

DRから始めるAWS移行

1億PVのメディアサイトをAWSへ

アイティメディア株式会社

木村 友則



アイティメディアのサイトってどんなもの？



ページビュー数
約**1億** / 月

ユニークブラウザ数
約**2,500万** / 月

配信記事数
約**3,000本** / 月

ITmedia
ニュース

ITmedia
Mobile

MONOist

ITmediaビジネス
ONLINE

ねとろぼ

@IT
atmarkIT

ITmedia
イマ-プライズ

ITmedia
PC USER

EDN
Japan

ITmedia
マーケティング

質問!
ITmedia

QA @IT

ITmedia
イグゼクティブ

ITmedia
LifeStyle

EE Times
Japan

調査のチカラ
Powered by ITmedia Inc.

ITmedia
名作文庫

TechTarget
Japan

ITmedia
eBook
USER

スマート
ジャパン
Smart Japan

キーマンズ ネット

ITmedia
デジカメプラス

モテラボ
3D MODEL LABO

ITmediaヘルスケア
HealthCare

ITmedia

詳しくはコーポレートサイトを参照ください <http://corp.itmedia.co.jp/>



名前	きむら ともりの 木村 友則
----	--------------------------

所属	アイティメディア株式会社 入社1年3ヶ月目 運営サイトのインフラエンジニアです
----	--

好きなAWSサービス
Route53、ELB

最近気になっているAWSサービス
EFS、Lambda



- ▶ **移行検討のきっかけ**
- ▶ **移行計画**
- ▶ **システム概要**
- ▶ **DRサイトの災害時切り替え仕様**
- ▶ **AWSでのシステム構築**
- ▶ **移行実施**
- ▶ **まとめ**



移行検討のきっかけ



ハードウェアのリプレイス時期が迫ってきた

- ▶ **リプレイスしたら5年は継続することになる**
 - データセンタの維持費は割高感があるうえに固定
 - ネットワーク利用料は別途発生
 - 見えにくいメンテナンスコスト
- ▶ **オンプレであることが望ましい理由はあるのか？**
 - ハードウェアを保有する時代ではないのではないか
- ▶ **クラウド移行を5年も先送りにして良いのか？**
 - オンプレのサーバ同様に利用でき、価格も安くなった



今ならいける。 クラウドに行こう！

5年後では遅過ぎるし、
オンプレであることが望ましい理由もない。
クラウド移行の学習コストは高いが、その価値がある。



▶ オンプレの課題

- 仮想環境全体のキャパシティ管理
- リソースの適切な再割り当てが難しい
- スケールできない

▶ クラウド利用の課題

- オンプレの課題は解決できるのか？
- クラウドでの運用ノウハウが一切ない・・・
- 未知の問題点がいろいろ出るのではないか？



クラウドと言ってもどこに行くのか？

- ▶ 運用ノウハウはないが、調査はしていました
- ▶ 候補はいくつかのパブリック・クラウド
- ▶ 詳細は割愛しますが次のような理由でAWSに決定
 - 安全性 (ISO27001等のセキュリティ基準への適合)
 - 信頼性 (EC2のSLA 99.95%等)
 - 柔軟性・先進性
 - 投資効率



移行計画



▶ 原則、現状のままを移行する

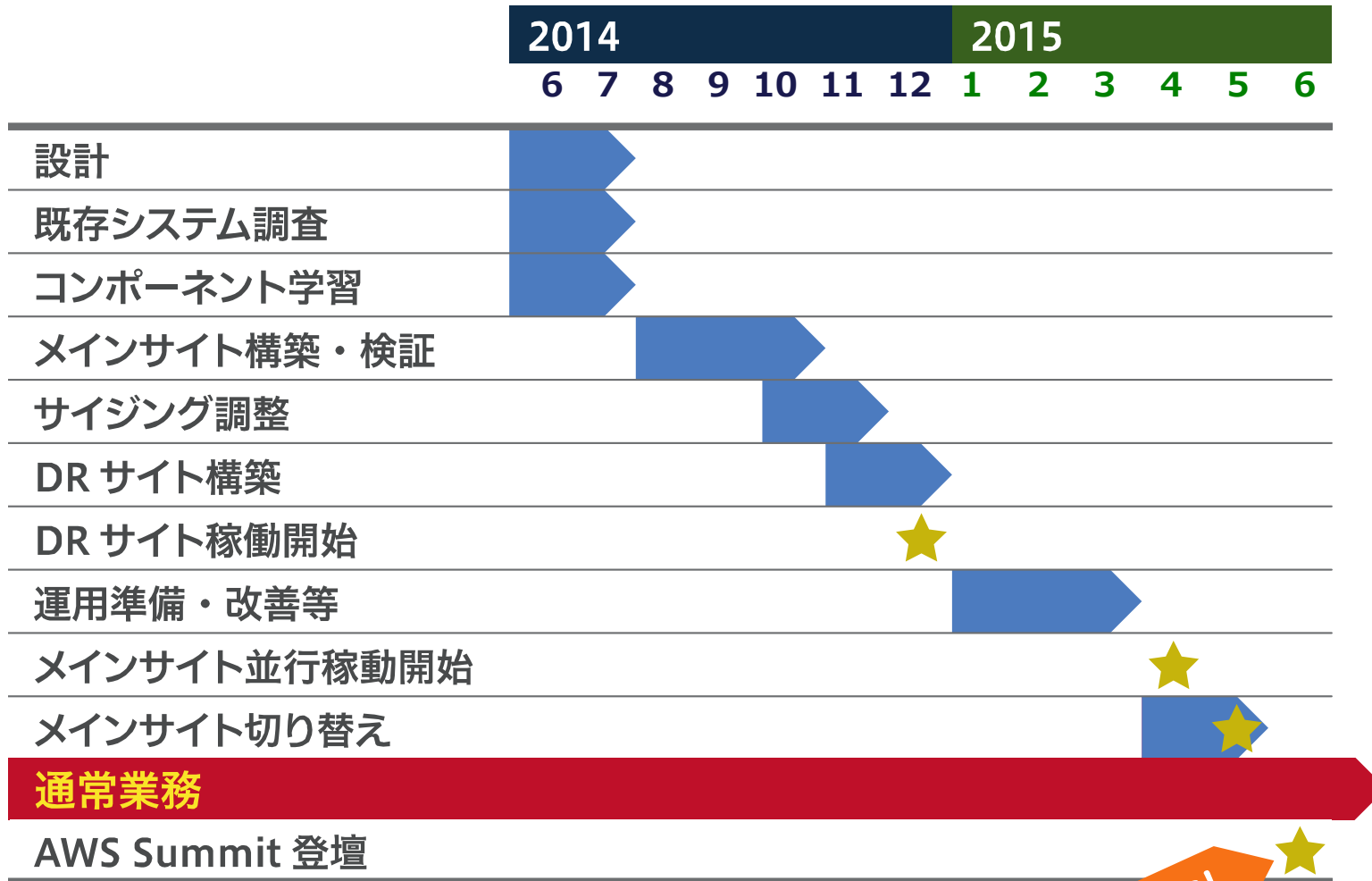
- クラウドへの最適化は移行後に計画・実施
- 移行後の方がTry&Errorが容易
- ただし無理なくできることは少しずつでも実施しておく

▶ まずはDR(Disaster Recovery)サイトを移行する

- 災害等による東京壊滅を想定した代替サイト
- DRサイトの移行作業を通してノウハウの蓄積を行う
- 通常はサービス非稼働なので、比較的気軽に変更できる



移行までの大まかなスケジュール



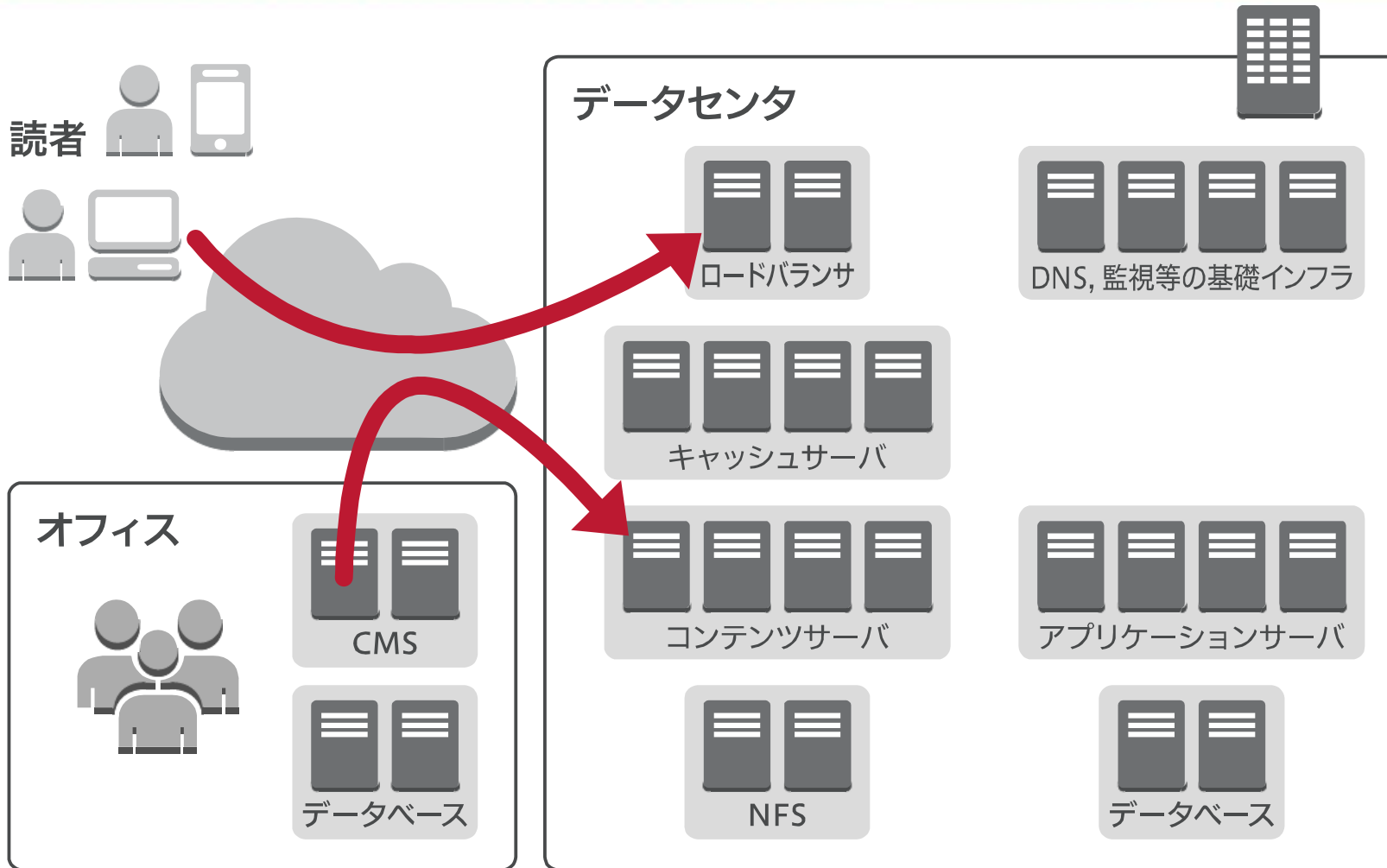
今ココ!



システム概要



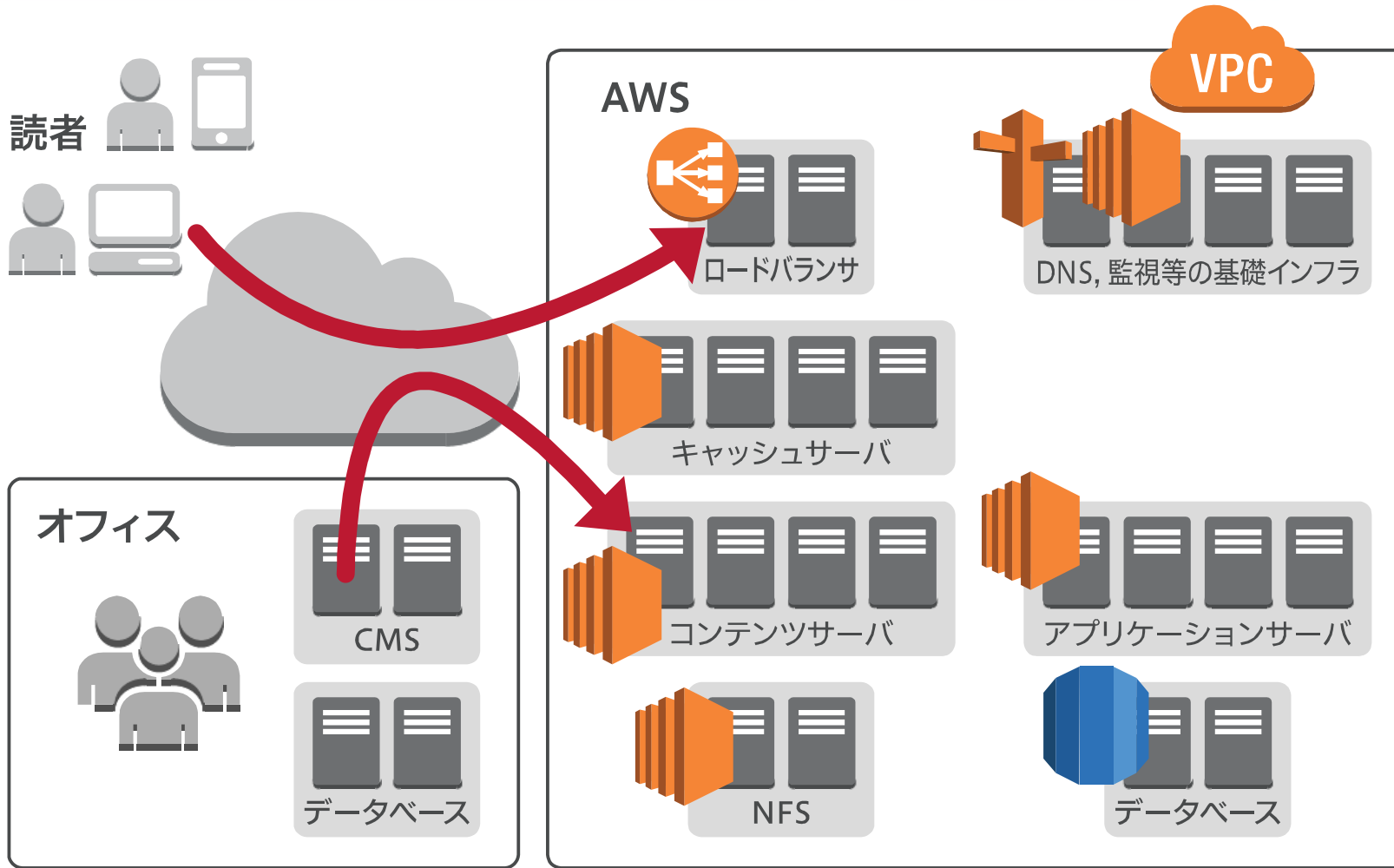
AWS移行前



※ システムの一部抜粋 ※ DRサイトは同構成の縮小版を別データセンタに構築



AWS移行後



※ システムの一部抜粋 ※ DRサイトは同構成の縮小版



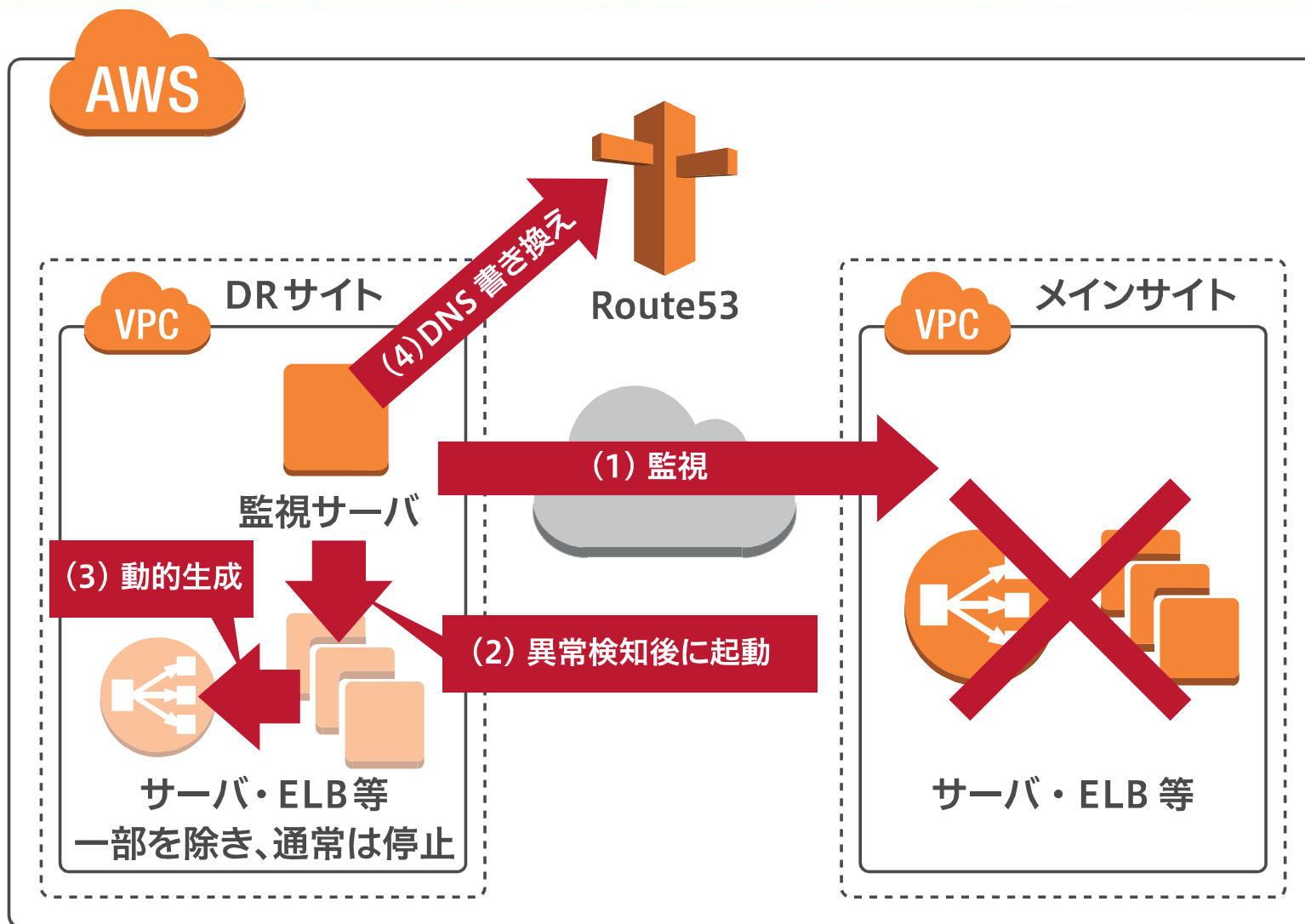
DRサイトの災害時切り替え仕様



- ▶ 災害等による東京壊滅時の代替サイト
- ▶ データ同期を定期的に実施
- ▶ 監視結果に基づき自動的に切り替え、切り戻しは手動
- ▶ 従来の課題
 - DNSの名前解決は一部タイムアウト発生を許容
 - DRサイトとはいえ常時フル稼働
 - 拠点が国内



切り替え方法



▶ 待機時にインスタンスがフル稼働では無駄が多い

▶ EC2の動的起動には注意も必要

- DRサイトの稼働開始タイミングは集中が予想される
- リザーブドインスタンスにしない限り起動の確約はないがコスト削減幅は減る

▶ ELBはEC2の起動時に動的生成

- 1 ELB辺り およそ\$240/年程度
- 数が増えるとそれなりのボリューム

▶ RDSはスナップショットの転送のみ

- 使うときに復元



AWSでのシステム構築



▶ DNS、NFS等の基礎インフラ関連を最初に準備

- ネットワーク・セキュリティ関連、DNS、NFS等

▶ メインサイトに各種サーバを構築

- 既存環境の構築手順書を基にする

▶ テスト実施

- 単体テスト、結合テスト、性能テスト
- T2インスタンスを想定するものは、CPUクレジットを空にする
- I/Oが極端に多いものは、I/Oクレジットを空にする

▶ カスタムAMIを作成しDRサイトにコピー

- Launchし、再テストを実施で完了



▶ 基本的に設計をミスったらやり直し

- スクラップ&ビルドを繰り返すくらいがちょうど良い

▶ ACL/SecurityGroupの適用仕様

- ステートフル? ステートレス?
- いつ適用される?
- 等々

▶ SecurityGroupの粒度に注意

- 当社では共通のベース + プロトコル&from別定義
- SecurityGroup定義の上限値に注意



▶ BINDからRoute53への移行

- 元々BINDのviewを使って、Public/Privateゾーンで返す内容を変えていた
- Public用にRoute53、Private用にBIND継続 (当初の想定)
- Route53とBINDの重複管理が必要?!?!

▶ Route53とBINDの全レコードを共通データで管理

- 重複管理での間違い防止対策
- 内製ツールでそれぞれに一括適用
 - Route53はAPI
 - BINDはnsupdateコマンドで・・・
- (せっかく作りましたが、Route53のPrivate対応がリリースされたので、BINDは捨ててRoute53に統一。嬉しいやら悲しいやら・・・)



▶ 運用のキモなので当然冗長化

▶ データ同期方法

- GlusterFS、DRBD等のさまざまな方法がある
- 当社でも検証をしたがシンプルなlsyncdを選択

▶ 切り替え方法

- マルチAZを想定すると、単純なIPの引き継ぎができない
- 手動でNFSクライアント側から切り替え？
- VIP割り当て & RouteTable書き換え？
- NFS以外のプロトコルで切り替え不要にする？

▶ AZ越えの遅延

- NFSクライアントで noac オプションを使うと遅延の影響大



▶ 負荷に波がある用途にはT2インスタンスが良い

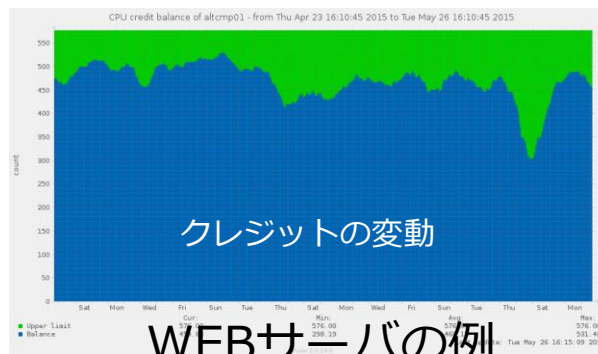
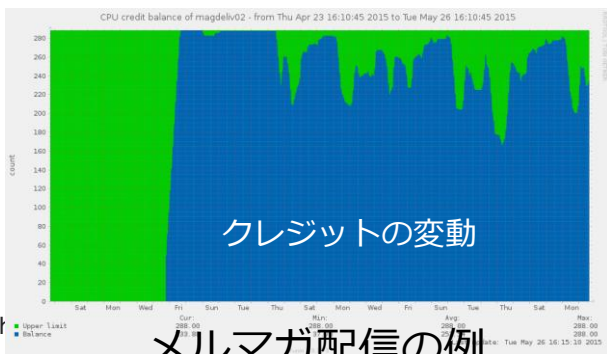
- CPUクレジットを使い切った場合での性能が許容できるか？
- 使いきる前にクレジットの回復が見込めるか？

▶ コマンド '`yes > /dev/null`' で強制的に消費して検証

- 30分程度で消費できる
- APIでクレジットの無効化ができれば良いのに・・・

▶ 当社の場合

- メルマガ配信や一部WEBサーバに利用



▶ GeneralPurpose、MagneticタイプEBSで要注意

- GiB容量 ×3 IOPSが保証されるベースライン

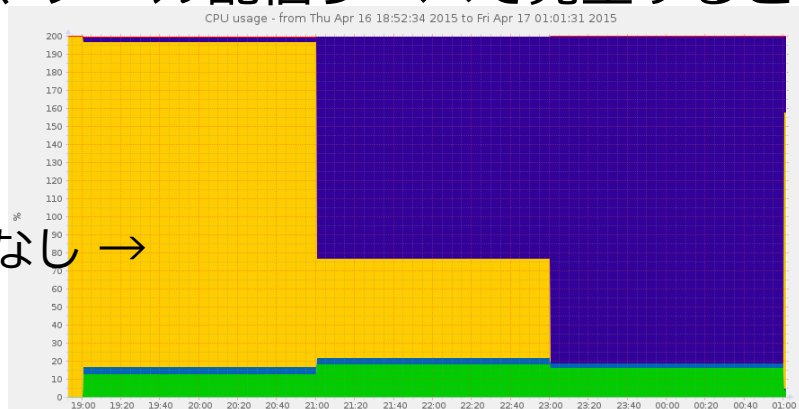
▶ ベンチマークソフトでI/Oクレジットを使い切る

- fio等を使って30分程度で消費できる (容量による)

▶ キャッシュサーバやメール配信は特に注意

- 突然IOPSがベースラインまでしか使えなくなる
- たとえば、ELB配下のキャッシュサーバで発生すると・・・
- たとえば、メール配信サーバで発生すると・・・

最初は
I/O waitほぼなし →



←突然のI/O Waitで
身動き取れず



移行実施



▶ DRサイト

- 2014年12月に実施
- 各ドメインのDNSをRoute53に移行
- データの同期先をDRサイトに変更

▶ メインサイト

- 2015年4～5月に実施
- 段階的な移行を計画
 1. 他との依存関係がなく独立したもの
 2. 静的コンテンツの並行稼働 (オンプレ50%、AWS50%等)
 3. 会員サービス等のDBが関連するもの
 4. 静的コンテンツの全面稼働
- 準備は3月まで、4～5月の移行当日は手順をこなすだけ
 - 当日は無理しちゃ負け、努力と根性は前日まで ;-) by ボス



▶ 大きなトラブルもなく移行完了

- 準備が良かったようです (◡‿◡)
- なので、ネタにするほど書くことはありません m(__)m

▶ とはいえ、多少の注意点はあります

- ビジネス以上のサポートに加入しておきましょう
- 問題なく動き始めたからといって、すぐには気を緩めない
 - CPUとI/Oのクレジット切れには要注意
 - AWS向けに監視機能を拡充したほうが良い



▶ わりと突然サーバが落ちる

- 直近3ヶ月で3回 (計80弱のEC2)
- EC2 AutoRecoveryで自動再起動して復旧
- 対象によってはリカバリー操作が必要な場合もある

▶ メンテナンスに伴う再起動要請の対応どうする

- 物理インフラの障害等で再起動要請がある
- ある時は2週間の猶予があったが、期間はケースバイケース
- 原則として別AZで冗長構成をとっておく



まとめ



▶ クラウドは怖くない

- 特に困ることもなく、良さの方が目につく

▶ API/CLIは積極的に活用したい

- 活用することでメリットを最大限享受できる

▶ 制限事項、仕様をよく確認

- ググる前にドキュメントを

▶ コストに対して敏感になった

- 過剰なリソースを見ると、どうにかして無駄を省きたくなる

▶ 疎結合化を意識するようになった

- クラウドへの最適化を考えると、システムの疎結合化が必須



- ▶ **無駄を省き、コストの最適化を進める**
 - 過剰リソースは障害同様に即時対応すべき
- ▶ **疎結合化を進め、メンテナンス性の向上**
 - 脆弱性対応等でも事前ネゴに手間をかけずに再起動したい
- ▶ **AWSに依存しすぎないように注意**
 - スイッチングコストが高くなり過ぎては身動きが取れない
- ▶ **一緒にやってくれる人募集しています**
 - 詳しくはこちら <http://corp.itmedia.co.jp/>



ご清聴ありがとうございました。

WEBサイトにも是非お立ち寄りください。



<http://www.itmedia.co.jp/>

