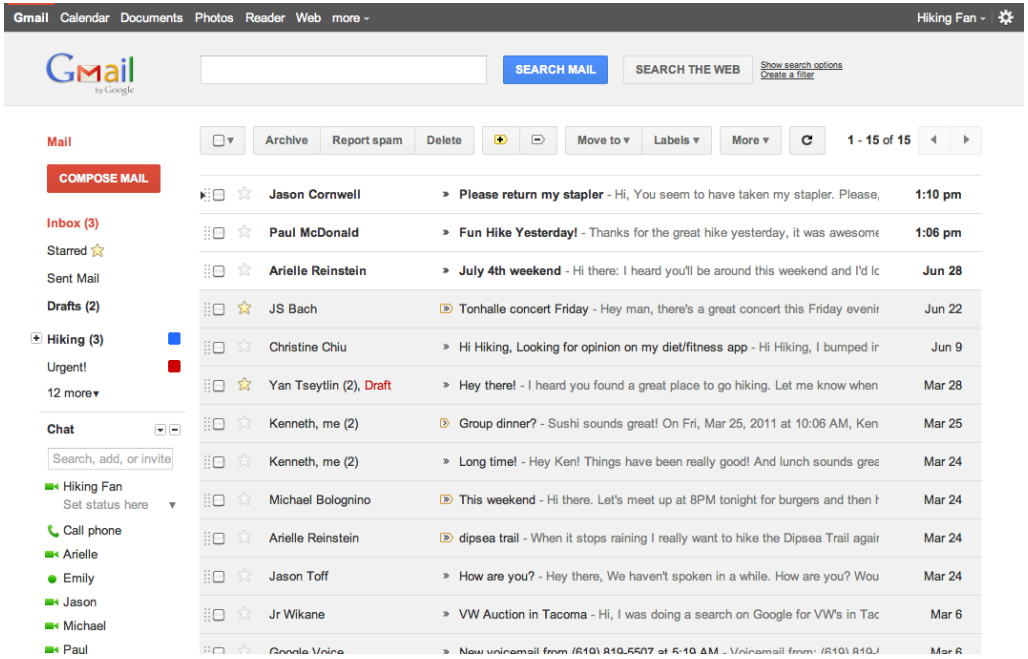


モバイルアプリ

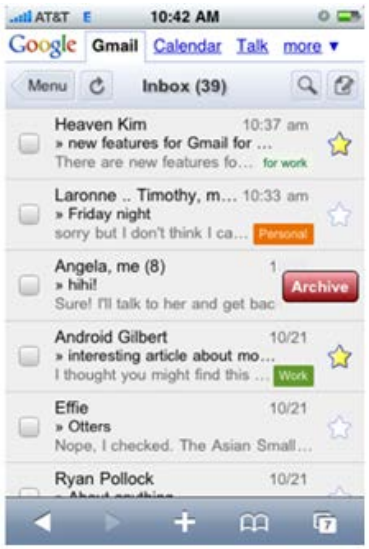
3.4.4 PCアプリとモバイルアプリの比較

- 中核を担保するPCアプリ，実用性重視のモバイルアプリ

Gmail



PCアプリ



モバイルアプリ

4.1 クラウドがもたらすパラダイムシフト

■ ものを手元に持たず，サービスを使った分だけ対価を支払う

- 所有から利用へ
  - 手元に持たず，使った分だけ支払う
- モノからサービスへ
  - モノを買うのではなく，サービスを買う
- シック（ファット）からシンへ
  - 手元のマシンはクラウドを使うための窓口
- 分散から集中へ
  - ネットワークを１つの大きな仮想コンピュータとみなす
- [ひと to マシン]から[マシンtoマシン]へ
  - サービスのAPIへプログラムから直接アクセス

4.2 クラウドの長所（続く）

■ どこからでも何からでもアクセスできる

- 手元に抱えなくてよい
  - 置き場所，配線，インストール，メンテナンスが不要
- 手軽に導入，簡単に破棄できる
  - スモールスタート，使い捨てが可能
- どこからでも何からでもアクセスできる
  - データやソフトはクラウドにあり，モバイル端末からでもアクセス
- 共有が容易
  - 資源のアクセスはURL(WebAPI, Webページ)で画一的に行える
- 重い処理や大きなデータの保存を外注できる
  - 手元でできないことをクラウドにお願いしてやってもらう

4.2 クラウドの長所（続き）

■ APIを組み合わせて様々な自動化サービスを作成可能

- 必要なリソースを必要なだけ使える
  - とりあえず今必要なだけ使う．後で追加できる
- スケーラビリティ，耐故障性を確保できる
  - 台数を増やしてスケールアウト，レプリケーションで故障に対応
- プログラマブルである
  - APIを組み合わせて様々な自動化サービスを作成可能

4.3 クラウドの短所

細かい設定・調整ができない

- セキュリティ，共用に不安
  - 貴重なデータを社外へ？ 他者と同じ物理サーバに入る？
- クラウド提供者への強依存
  - クラウドが潰れれば道連れ？
- 細かい設定・調整ができない
  - 用意されているサービスを使うというスタンス
- デプロイすると動かない
  - 手元で動いたのに，クラウドでは動かない・・・
- 今までの開発手法が通用しない
  - クラウドの新技術 (例：MapReduce, KVS) をどう使いこなすか