AWS Summits 2014

TE-08 実用フェーズに入ったHPCクラウドの実力

松尾康博 (matsuoy@amazon.co.jp)
アマゾン データ サービス ジャパン
ソリューション アーキテクト

amazon web services

自己紹介

- 名前
  - 松尾康博(matusoy@amazon.co.jp)
- 仕事
  - ソリューションアーキテクト
  - HPC, ビッグデータに関するお客様を担当
- 好きなAWSのサービス
  - C3.8xlarge , API

Agenda

- AWSとは？
- なぜHPC on AWSなのか？
- クラスタインスタンスの性能
- お客様事例

Agenda

- AWSとは？
- なぜHPC on AWSなのか？
- クラスタインスタンスの性能
- お客様事例

Agenda

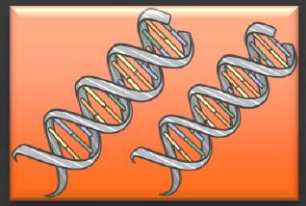
- ~~AWSとは？~~
- なぜHPC on AWSなのか？
- クラスタインスタンスの性能
- お客様事例

Agenda

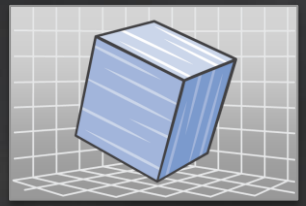
- AWSとは？
- なぜHPC on AWSなのか？
- クラスタインスタンスの性能
- お客様事例

主要なHPCアプリケーション

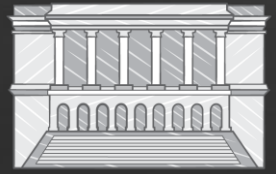
ゲノム解析



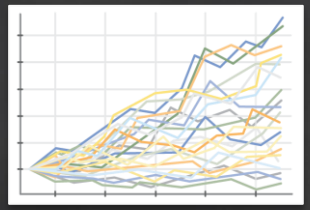
モデリング
シミュレーション



教育機関・政府機関

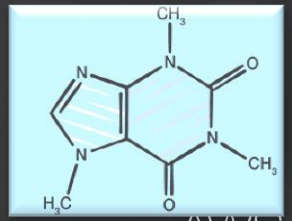


モンテカルロ
シミュレーション



トランスコーディング
エンコーディング

計算化学



お客様のお悩み

集約した共用計算機だと・・・

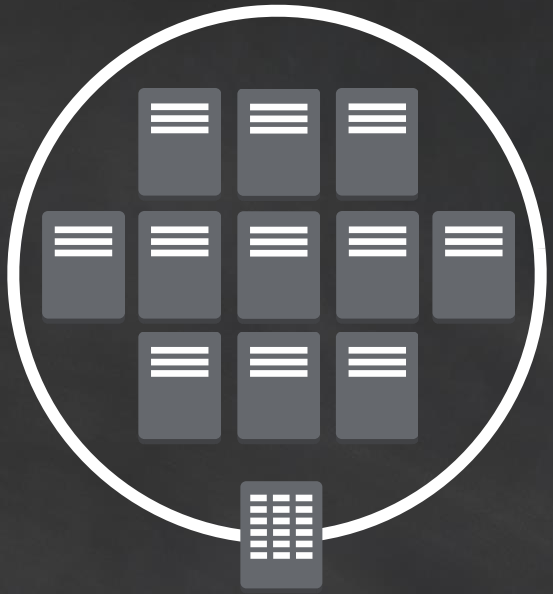
- 長い待ち時間
- スペックミスマッチ
- コア数不足

各自で計算機を持つと・・・

- 予算・調達・構築
- 場所・電源・空調・騒音
- 運用管理

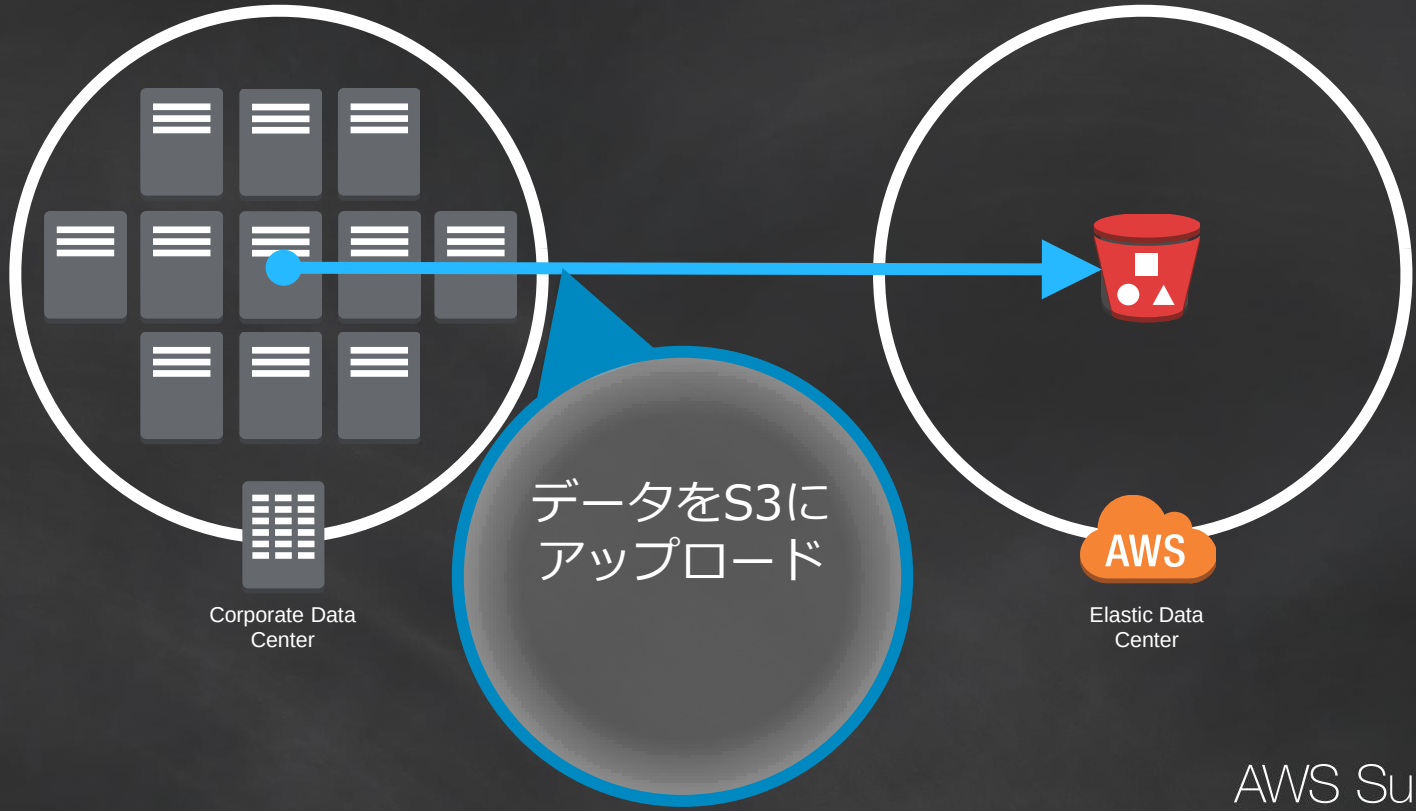
AWSなら

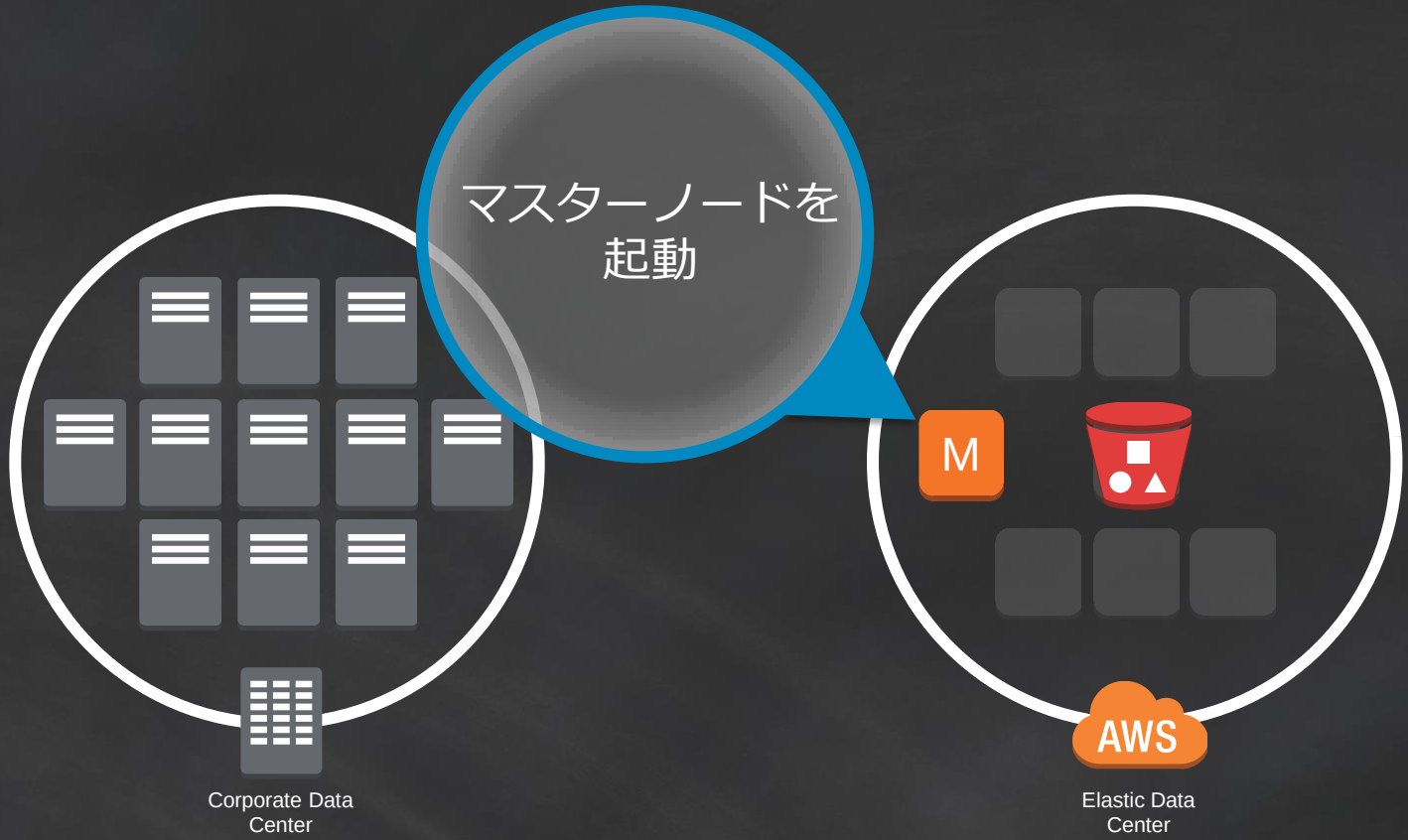
気軽に
待たずに
必要な時に
必要なだけのコアで

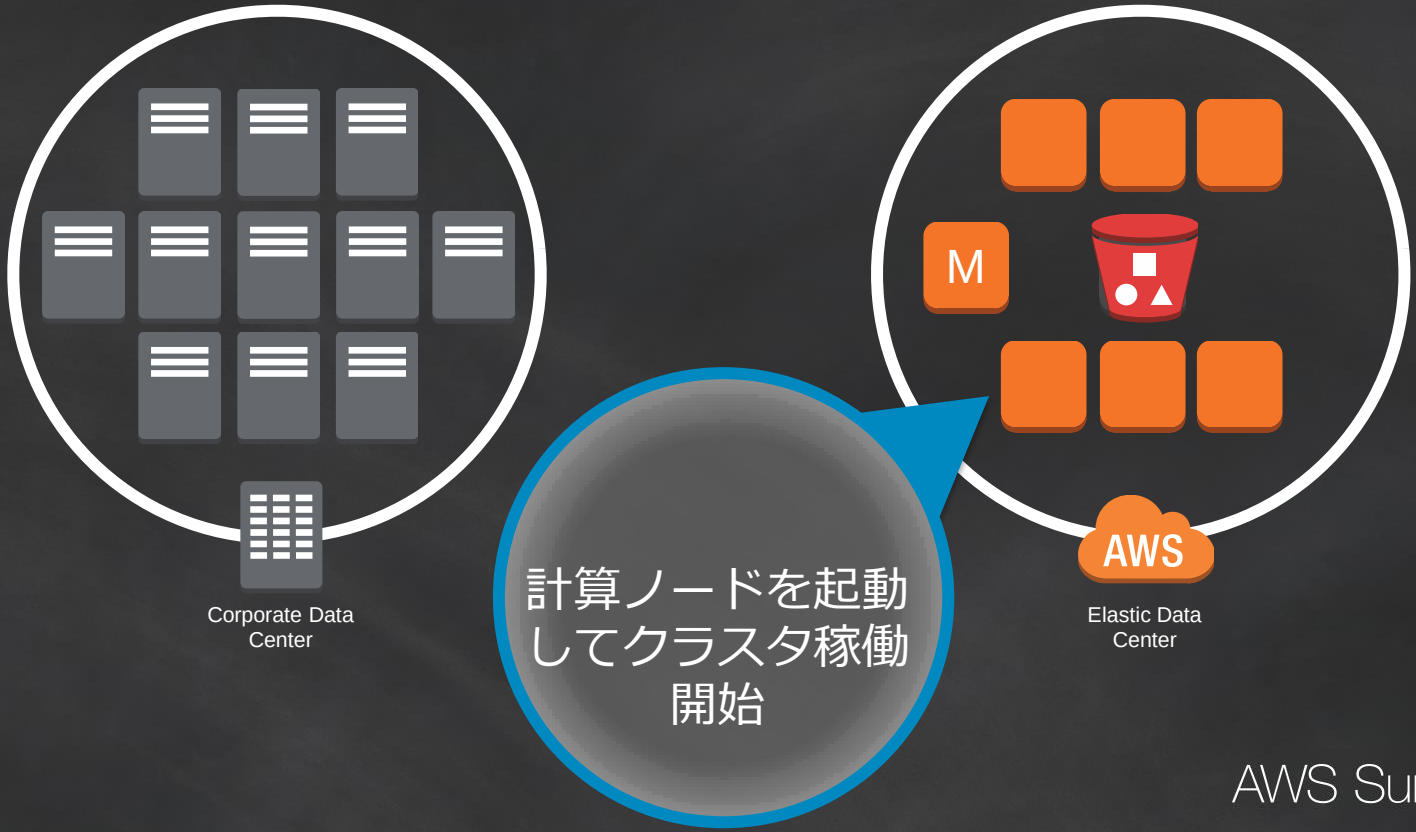


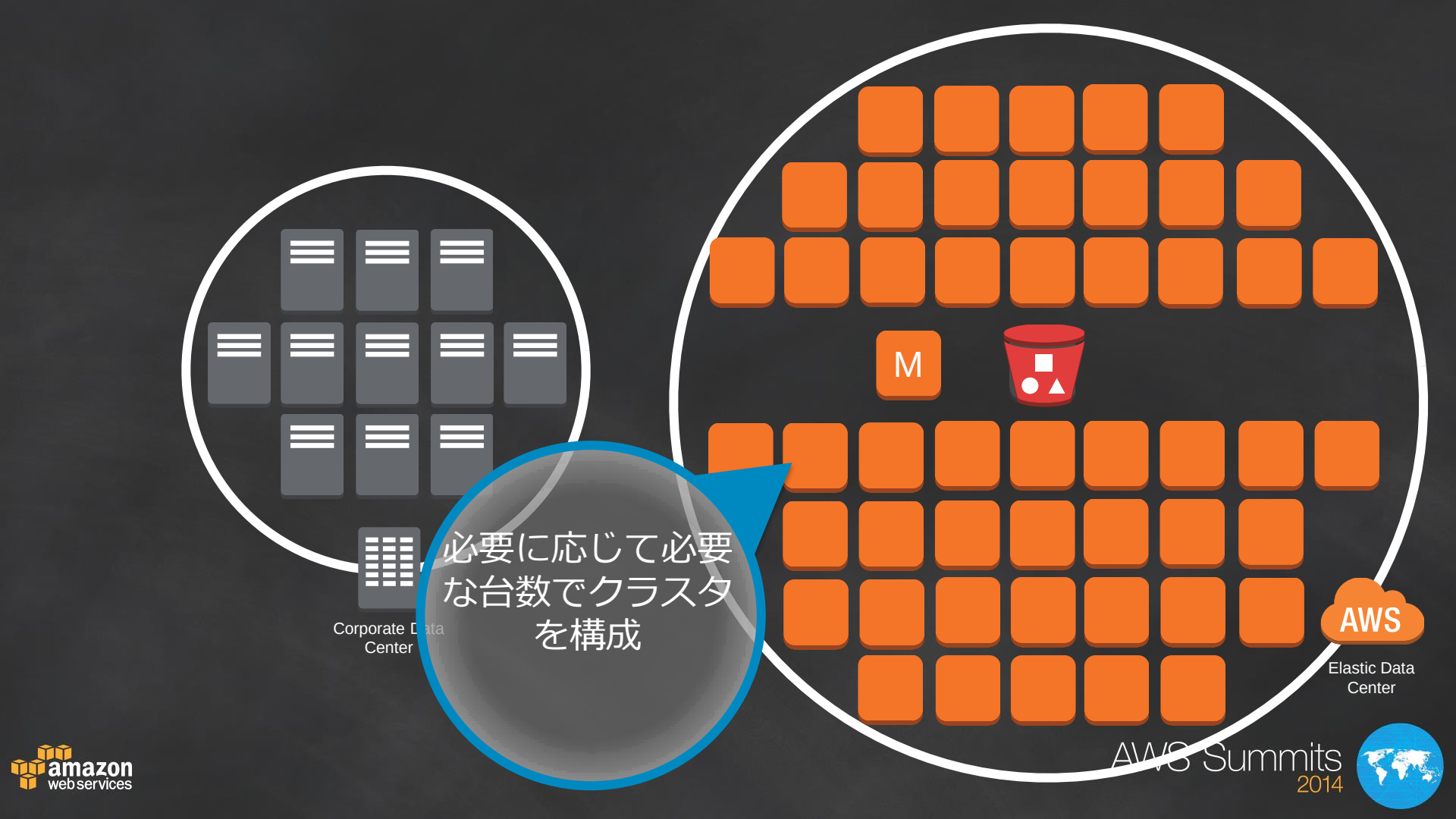
Corporate Data Center

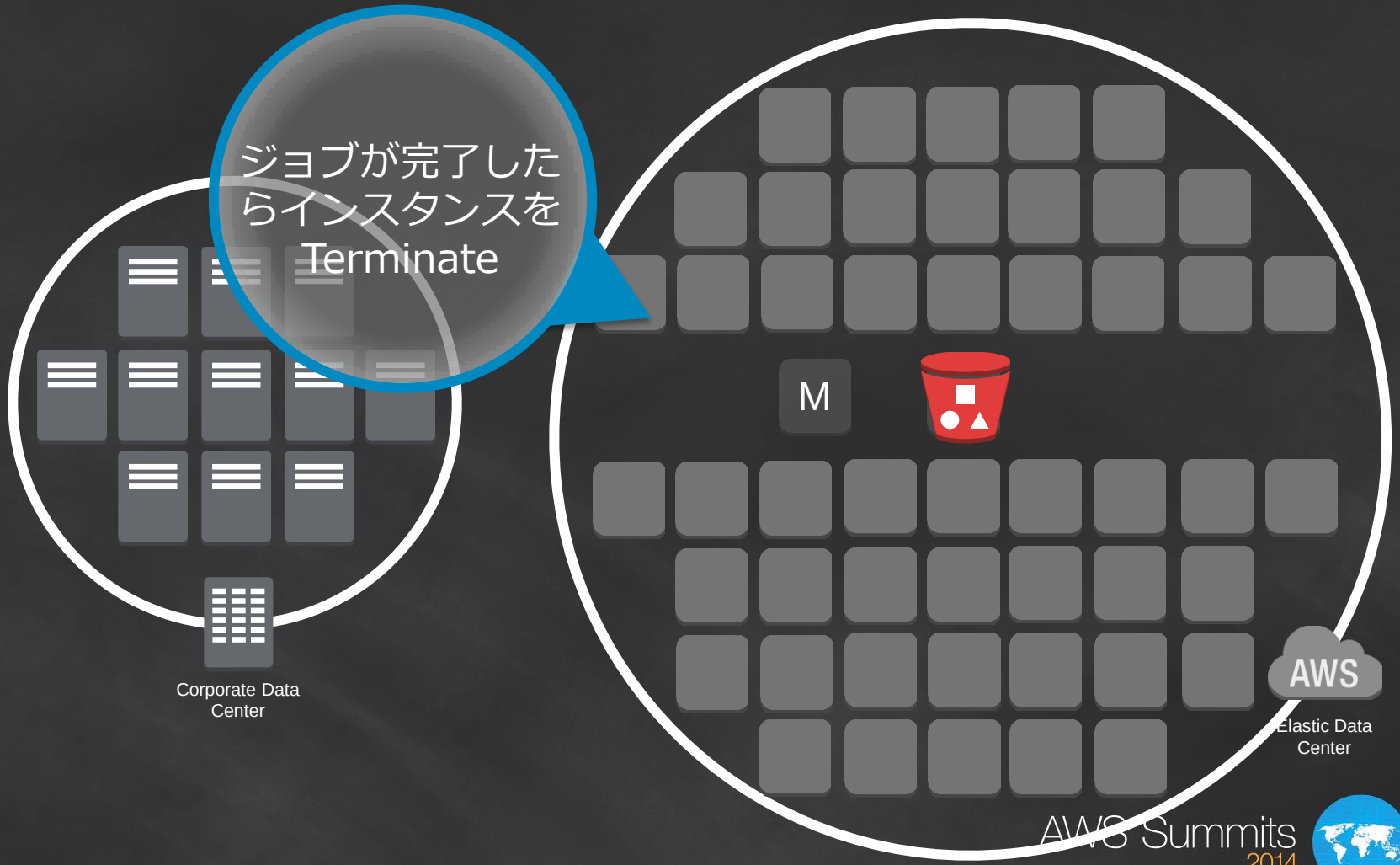
Elastic Data Center

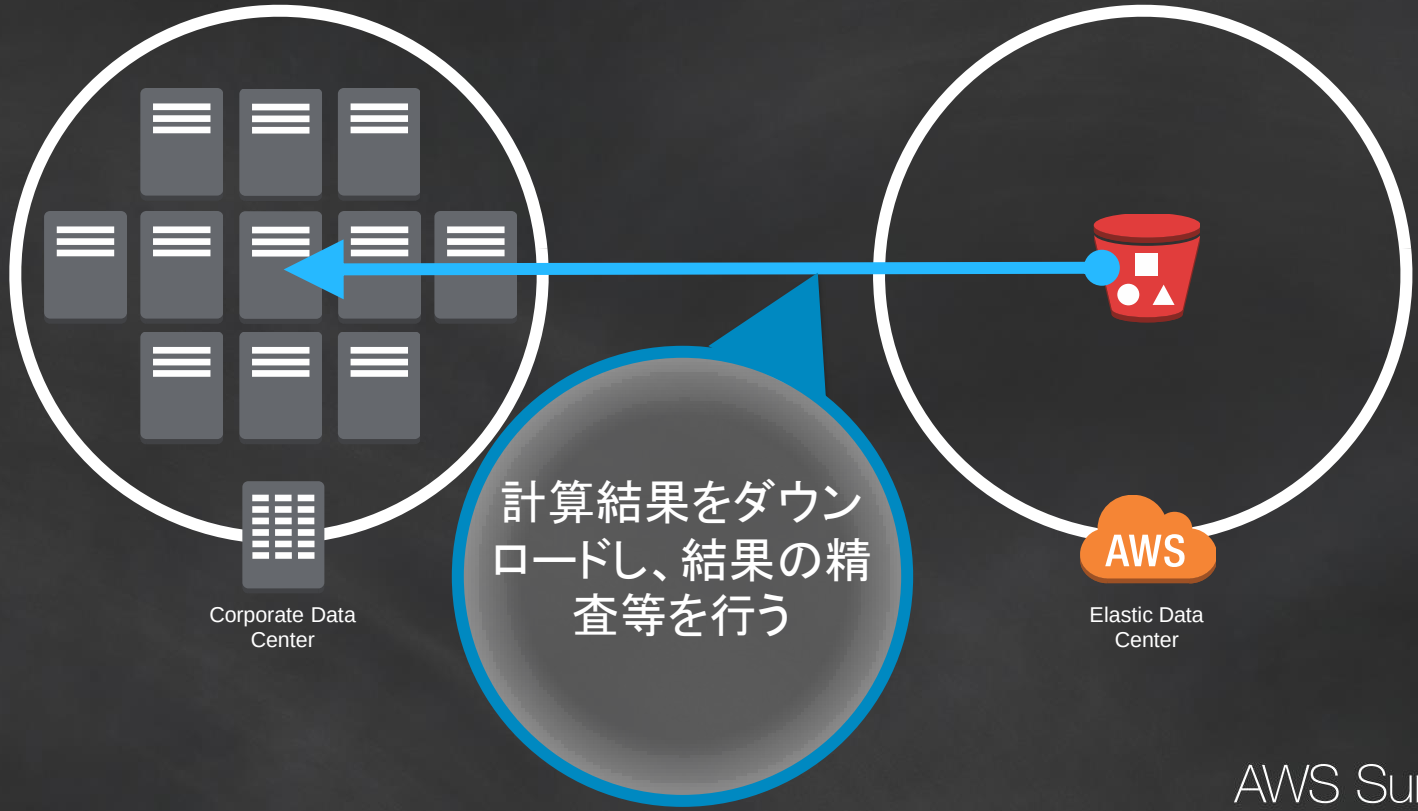

データをS3に
アップロード
Corporate Data
Center
AWS
Elastic Data
Center


マスターノードを
起動
M
AWS
Corporate Data
Center
Elastic Data
Center


M
AWS
Corporate Data Center
計算ノードを起動してクラスタ稼働開始
Elastic Data Center


M
AWS
Elastic Data Center
Corporate Data Center
必要に応じて必要な台数でクラスタを構成


ジョブが完了したらインスタンスをTerminate
M
Corporate Data Center
AWS
Elastic Data Center


計算結果をダウンロードし、結果の精査等を行う
Corporate Data Center
AWS
Elastic Data Center

```ruby
  # generate a key pair
  key_pair = ec2.key_pairs.create("ruby-sample-#{Time.now.to_i}")
  puts "Generated keypair #{key_pair.name}, fingerprint: #{key_pair.fingerprint}"

  # open SSH access
  group = ec2.security_groups.create("ruby-sample-#{Time.now.to_i}")
  group.authorize_ingress(:tcp, 22, "0.0.0.0/0")
  puts "Using security group: #{group.name}"

  # launch the instance
  instance = image.run_instance(:key_pair => key_pair,
                                :security_groups => group)
  sleep 1 until instance.status != :pending
  puts "Launched instance #{instance.id}, status: #{instance.status}"

  exit 1 unless instance.status == :running

  begin
    Net::SSH.start(instance.ip_address, "ec2-user",
                   :key_data => [key_pair.private_key]) do |ssh|
      puts "Running 'uname -a' on the instance yields:"
      puts ssh.exec!("uname -a")
    end
  rescue SystemCallError, Timeout::Error => e
    # port 22 might not be available immediately after the instance finishes launching
    sleep 1
    retry
  end

ensure
  # clean up
  [instance,
   group,
   key_pair].compact.each(&:delete)
end
```

プログラムで操作可能

```
# open SSH access
group = ec2.security_groups.create("ruby-sample-#{Time.now.to_i}")
group.authorize_ingress(:tcp, 22, "0.0.0.0/0")
puts "Using security group: #{group.name}"

# launch the instance
instance = image.run_instance(:key_pair => key_pair,
                              :security_groups => group)
sleep 1 until instance.status != :pending
puts "Launched instance #{instance.id}, status: #{instance.status}"

exit 1 unless instance.status == :running

# port 22 might not be available immediately after the instance finishes launching
sleep 1
```

AWS APIを使ったToolkit
MIT Starcluster だと

$ starcluster start -s 16 samplecluster

$ starcluster start -s 16 samplecluster

StarCluster - (http://web.mit.edu/starcluster) (v. 0.93.3)
Software Tools for Academics and Researchers (STAR)
Please submit bug reports to starcluster@mit.edu

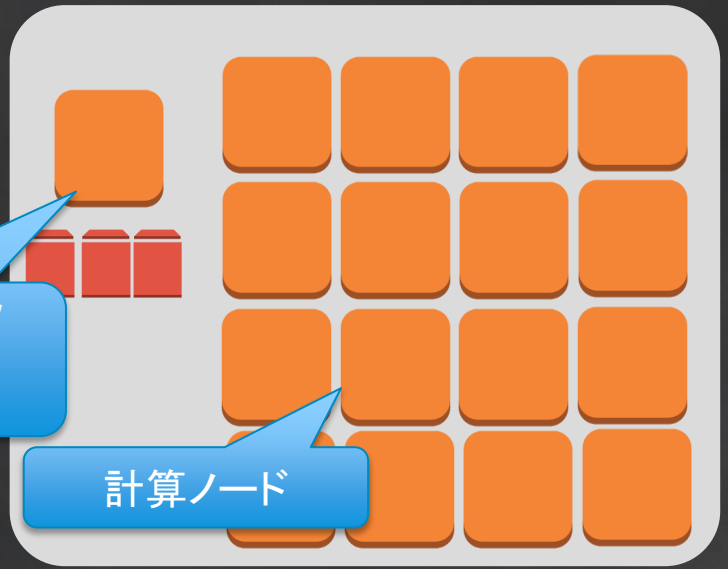
>>> Using default cluster template: smallcluster
>>> Validating cluster template settings...
>>> Cluster template settings are valid
>>> Starting cluster...
>>> Launching a 16-node cluster...

>>> Waiting for cluster to come up... (updating every 30s)
20/20 |||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||| 100%

>>> Configuring SGE...
>>> Configuring NFS exports path(s):
/opt/sge6
>>> Mounting all NFS export path(s) on 16 worker node(s)
16/16 |||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||||| 100%
>>> Setting up NFS took 0.198 mins

Name	Instance	AMI ID	Root Devic	Zone	Type	State	Status Che	Alarm Stat	Monitoring	Security G	Ke
master	i-3ece58	ami-5766ee	ebs	ap-northeas	cc2.8xlarge	running	2/2 che	none	basic	@sc-democ	id_
node001	i-3cce58	ami-5766ee	ebs	ap-northeas	cc2.8xlarge	running	2/2 che	none	basic	@sc-democ	id_
node002	i-42ce58	ami-5766ee	ebs	ap-northeas	cc2.8xlarge	running	2/2 che	none	basic	@sc-democ	id_
node003	i-40ce58	ami-5766ee	ebs	ap-northeas	cc2.8xlarge	running	2/2 che	none	basic	@sc-democ	id_
node004	i-46ce58	ami-5766ee	ebs	ap-northeas	cc2.8xlarge	running	2/2 che	none	basic	@sc-democ	id_
node005	i-44ce58	ami-5766ee	ebs	ap-northeas	cc2.8xlarge	running	2/2 che	none	basic	@sc-democ	id_
node006	i-4ace58	ami-5766ee	ebs	ap-northeas	cc2.8xlarge	running	2/2 che	none	basic	@sc-democ	id_
node007	i-48ce58	ami-5766ee	ebs	ap-northeas	cc2.8xlarge	running	2/2 che	none	basic	@sc-democ	id_
node008	i-4ece58	ami-5766ee	ebs	ap-northeas	cc2.8xlarge	running	2/2 che	none	basic	@sc-democ	id_
node009	i-4cce58	ami-5766ee	ebs	ap-northeas	cc2.8xlarge	running	2/2 che	none	basic	@sc-democ	id_
node010	i-52ce58	ami-5766ee	ebs	ap-northeas	cc2.8xlarge	running	2/2 che	none	basic	@sc-democ	id_
node011	i-50ce58	ami-5766ee	ebs	ap-northeas	cc2.8xlarge	running	2/2 che	none	basic	@sc-democ	id_
node012	i-56ce58	ami-5766ee	ebs	ap-northeas	cc2.8xlarge	running	2/2 che	none	basic	@sc-democ	id_
node013	i-54ce58	ami-5766ee	ebs	ap-northeas	cc2.8xlarge	running	2/2 che	none	basic	@sc-democ	id_

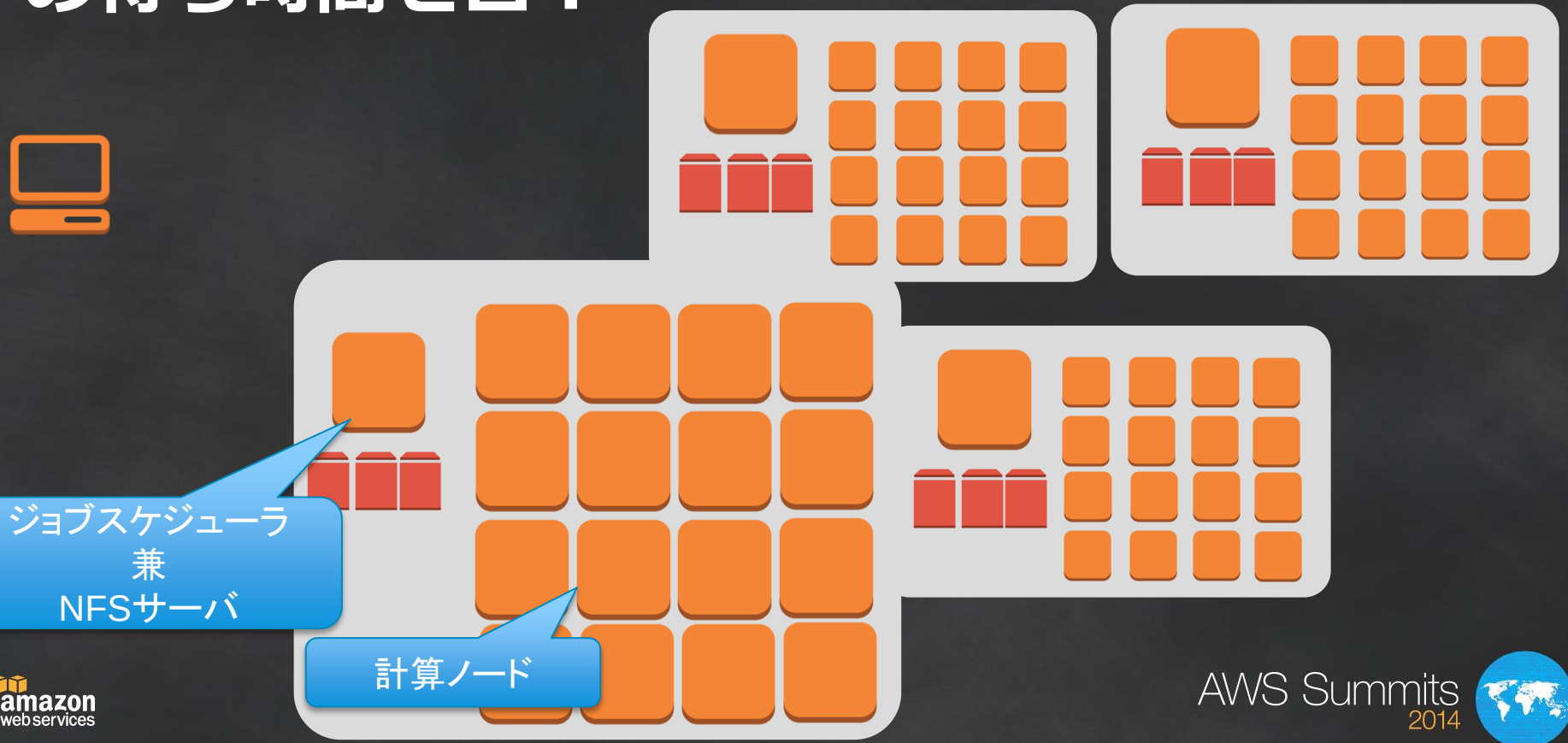
コマンド1つでこの構成が！



```
sgeadmin@master:~$ qhost
HOSTNAME                ARCH         NCPU  LOAD  MEMTOT  MEMUSE  SWAPTO  SWAPUS
----------------------------------------------------------------------------------
global                  -               -     -       -       -       -       -
master                  linux-x64      32  0.01   58.5G  737.2M     0.0     0.0
node001                 linux-x64      32  0.01   58.5G  720.2M     0.0     0.0
node002                 linux-x64      32  0.01   58.5G  718.9M     0.0     0.0
node003                 linux-x64      32  0.01   58.5G  719.6M     0.0     0.0
node004                 linux-x64      32  0.01   58.5G  719.9M     0.0     0.0
node005                 linux-x64      32  0.01   58.5G  718.5M     0.0     0.0
node006                 linux-x64      32  0.01   58.5G  720.8M     0.0     0.0
node007                 linux-x64      32  0.02   58.5G  720.6M     0.0     0.0
node008                 linux-x64      32  0.01   58.5G  717.8M     0.0     0.0
node009                 linux-x64      32  0.01   58.5G  717.3M     0.0     0.0
node010                 linux-x64      32  0.01   58.5G  718.9M     0.0     0.0
node011                 linux-x64      32  0.01   58.5G  717.9M     0.0     0.0
node012                 linux-x64      32  0.02   58.5G  717.0M     0.0     0.0
node013                 linux-x64      32  0.01   58.5G  719.1M     0.0     0.0
node014                 linux-x64      32  0.01   58.5G  719.2M     0.0     0.0
node015                 linux-x64      32  0.01   58.5G  718.3M     0.0     0.0
```

```
$ starcluster start -s 16  cluster1
$ starcluster start -s 16  cluster2
$ starcluster start -s 16  cluster3
```

ジョブごとにクラスタを用意すればジョブの待ち時間ゼロ！



```
$ starcluster terminate cluster1
$ starcluster terminate cluster2
$ starcluster terminate cluster3
$ starcluster terminate samplecluster
```

ジョブが終われば、クラスタを削除してコスト削減

Agenda

- AWSとは？
- なぜHPC on AWSなのか？
- クラスタインスタンスの性能
- お客様事例

AWS Summit Tokyo 2013 （ 2013/6/5)

ついにあのインスタンスタイプが東京に

クラスタコンピュート
cc2.8xlarge
60.5GB RAM
3.4TB

クラスタハイメモリ
cr1.8xlarge
144GB RAM
2x120GB SSD

Intel® Xeon® E5-2670 x 2, 16コア
10Gbps full bi-section

AWS Summit Tokyo 2013 （ 2013/6/5)

ついにあのインスタンスタイプが東京に

クラスタコン
cc2.8xlarge
60.5GB RAM
3.4TB

事例

CC2 インスタンスクラスタ
1026ノード(17024コア)

240 TFLOPS

$2554/hour

TOP500 SUPERCOMPUTER SITES

List	Rank	System	Vendors	Cores	Rmax (GFlop/s)	Rpeak (GFlop/s)
11/2011	42	Amazon EC2 Cluster Compute Instances - Amazon EC2 Cluster, Xeon 8C 2.60GHz, 10G Ethernet	Self-made	17024	240090	354099.2

amazon web services

インスタンスタイプの歴史

AWSを開始した2006年より、様々な用途に応じた インスタンスタイプを随時追加し、利用可能

(2014年7月18日時点で37タイプ)

今後も新しいインスタンスタイプを追加予定

2013年11月 最新・高速インスタンス登場

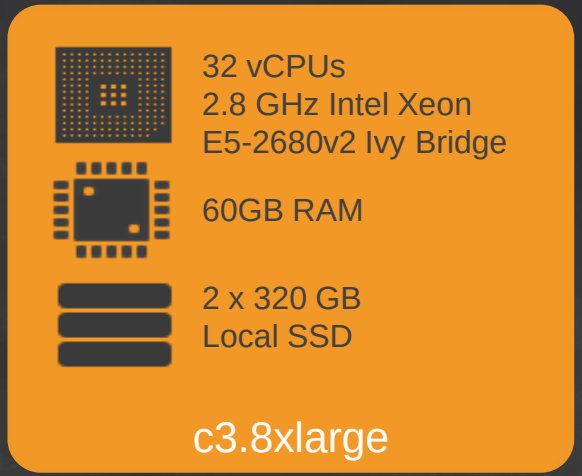
C3.8xlarge インスタンス

Intel® Xeon® E5-2680v2 Ivy Bridge

I/O Performance: Very High (10 Gigabit Ethernet)

Enhanced Networking (SR-IOV)

Intel® Turbo Boost Technology



高性能インスタンスの変遷

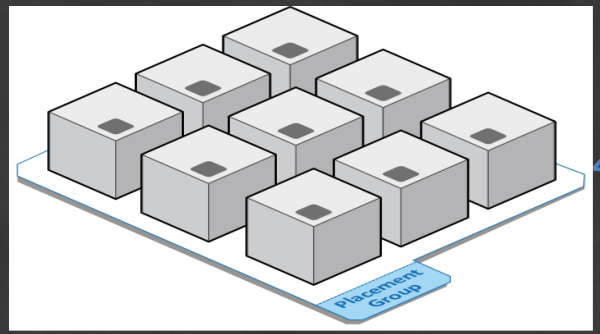
	CC1	CC2.8xlarge	C3.8xlarge
vCPU	16	32	32
RAM (GiB)	23	60.5	60
CPU	Xeon X5570 (Nehalem)	Xeon E5-2670 (Sandy Bridge)	Xeon E5-2680v2 (Ivy Bridge)
NIC	10Gbps	10Gbps	10Gbps(SR-IOV)
Launch Date	Jul, 2010	Nov, 2011	Nov, 2013

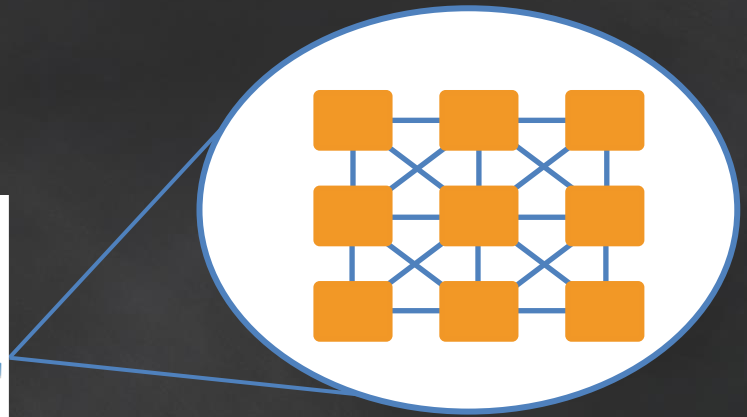
$2.000/hour　　$1.680/hour

※2014年7月18日時点の US-EAST Linuxの価格

10GbE クラスタネットワーク + Enhanced Network

- Full bisection 10Gbps
- 低レイテンシ、低ジッター
- プレイスメントグループ内にインスタンスを配備
- SR-IOV対応インスタンスはさらなる低レイテンシを実現





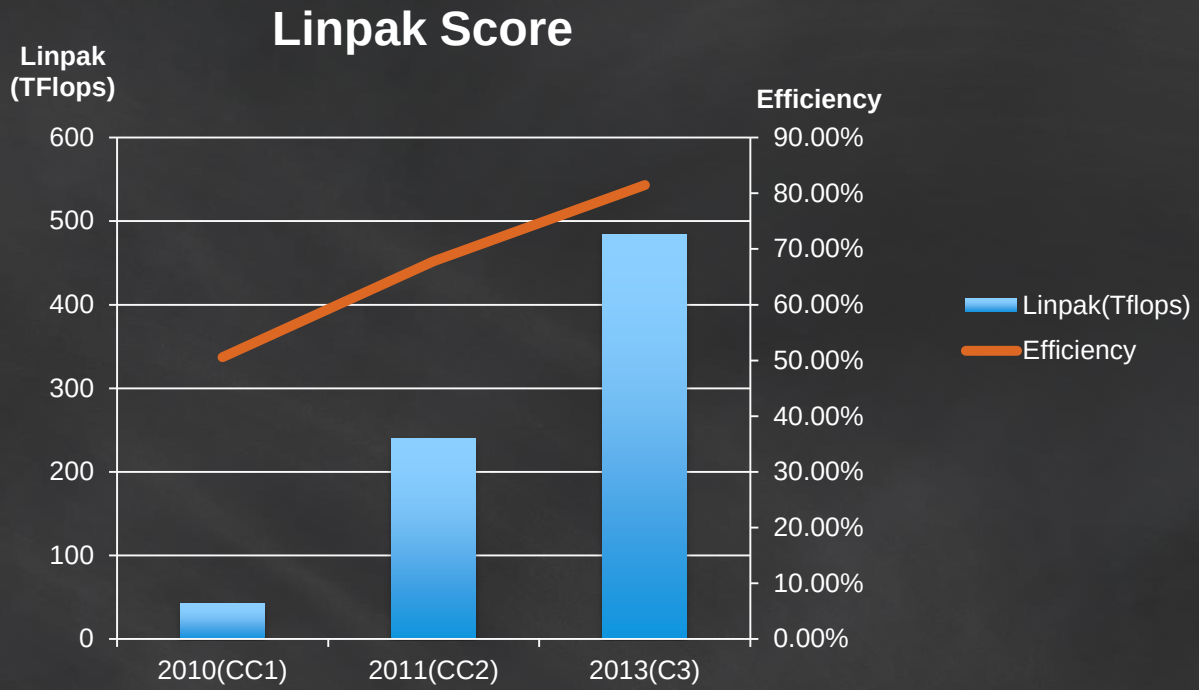
Amazon EC2
c3.8xlarge インスタンスクラスタ
1,656ノード(26,496コア)

484.2 TFLOPS

TOP500 64位 (Nov 2013)
一時間当たり約29万円から

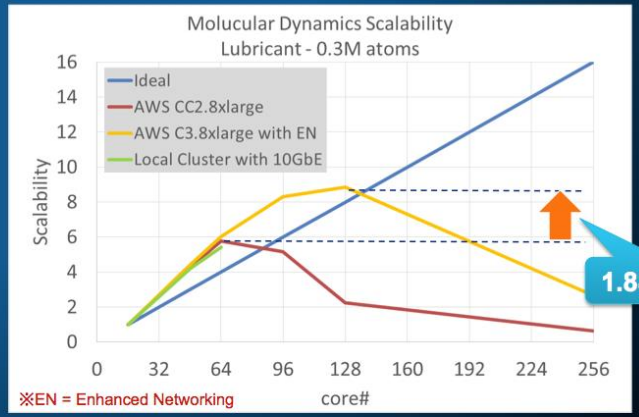
List	Rank	System	Vendor	Total Cores	Rmax (TFlops)	Rpeak (TFlops)	Power (kW)
11/2013	64	Amazon EC2 Cluster, Intel Xeon E5-2670v2 10C 2.500GHz, 10G Ethernet	Self-made	26,496	484.2	593.9	

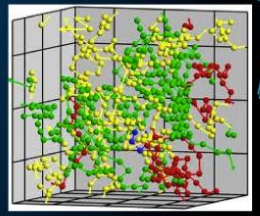
Top500 性能の変遷



実際の計測結果

Molecular Dynamics





1.88x faster

- CC2 & C3 cluster have equivalent scalability of HGST local cluster with 10GbE around 64cores
- C3 provide significant improvement to the scalability
- C3 is ***1.88x faster*** than CC2

HGST

 9

64コアまではオンプレと同等性能

C3では128コアまでスケール

CC2とC3の性能差は 1.88倍

GPU インスタンス

CG1 インスタンス

Intel® Xeon® X5570 processors

2 x NVIDIA Tesla “Fermi” M2050 GPUs

I/O Performance: Very High (10 Gigabit Ethernet)

33.5 EC2 Compute Units

20GB RAM

2x NVIDIA GPU
448 Cores
3GB Mem

cg1.8xlarge

G2

Intel® Intel Xeon E5-2670

1 NVIDIA Kepler GK104 GPU

I/O Performance: Very High (10 Gigabit Ethernet)

26 EC2 Compute Units

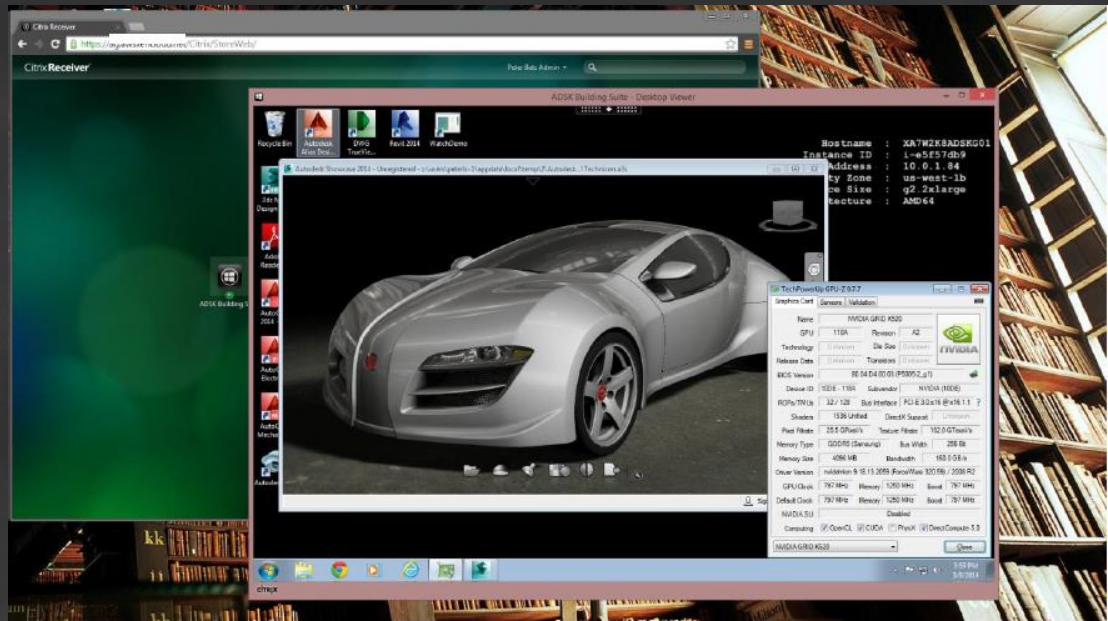
16GB RAM

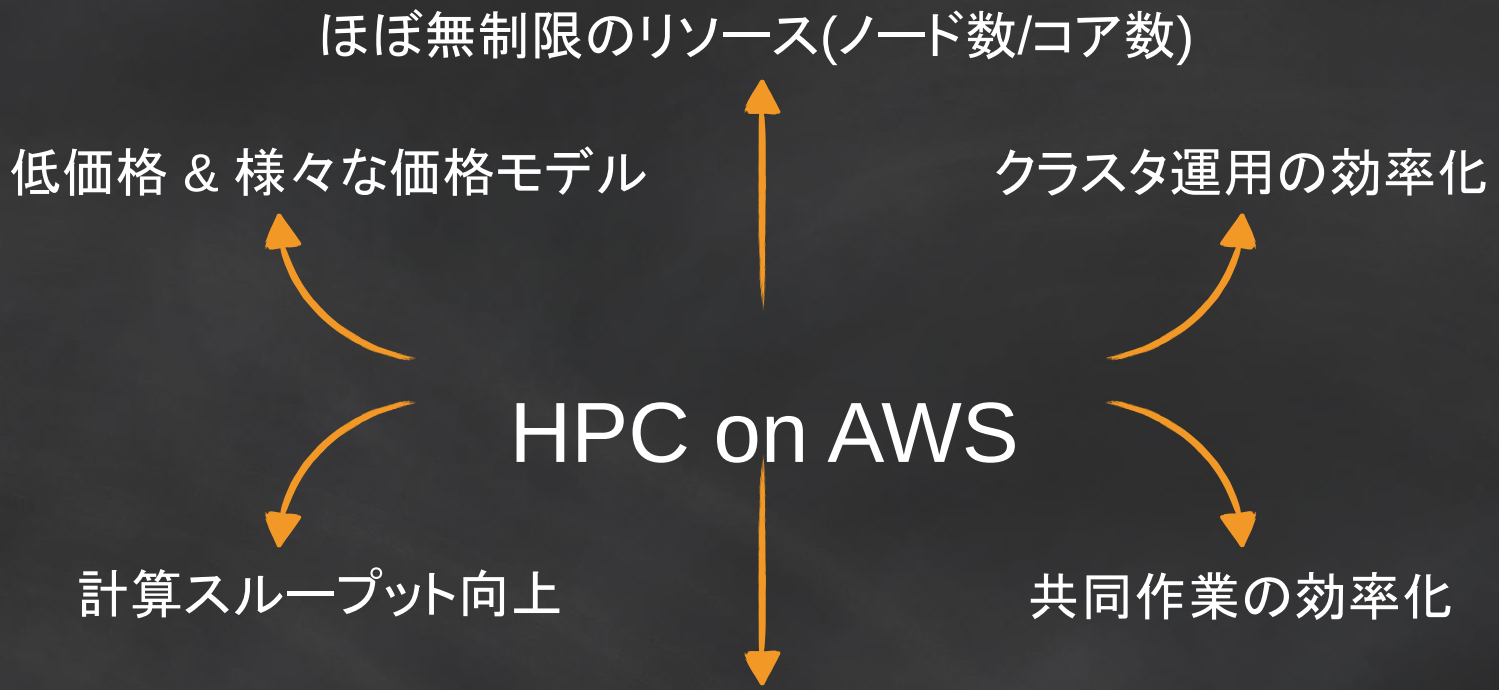
1x NVIDIA GPU
1536 Cores
4GB Mem

g2.2xlarge

G2 + Citrix XenAppによるリモートグラフィックス

- g2.2xlargeではHDX 3D Proを有効にしてサーバサイドで3Dアプリケーションの実行が可能
  - 3D CAD
  - プリポスト処理
  - アニメーション生成
  - 医療用画像処理



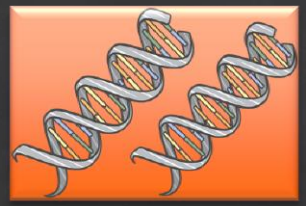

ほぼ無制限のリソース(ノード数/コア数)
低価格 & 様々な価格モデル
クラスタ運用の効率化
HPC on AWS
計算スループット向上
共同作業の効率化
需要に応じて複数クラスタ起動

Agenda

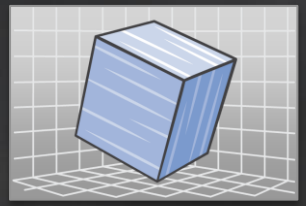
- AWSとは？
- なぜHPC on AWSなのか？
- クラスタインスタンスの性能
- お客様事例

主要なHPCアプリケーション

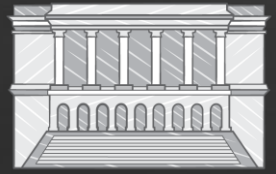
ゲノム解析



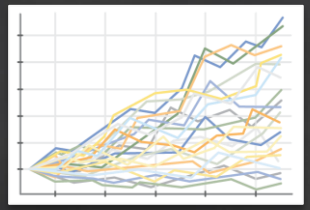
モデリング
シミュレーション



教育機関・政府機関

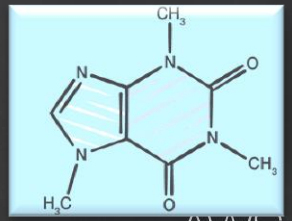


モンテカルロ
シミュレーション

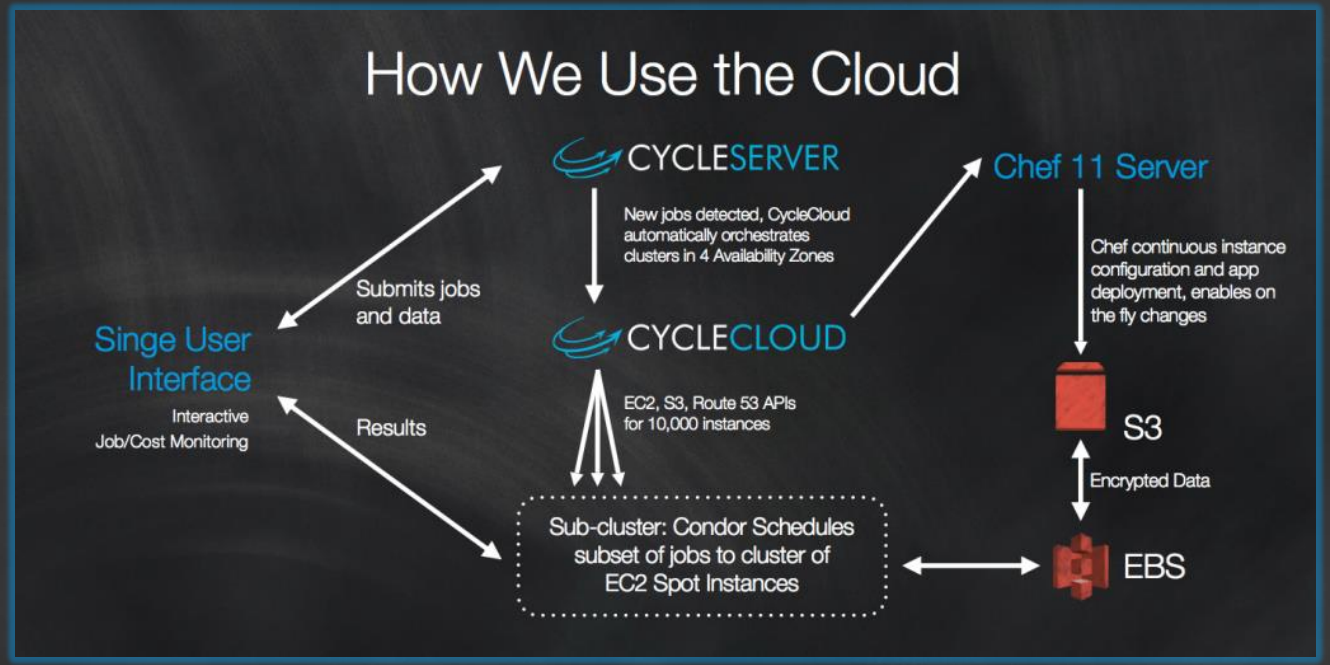


トランスコーディング
エンコーディング

計算化学



Novartis Institutes for Biomedical Research

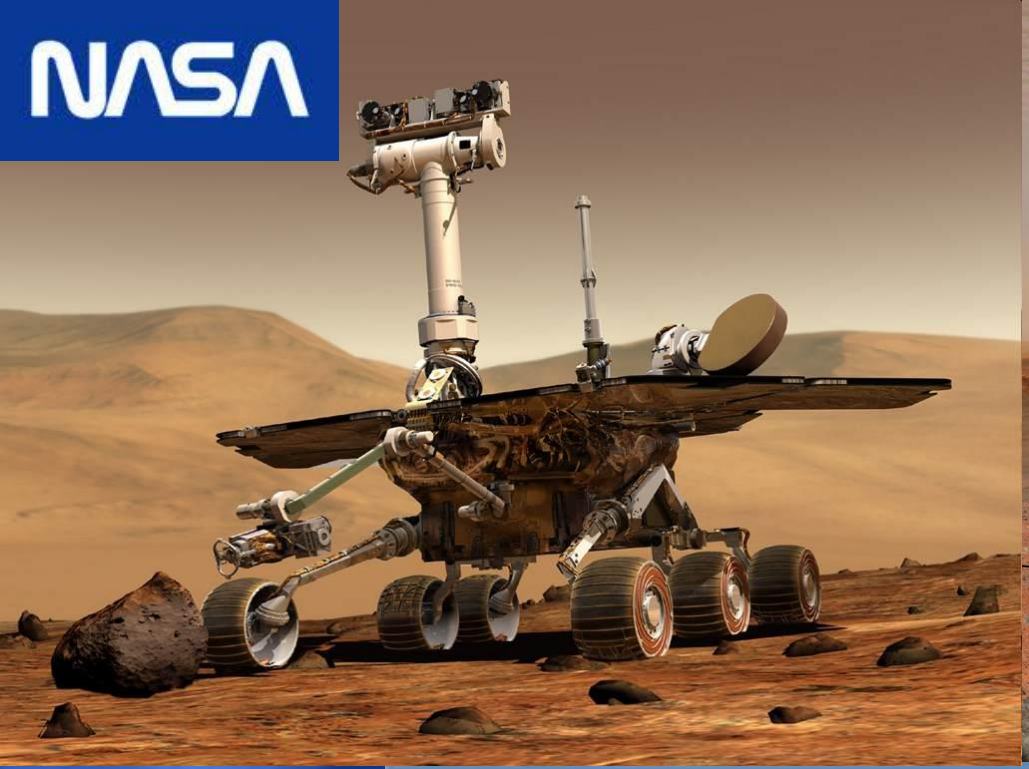


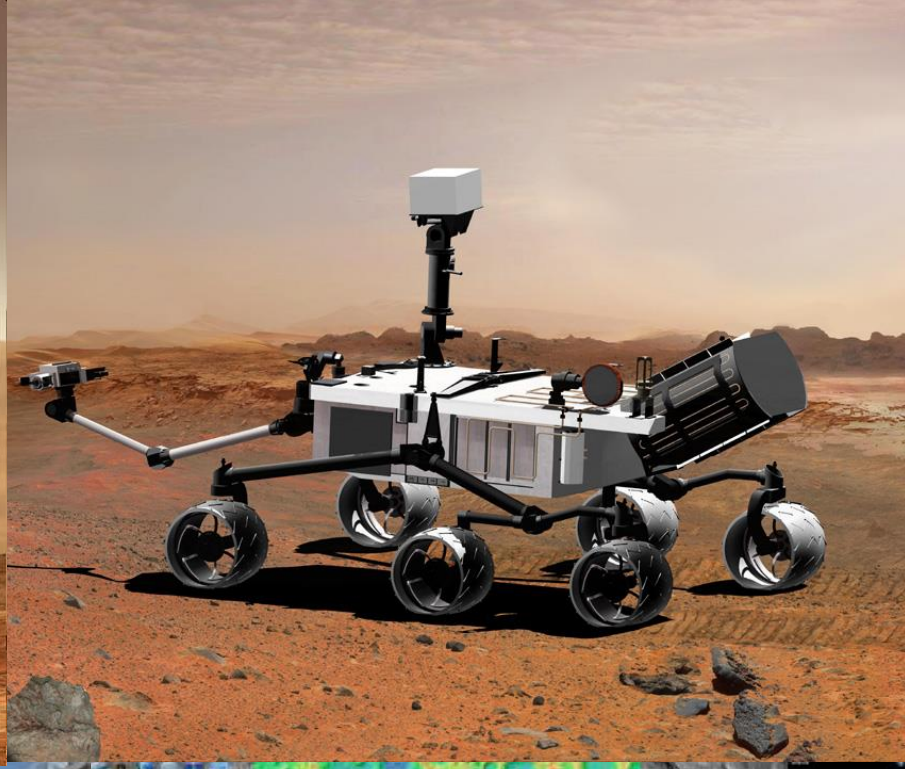
$44M 相当のスパコンを
$5K で実現

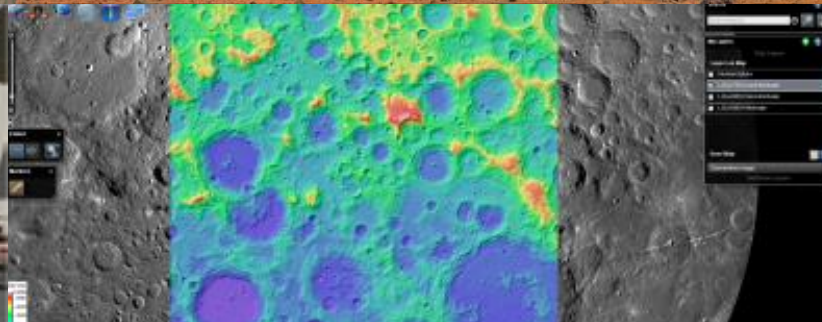
10,000台をSpotで起動

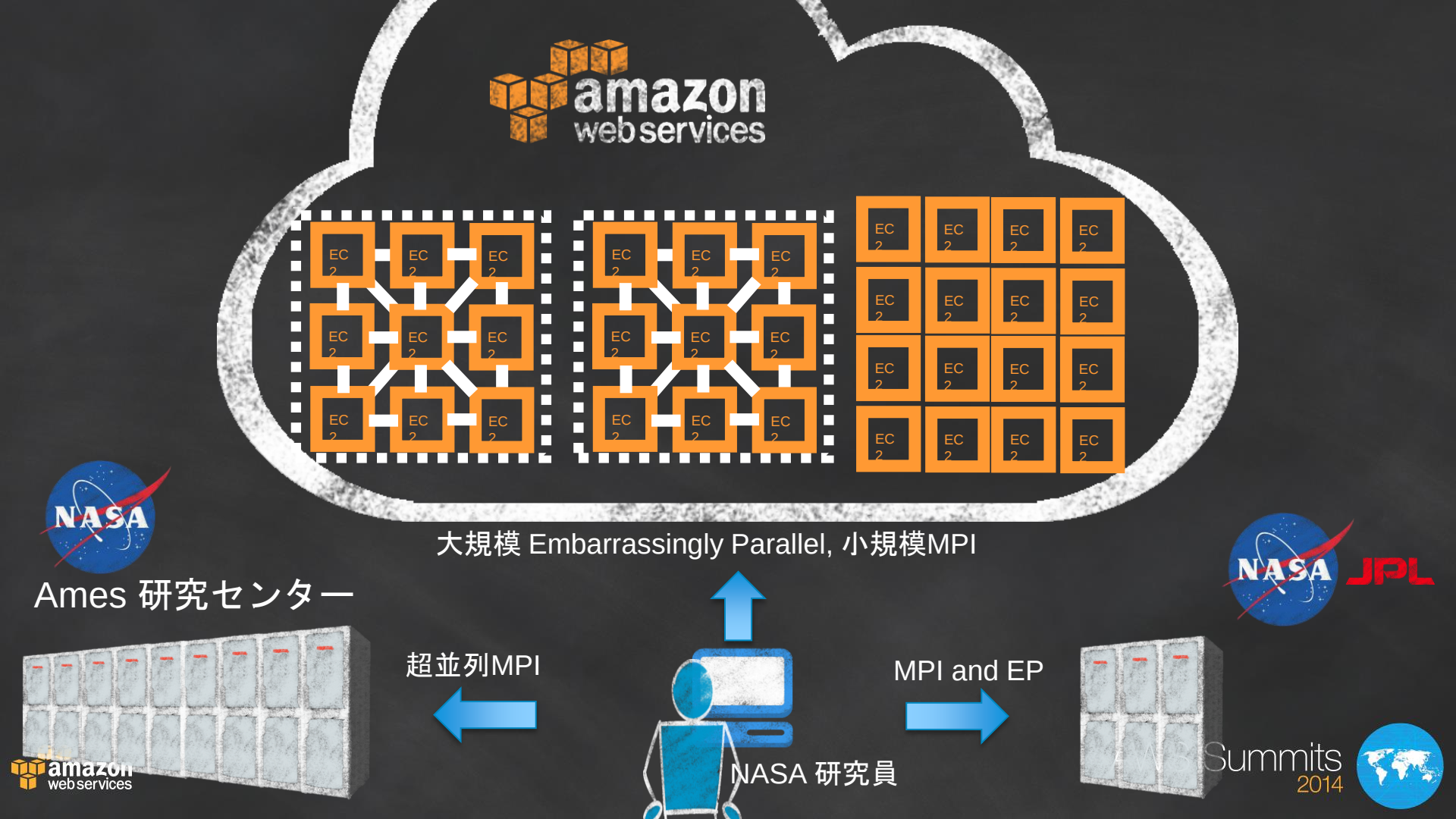
39年かかる計算を
11時間で完了

NASA








amazon web services
EC2
大規模 Embarrassingly Parallel, 小規模MPI
NASA
Ames 研究センター
NASA JPL
超並列MPI
MPI and EP
NASA 研究員
amazon web services
AWS Summits 2014


Park
Friday, April 17th 2009 4:38 PM
INTERNET
NAVIGATION
CAR
BACKUP
PHONE
POWER
RADIO
MEDIA
STREAMING
Fake Plastic Tr...
Radiohead
The Bends
RADIOHEAD-FAKE PLA
RADIOHEAD-STOP WHISPERING
RADIOHEAD-BONES
RADIOHEAD-I MIGHT BE WRONG
1:18
-3:31
mute
<<
Play
>>
shuffle
Playlists
Artists
Songs
Outside Temp: 82
Off
74
75
76
70
71
72
Off

CYPRESS PERFORM

简体中文 | 日本語 | 한국어 | About Us | Cont

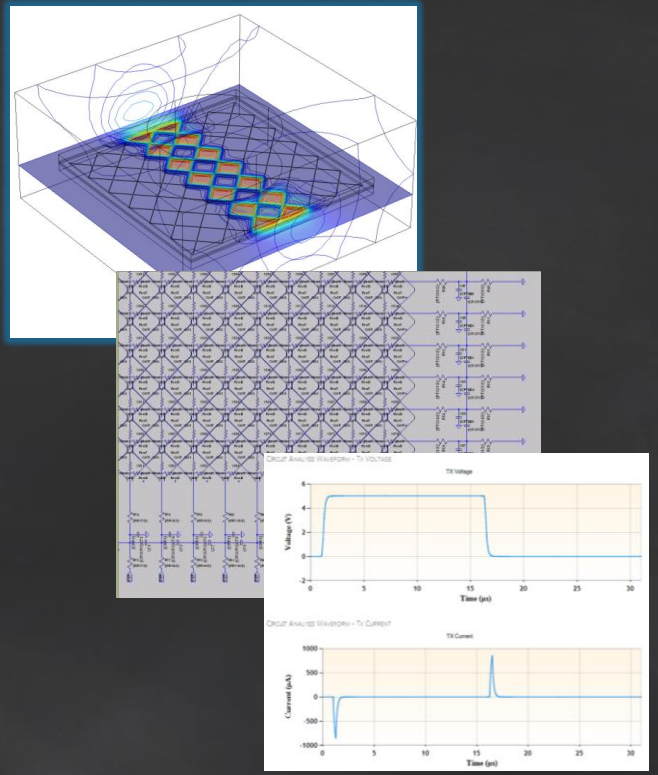
Products | Applications | Support & Community | Buy | Part Search

Home > Products > Touch Sensing

Touch Sensing

CapSense® Controllers

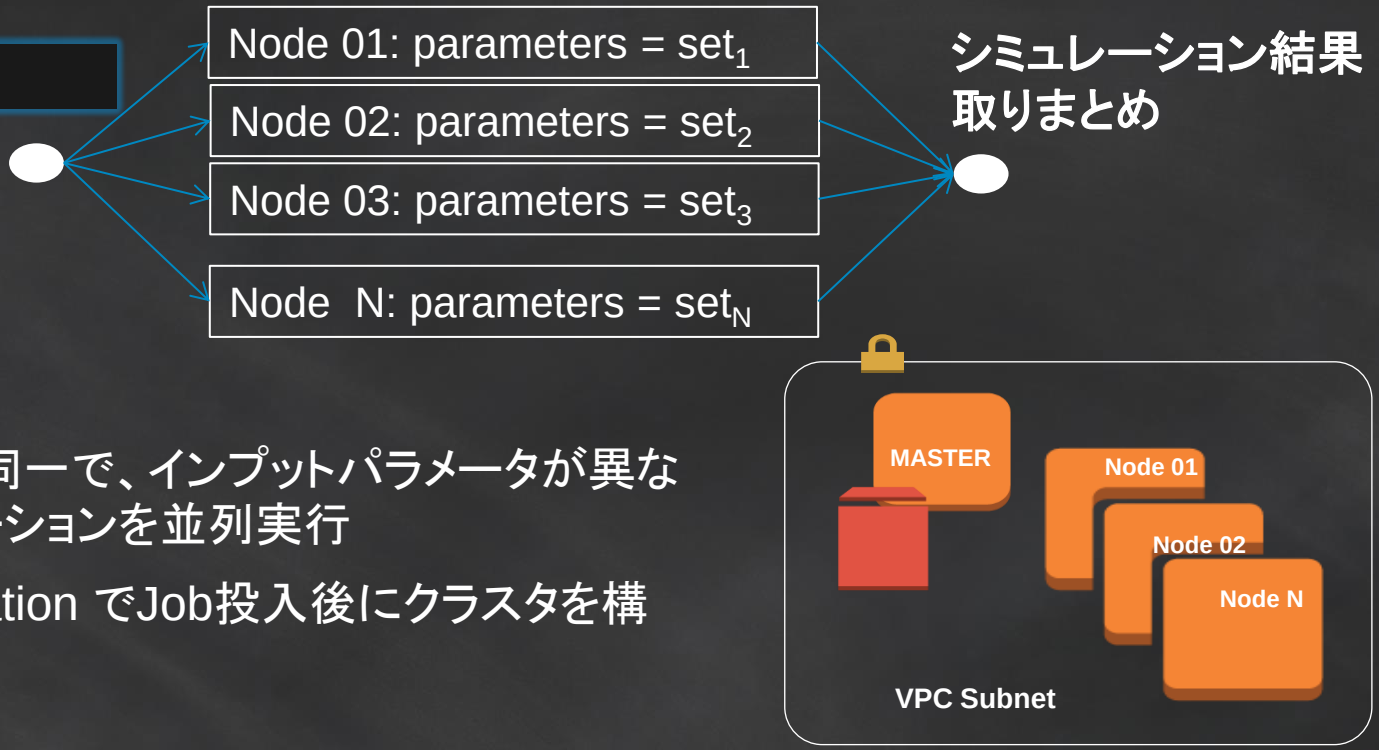
The CapSense Family is Cypress's market leading Capacitive Sensing solution that has replaced more than 3.5 Billion sensors with a Capacitive touch interface over the last many years. The CapSense Family is based on Cypress's Programmable System on Chip (PSoC) platform that provides customers with the flexibility needed to make last minute changes and get to market before the competition. Cypress's latest addition to this portfolio is



EM シミュレーションのスケーラビリティ

Job 投入

```
$ bsub -J "CyArray[1-N]"
```

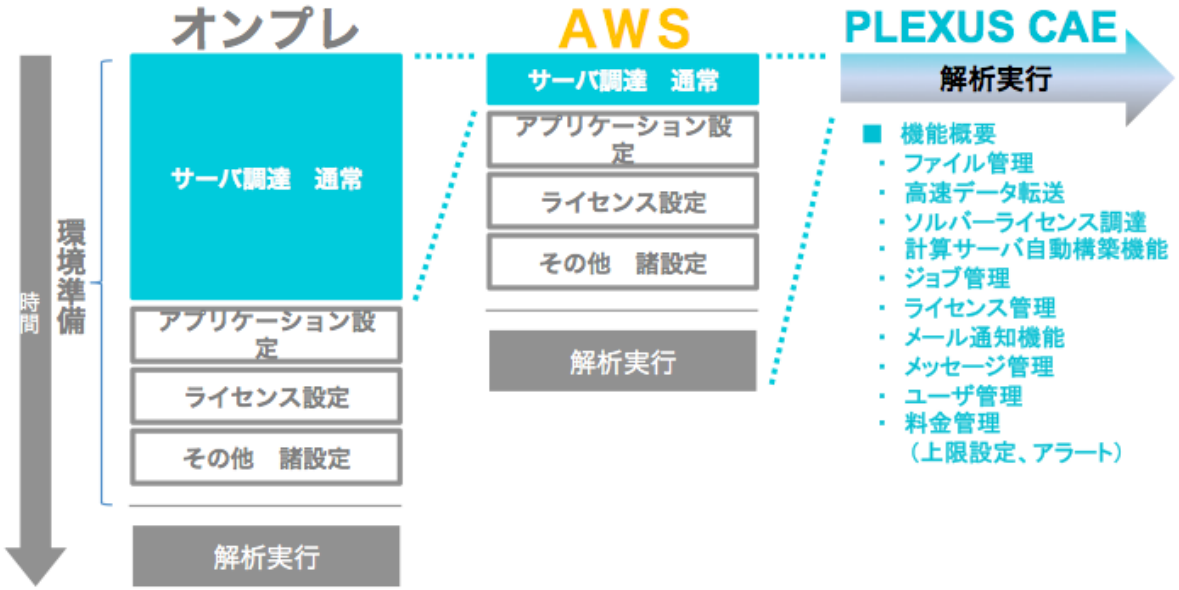


- 実行プログラムは同一で、インプットパラメータが異なる複数のシミュレーションを並列実行
- AWS CloudFormation でJob投入後にクラスタを構築し並列実行

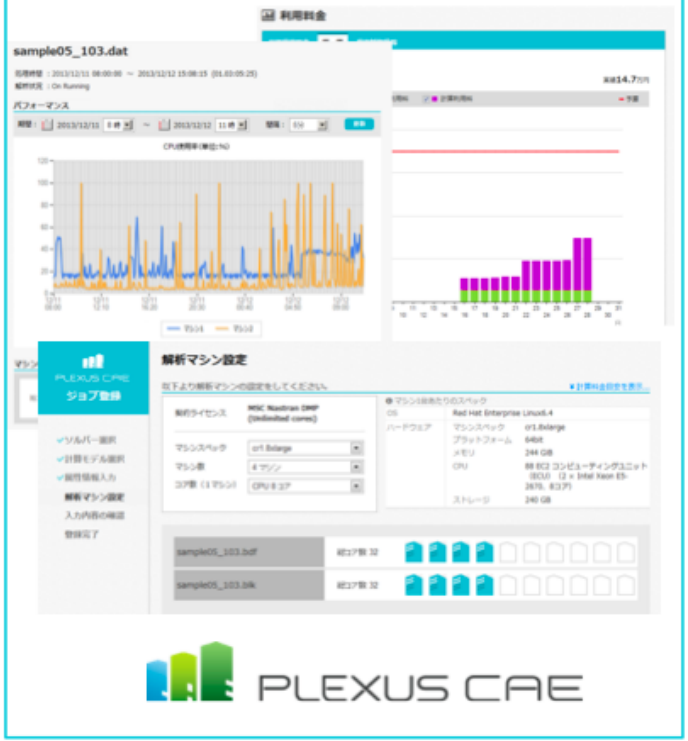
HPC パートナーと対応アプリケーション

【ISID】 PLEXUS CAE ご紹介

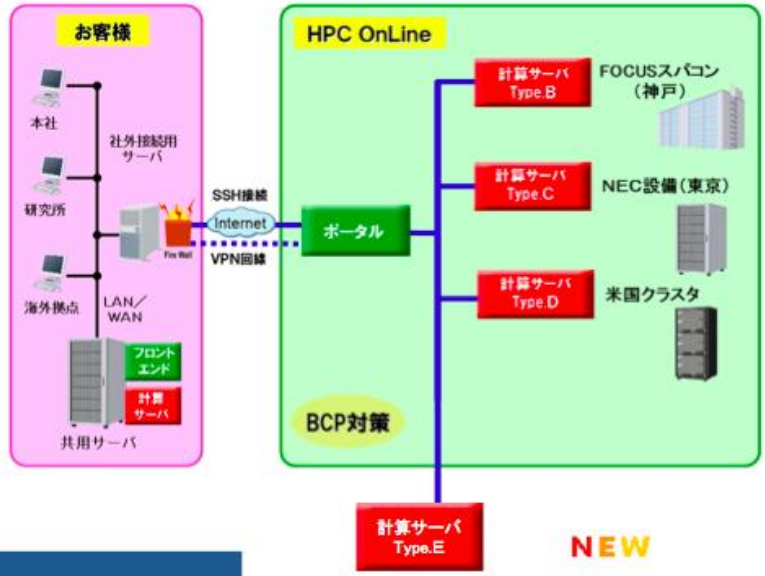
PLEXUS CAEは、解析実行環境を必要なタイミングで、必要なだけスピーディーに提供する**SaaS型サービス**です。
利用ユーザは**AWSを意識することなく**、サーバ調達から解析実行まで実行可能できます。



ISIDブースにてデモ実施中



NEC　クラウドサービス『 HPC OnLine 』



計算サーバとしてAWSを追加

powered by amazon web services

特長①　様々な解析ソフトを提供

プリインストールしたアプリケーションを利用可能（36種）

特長②　複数の計算サーバから選択可能

NEC設備／神戸FOCUS／米国ｸﾗｽﾀ／AWSを提供中

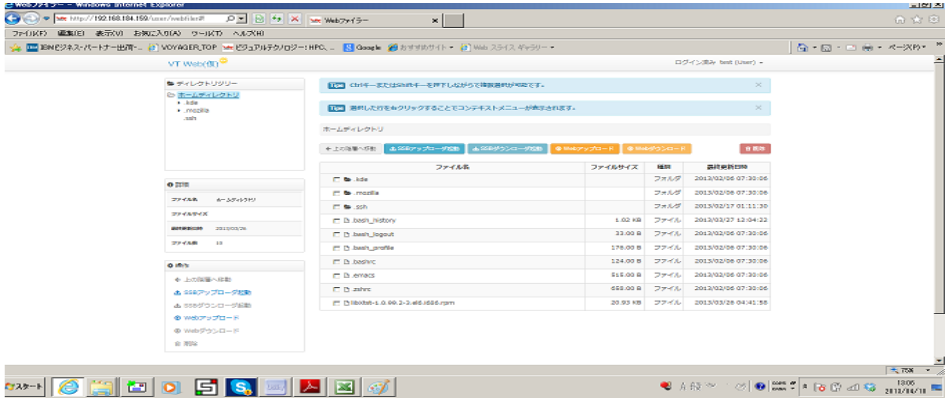
特長③　多くの導入実績

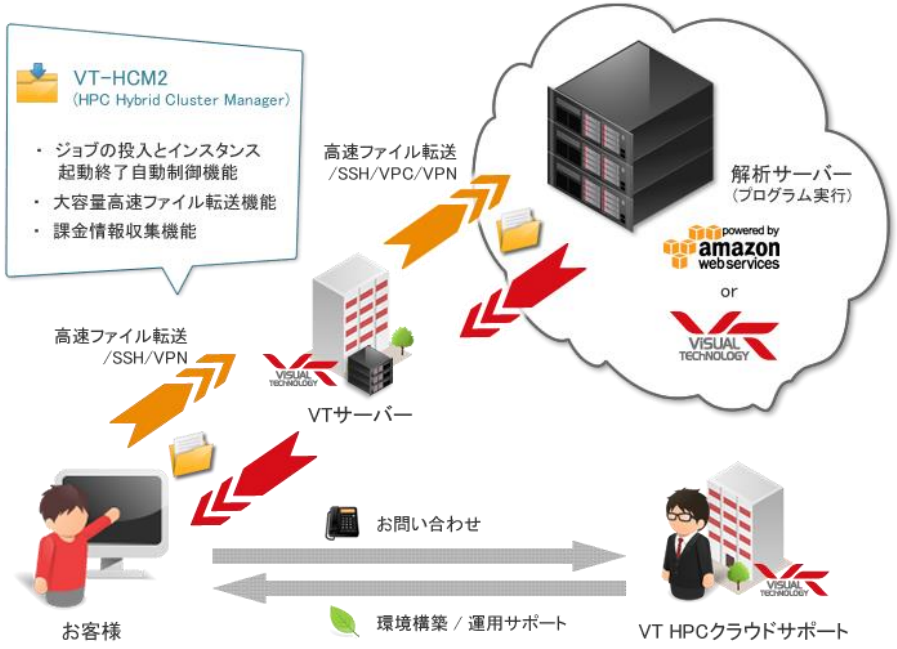
自動車/電機/建設/医薬/解析ソフトベンダ　etc

VT-HCM2(HPC Hybrid Cluster Manager)

主要機能

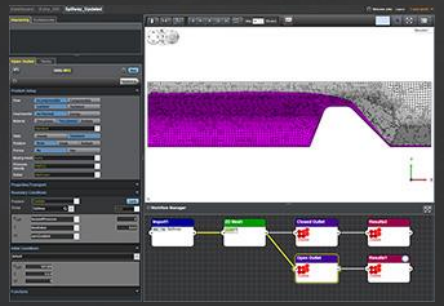
- Jobの投入とインスタンス起動終了自動制御
- 大容量高速ファイル転送
- 課金情報収集

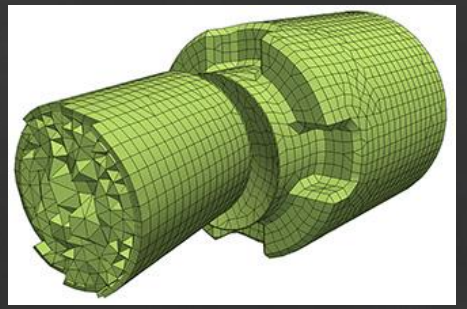


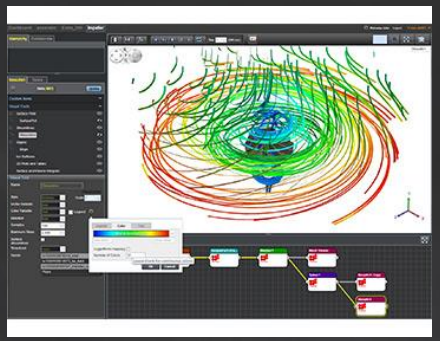


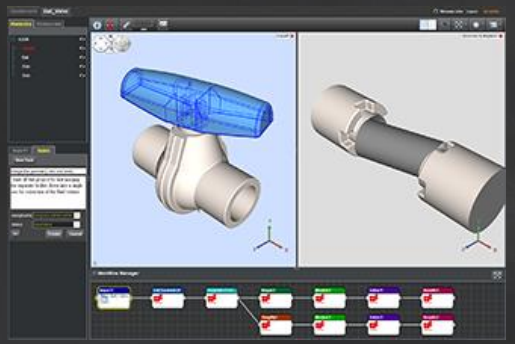
CieSpace

WebベースのSaaS CAEサービス









Solver: OpenFOAM
Meshing :
CAD Import: CATIA, NX, etc.
Visualization :
OpenFOAM,Star-CCM, Fluent, Flow3D
Nastran, Abaqus, Ansys, Marc, LS-Dyna

AWS Marketplace

ソフトウェア構築済み環境を
従量課金ですぐに利用可能

http://aws.amazon.com/marketplace/hpc

AWS HPC Test Drives

AWSパートナー様が提供する
無料検証環境

http://aws.amazon.com/testdrive/hpc
http://aws.amazon.com/jp/testdrive/japan/hpc/

まとめ

HPCクラウドを使うことで可能になること

カイゼン

待ち時間の削減
ハードウェア更改からの開放
コスト削減
データ共有の容易さ
生産性向上
リードタイム短縮

イノベーション

新しいHPC 領域へ
新規研究の検証
新しいHPCアプリ開発
HPCの教育
ベンチマーク調査

aws.amazon.com/hpc

aws.amazon.com/life-sciences

2014.09.09 SAVE THE DATE

http://csd.awseventsjapan.com/

Cloud Storage & DB Day　検索

amazon web services

AWS Summits 2014

Thank you!

AWS Summits 2014

TE-08 実用フェーズに入ったHPCクラウドの実力

松尾康博（ matsuoy@amazon.co.jp ）
アマゾン データ サービス ジャパン
ソリューション アーキテクト

amazon web services