

Amazon RDS for PostgreSQL 検証報告

AWSプロダクトシリーズ|よくわかるクラウドデータベース
2014/01/17

SRA OSS, Inc. 日本支社
石井 達夫
正野 裕大

性能

- スペックの異なる RDS インスタンスでベンチマークを測定

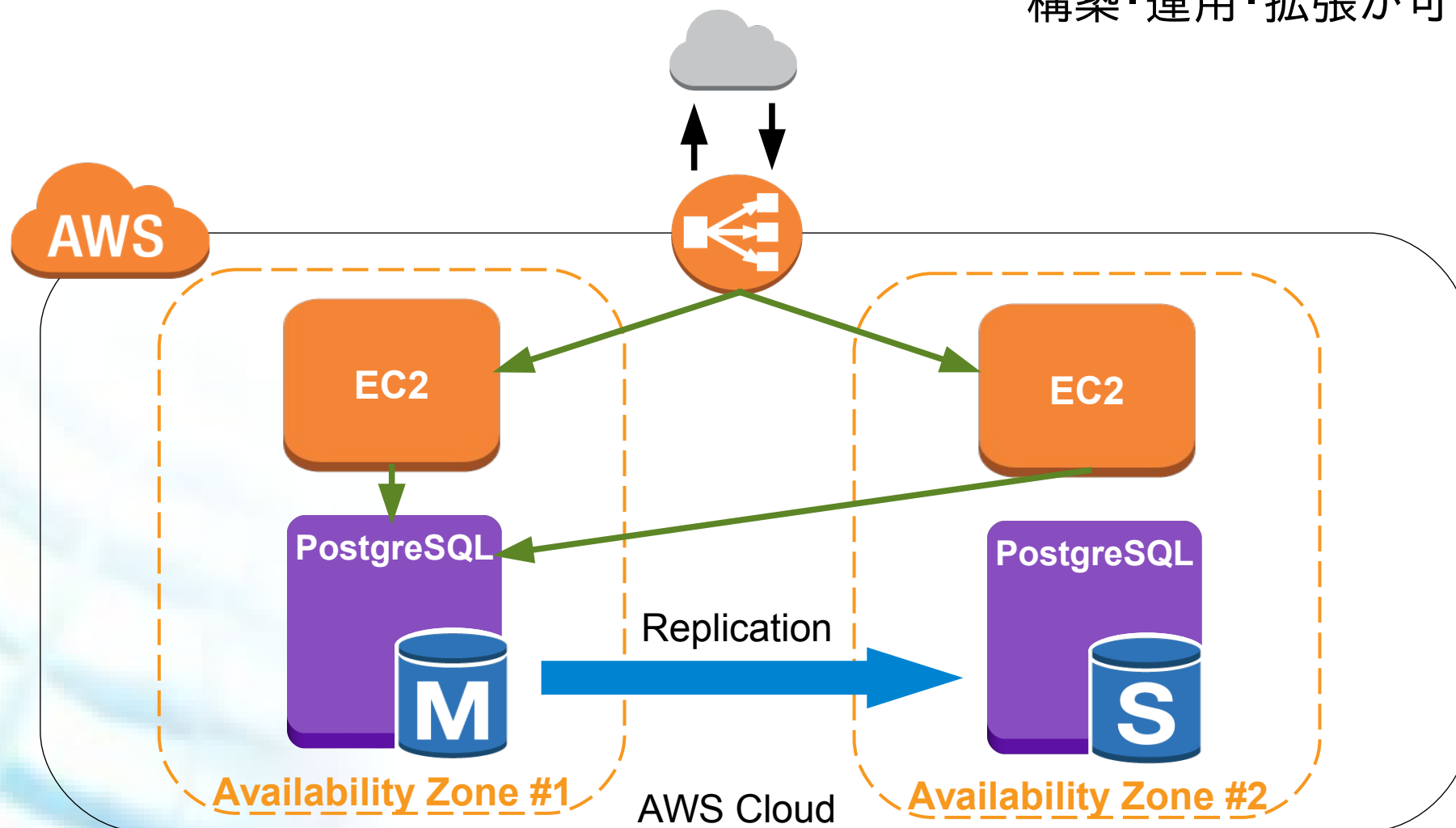
技術的トピック

- 素の PostgreSQL と違う点

- Amazon RDS と PostgreSQL の概要
- RDS for PostgreSQL を使ってみた
RDS インスタンスの作成 / Multi-AZ / バックアップ / スケールアップ / モニタリング

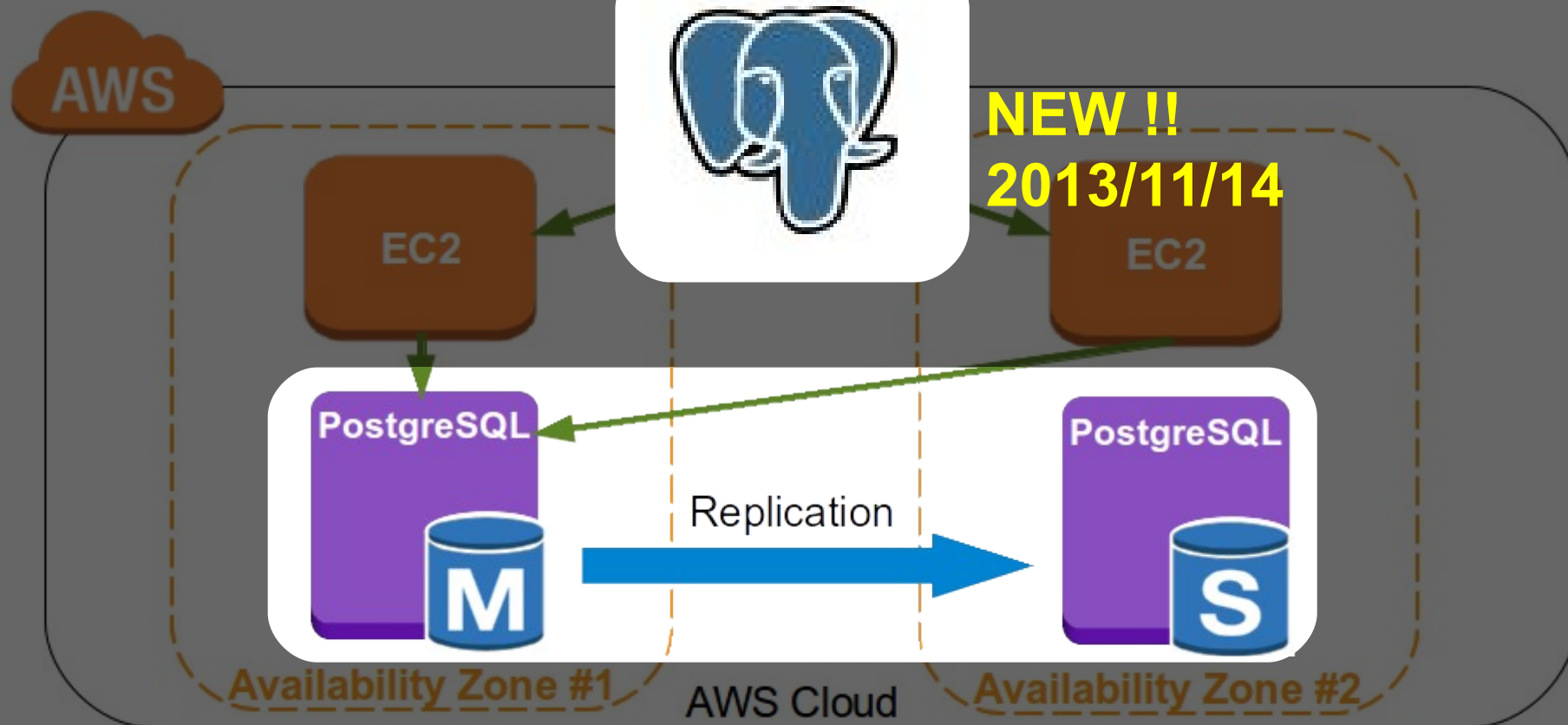
付録

- **Amazon Web Services** が提供する
マネージドデータベース (DB) サービス
 - 簡単に Web コンソールを通してクラウド上で DB サーバの
構築・運用・拡張が可能



- Amazon Web Services が提供する
マネージドデータベース (DB) サービス

•



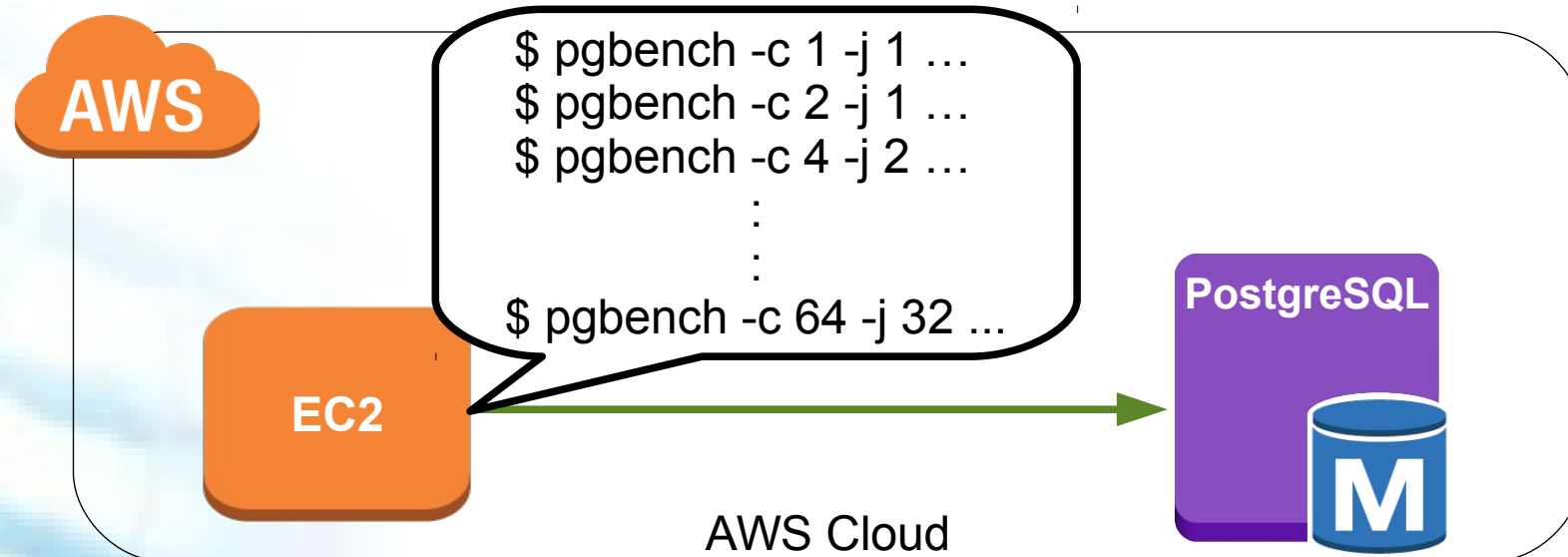
性能

- PostgreSQL に付属する標準ベンチマークツール (pgbench) で
ベンチマークを実施

- 3 つの RDS インスタンスでベンチマークを測定

DB インスタンスクラス	CPU	メモリ
ハイメモリエクストララージ	3.25 ECU × 2 仮想コア	17.1 GB
ハイメモリダブルエクストララージ	3.25 ECU × 4 仮想コ	34 GB
ハイメモリクラスタエイトエクストララージ	88 ECU (32 ハイパースレッドコア)	244 GB

- EC2 インスタンスをクライアントとして pgbench を実行



■ RDS インスタンスの情報 | DB サーバ (pgbench 対象)

DB インスタンスクラス	ストレージサイズ	IOPS	パラメータ	ベンチ用 DB
ハイメモリエクストララージ	100GB	1000	デフォルト	testdb
ハイメモリダブルエクストララージ				
ハイメモリクラスタエイトエクストララージ				

■ EC2 インスタンス | クライアント (pgbench 実行側)

インスタンスクラス	ハイメモリクラスタエイトエクストララージ (cr1.8xlarge)
スペック	32 コア (仮想) / 88 ECU / RAM 244GB
OS	Red Hat Enterprise Linux 6.4
PostgreSQL 9.3.1	PostgreSQL コミュニティのリポジトリを使用

pgbench 実行側が
ボトルネックにならないように
高性能のインスタンスを
使用

- pgbench 用 DB testdb の初期化
 - スケールファクタ 1000 (データ行 1 億件 / DB サイズ 15GB)

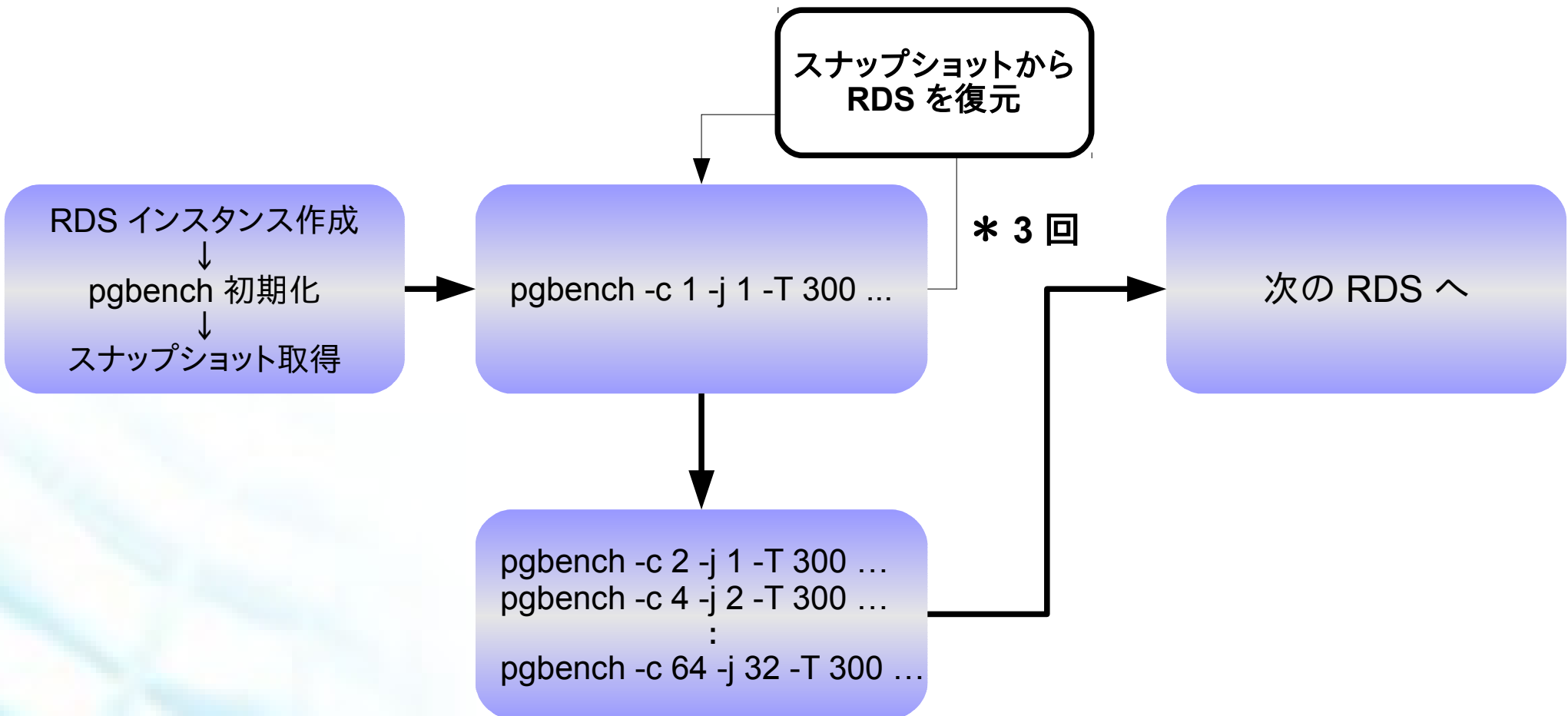
```
[ec2-user@ec2 ~]$ pgbench -i -s 1000 -U <DB USER> -h <RDS> testdb
```

- pgbench の実行
 - 標準 (TCP-B) の pgbench
 - 同時接続数 1, 2, 4, 8, 16, 32, 64 を 3 回ずつ実行
 - 実行時間は 300 秒
 - 実行すると TPS (1 秒間に実行したトランザクション数) が出力される

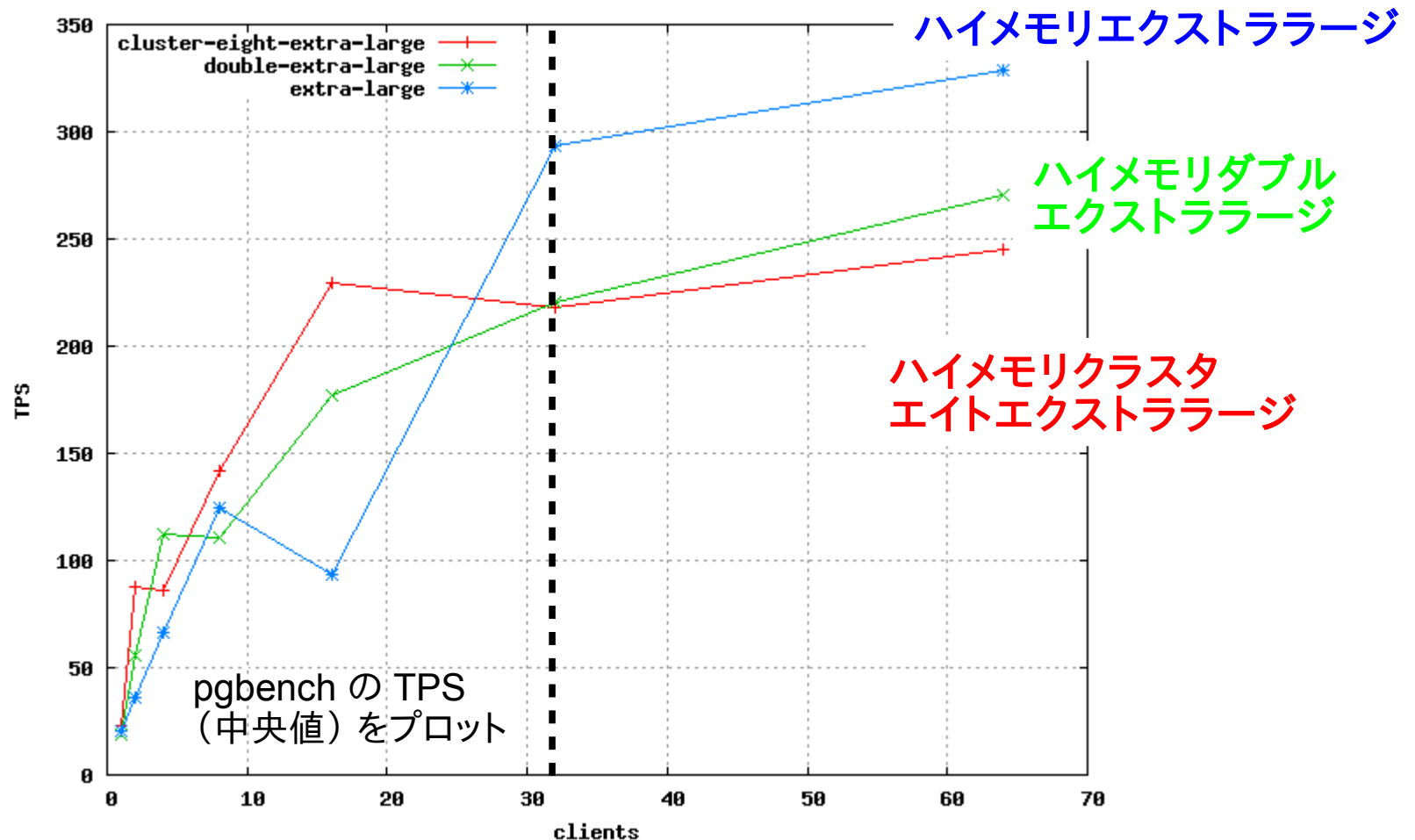
```
[ec2-user@ec2 ~]$ pgbench -c 1 -j 1 -T 300 -U <DB USER> -h <RDS> testdb * 3 回
[ec2-user@ec2 ~]$ pgbench -c 2 -j 1 -T 300 -U <DB USER> -h <RDS> testdb * 3 回
[ec2-user@ec2 ~]$ pgbench -c 4 -j 2 -T 300 -U <DB USER> -h <RDS> testdb * 3 回
[ec2-user@ec2 ~]$ pgbench -c 8 -j 4 -T 300 -U <DB USER> -h <RDS> testdb * 3 回
[ec2-user@ec2 ~]$ pgbench -c 16 -j 8 -T 300 -U <DB USER> -h <RDS> testdb * 3 回
[ec2-user@ec2 ~]$ pgbench -c 32 -j 16 -T 300 -U <DB USER> -h <RDS> testdb * 3 回
```

■ 測定の流れ

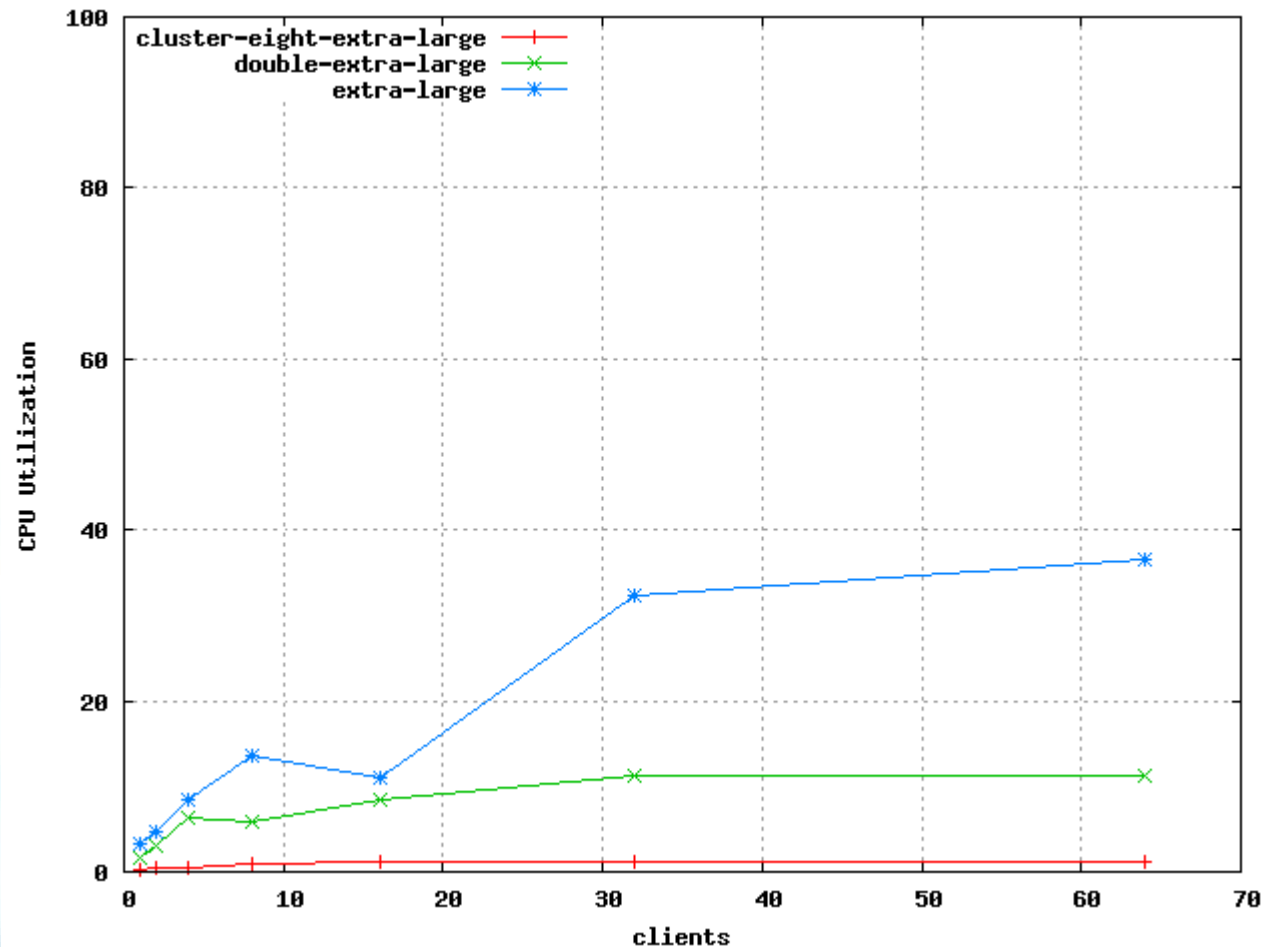
- 各インスタンスで以下を実行



- 同時接続数が増えるにつれて性能 (TPS) が上昇している
- 同時接続数がコア数を越えても性能が上昇し続けた
- 最も性能の高い DB インスタンスクラス が一番 TPS が低い



- 性能の高い DB インスタンスほど CPU 利用率が低い
- おそらく I/O がボトルネックになっている



- 以下の条件で `pgbench -c 64 -j 32 -T 300 ...` を実行
 - RDS インスタンスの情報 | DB サーバ (pgbench 対象)

DB インスタンスクラス	ストレージサイズ	IOPS	パラメータ	ベンチ用 DB
ハイメモリエクストララージ	600GB	6000	デフォルト	testdb

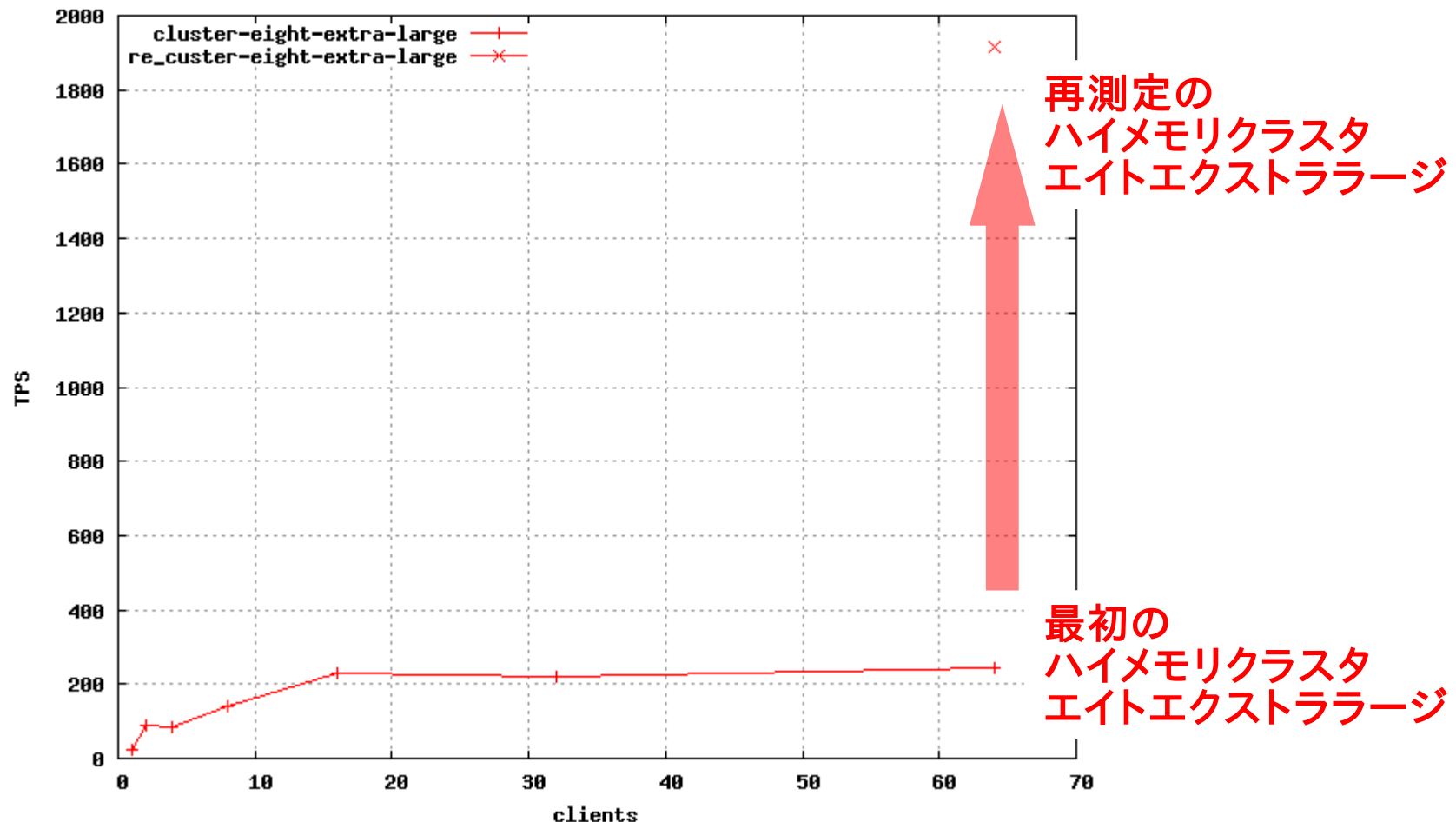
IOPS を 6 倍に

- EC2 インスタンス | クライアント (pgbench 実行側)

インスタンスクラス	M1 エクストララージ (m1.xlarge)
スペック	4 コア (仮想) / 8 ECU / RAM 15GB
OS	Red Hat Enterprise Linux 6.4
PostgreSQL 9.3.1	PostgreSQL コミュニティのリポジトリを使用

pgbench 実行側が
ボトルネックにならない程度に
高性能のインスタンスを
使用
(少しスペックダウン)

- IOPS を 6 倍にしたら TPS がおよそ 8 倍に
 - Write IOPS は 5000 前後 に
 - CPU 使用率は低いまま



技術的トピック

- 素の PostgreSQL と違う点

- クライアント認証の設定を行う手段が提供されていない
 - PostgreSQL では pg_hba.conf ファイルを編集してクライアント認証を設定する
 - 接続方式、DB ユーザ、接続 DB、接続元、認証方式 を制御できる
 - RDS では AWS のセキュリティグループでしか設定できない
 - 接続元のみ

- DB のスーパーユーザになれない
 - 以下はできる (AWS が PostgreSQL に手を加えている?)
 - テーブルスペースの作成
 - 他のユーザのセッションの確認と切断
 - AWS が用意した拡張モジュールのロード

■ PostgreSQL の拡張モジュールは以下が使える

btree_gin	btree_gist	chkpass	citext	cube
dblink	dict_int	dict_xsyn	earthdistance	fuzzystrmatch
hstore	intagg	intarray	isn	ltree
pgcrypto	pgrowlocks	pg_trgm	plperl	plpgsql
pltcl	postgis	postgis_tiger_geocoder	postgis_topology	sslnfo
tablefunc	tsearch2	unaccent	uuid-oss	

■ PostgreSQL の拡張モジュールは以下が使える

btree_gin	btree_gist	citext	cube
dblink	dict_int	dict_xsyn	earthdistance
hstore	intagg	intarray	fuzzystmatch
pgcrypto	pg_trgm	plperl	ltree
pltcl	postgis	postgis_topology	plpgsql
tablefunc	tsearch2	unaccent	sslnfo
		uuid-oss	

大文字/小文字の
区別がない文字列型を
提供

ユーザ定義関数の
手続き言語
(plperl, pltcl もある)

データベースの
暗号化と複合

地理空間情報を扱う

接続中クライアントの
SSL 証明書情報を
提供

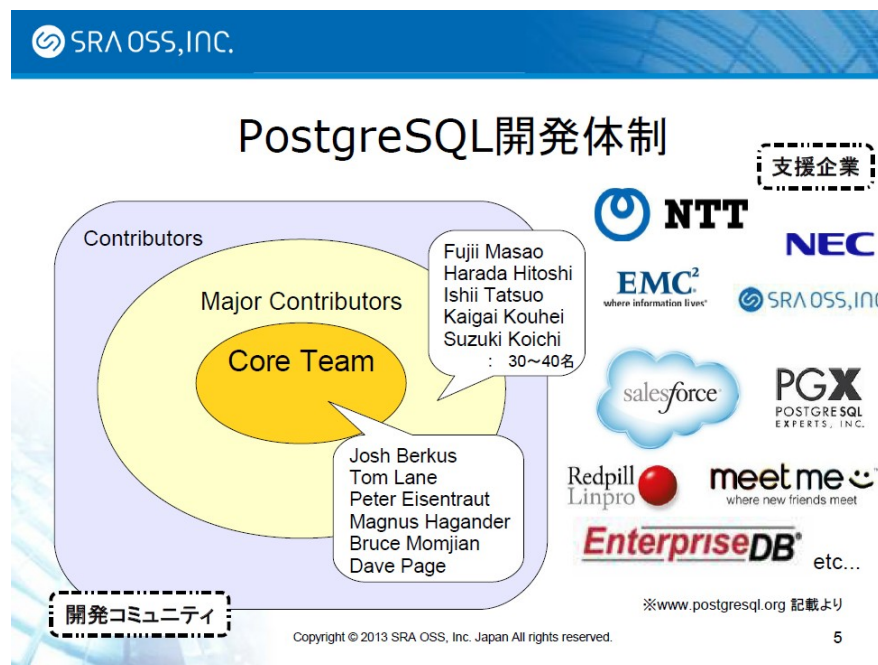
- デフォルトの設定では日本語を DB に格納するには不向き
 - 日本語のデータベースでは DB のロケールを C (ロケールなし) にしておく必要がある
 - ロケール: 文字列照合順序、通貨単位、小数点記号などを規定
- postgresql.conf パラメータは RDS 側で設定されたもの、固定されたものが運用上は特に問題なし
 - 共有メモリのサイズはおよそ RDS インスタンスのメモリの 1/4
 - WALバッファ(トランザクションログ書き込み時に利用されるバッファ)は共有メモリの 1/32 (PostgreSQL のデフォルト)
- ストリーミングレプリケーションは行えない
- デフォルトで SSL 通信が有効
 - MySQL および SQL Server、PostgreSQL のみ対応
 - SSL 通信はそこそこ重い処理
 - クライアント側で無効にできる

Amazon RDS と PostgreSQL の概要

- Amazon RDS for PostgreSQL の特徴
- RDBMS における PostgreSQL の位置づけ

- DB サーバのマネージドサービス
 - 手軽にセットアップ・運用・スケーリング
 - パラメータの最適化
 - CPU 使用率・メモリ・I/O・接続数などのモニタリング
 - PostgreSQL の自動アップデート
- HA 構成で可用性アップ
 - Multi-AZ で同期物理レプリケーションと自動フェイルオーバー
- バックアップ
 - 自動バックアップ
 - 任意のスナップショット

- 代表的なオープンソース RDBMS
 - PostgreSQL 6.0 (1996 ~) から 15 年以上の歴史
- 商用 DB と比較しても機能・性能面で引けをとらない
 - 業務基幹システムの商用 DB 製品の置き換え
 - Blog、SNS、ゲーム等の新しいオンラインサービス
- コミュニティベースで開発
 - 一つのオーナー企業、オーナー個人を持たない
 - 1 バージョン / 1 年 を 10 年以上

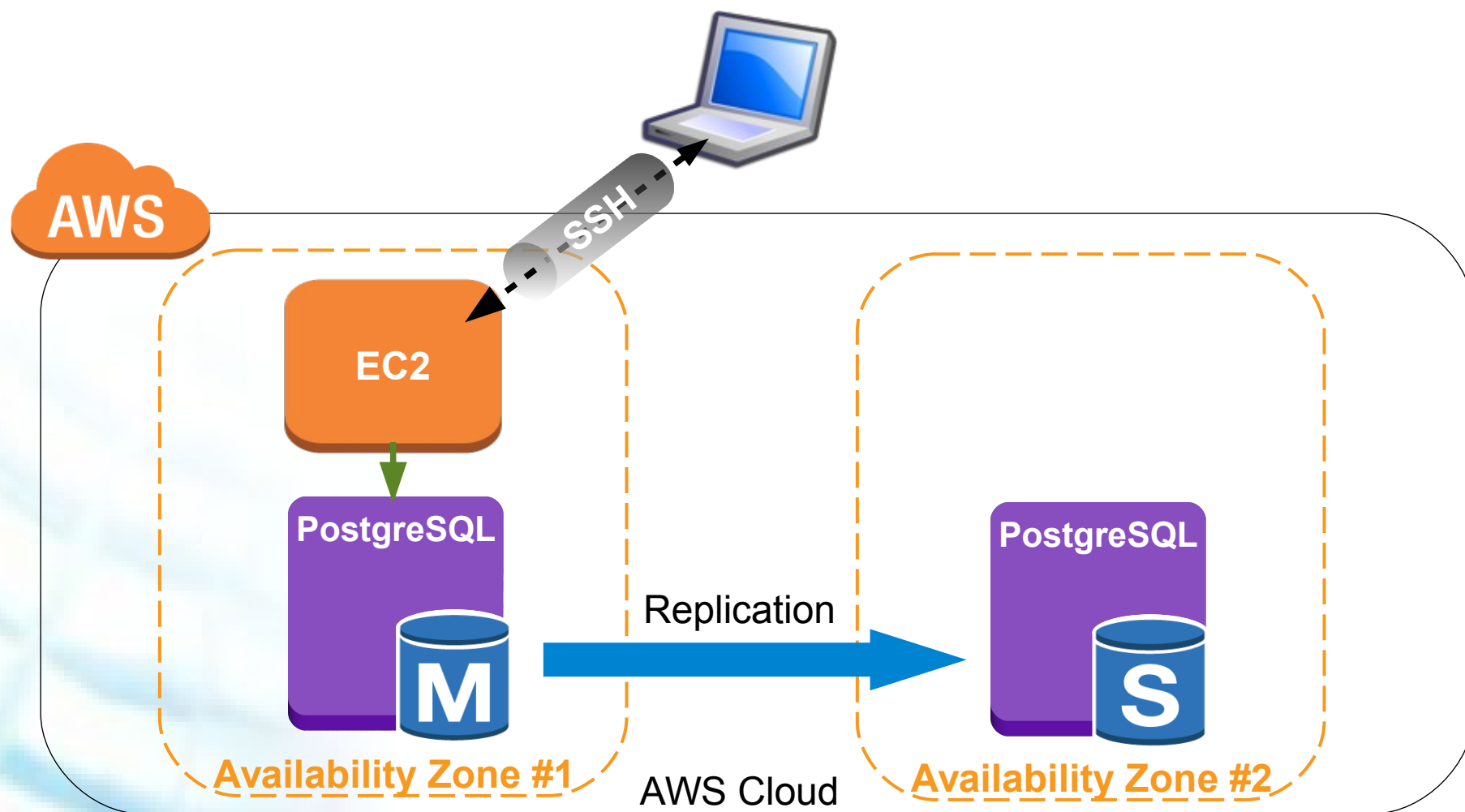


オープンソースデータベース PostgreSQL 最新動向のご紹介
(SRA OSS, Inc. 2013-06-27)

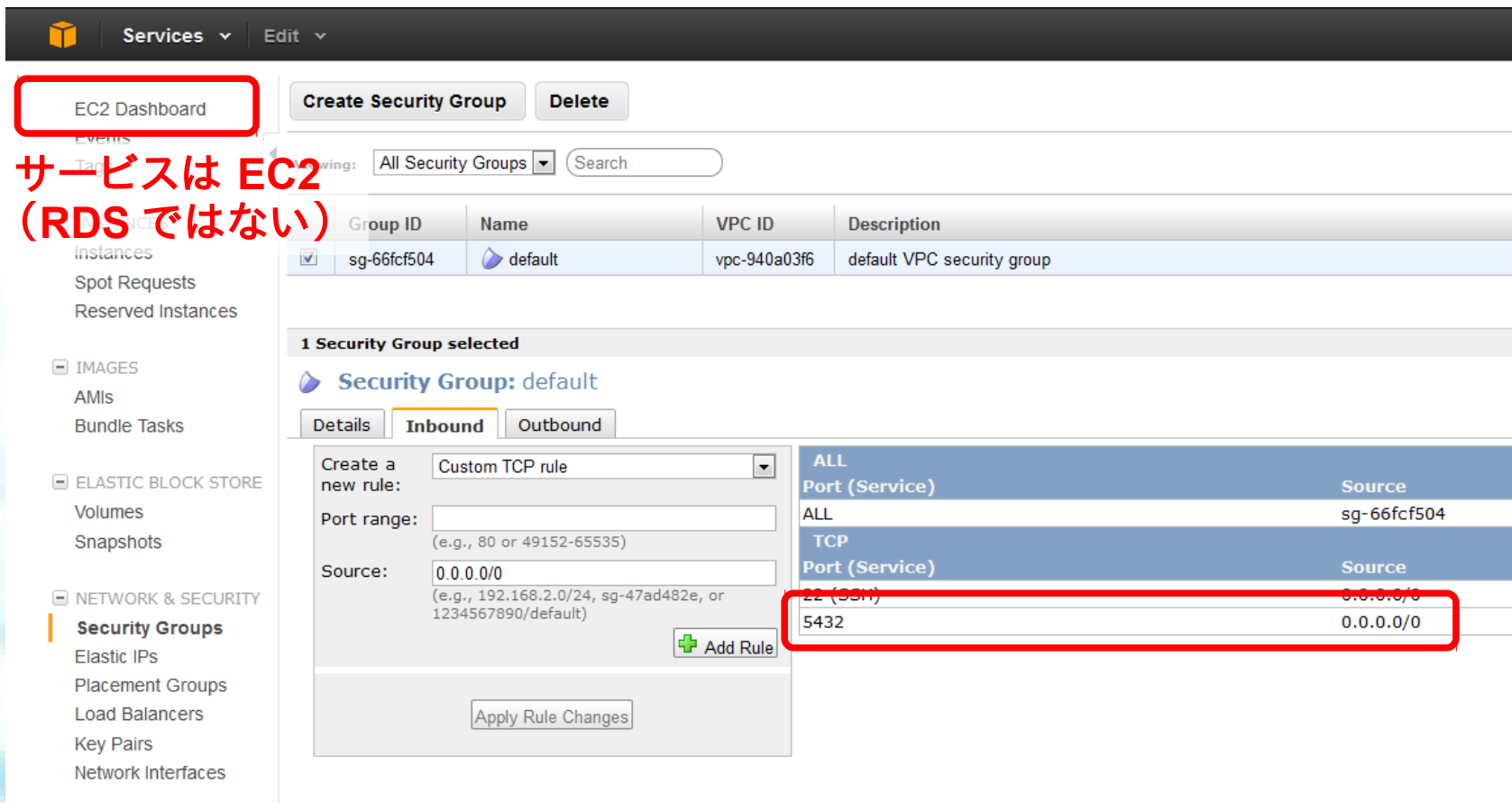
ユーザビリティ

- RDS インスタンスの作成
- バックアップ
- モニタリング
- Multi-AZ
- スケールアップ

- クライアント用 EC2 インスタンスは作製済み
- PostgreSQL の RDS インスタンス | HA 構成 を作ってみる



- 事前に**セキュリティグループ**を設定しておく
 - AWS の仮想ファイアウォール
 - インスタンスごとに任意のセキュリティグループを設定できる
 - 今回はデフォルトの設定 + 5432 ポートを開けておく
 - 5432 : PostgreSQL のデフォルトポート



サービスは EC2 (RDS ではない)

Group ID	Name	VPC ID	Description
sg-66cf504	default	vpc-940a03f6	default VPC security group

1 Security Group selected

Security Group: default

Details Inbound Outbound

Create a new rule: Custom TCP rule

Port range: (e.g., 80 or 49152-65535)

Source: 0.0.0.0/0 (e.g., 192.168.2.0/24, sg-47ad482e, or 1234567890/default)

ALL Port (Service) Source

ALL sg-66cf504

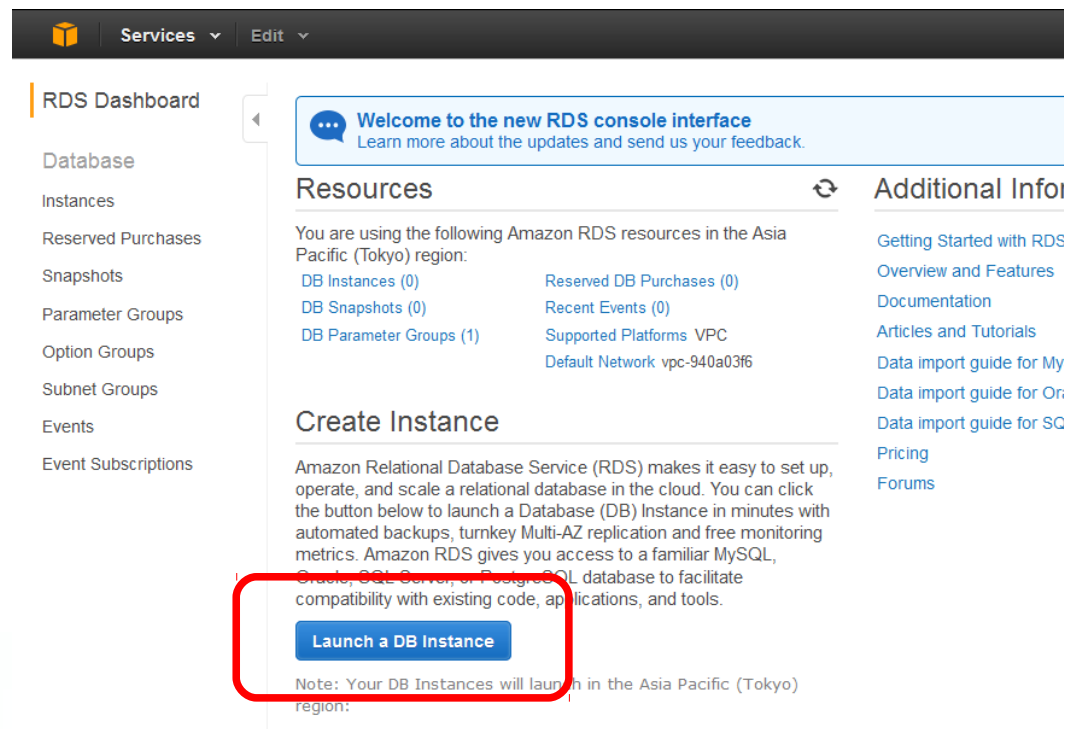
TCP Port (Service) Source

22 (SSH) 0.0.0.0/0

5432 0.0.0.0/0

Apply Rule Changes

■ Web コンソールから Launch a DB Instance !



■ ウィザードに従って項目を埋めていくだけ

DB エンジン

→
インスタンス
スペック

→
セキュリティ
ネットワーク構成

→
DB パラメータ

→
バックアップ
メンテナンス

■ PostgreSQL を選択

 Services ▾ Edit ▾

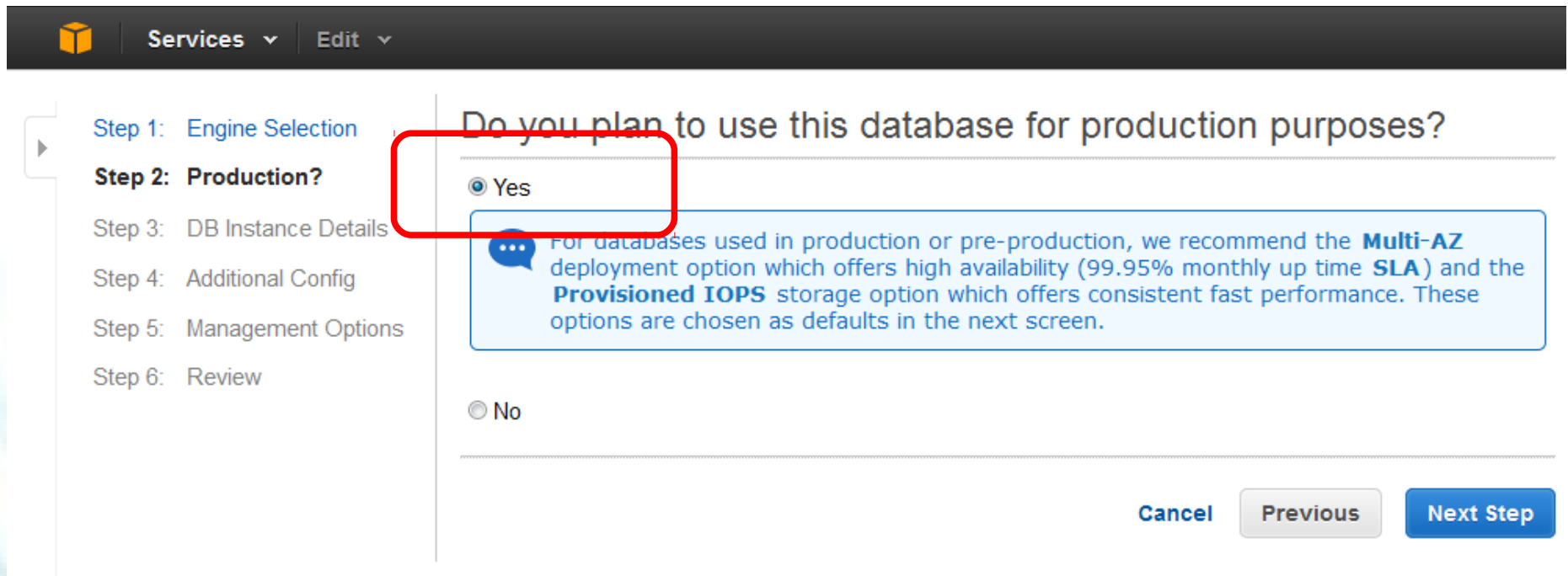
Step 1: Engine Selection
...

Engine Selection

To get started, choose the DB Instance details below and click Select

	mysql MySQL Community Edition	Select ▶
	postgres PostgreSQL	Select ▶
	oracle-se1 Oracle Database Standard Edition One	Select ▶
	oracle-se Oracle Database Standard Edition	Select ▶
	oracle-ee Oracle Database Enterprise Edition	Select ▶
	sqlserver-ex Microsoft SQL Server Express Edition <i>Note that SQL Server Express Edition limits the storage of per database to a maximum of 10GB. Refer to this link for more details.</i>	Select ▶
	sqlserver-web Microsoft SQL Server Web Edition	

- Yes Multi-AZ ・ Provisioned IOPS を利用する
- No Multi-AZ ・ Provisioned IOPS を利用しない



Services ▾ Edit ▾

Step 1: [Engine Selection](#)

Step 2: Production?

Step 3: [DB Instance Details](#)

Step 4: [Additional Config](#)

Step 5: [Management Options](#)

Step 6: [Review](#)

Do you plan to use this database for production purposes?

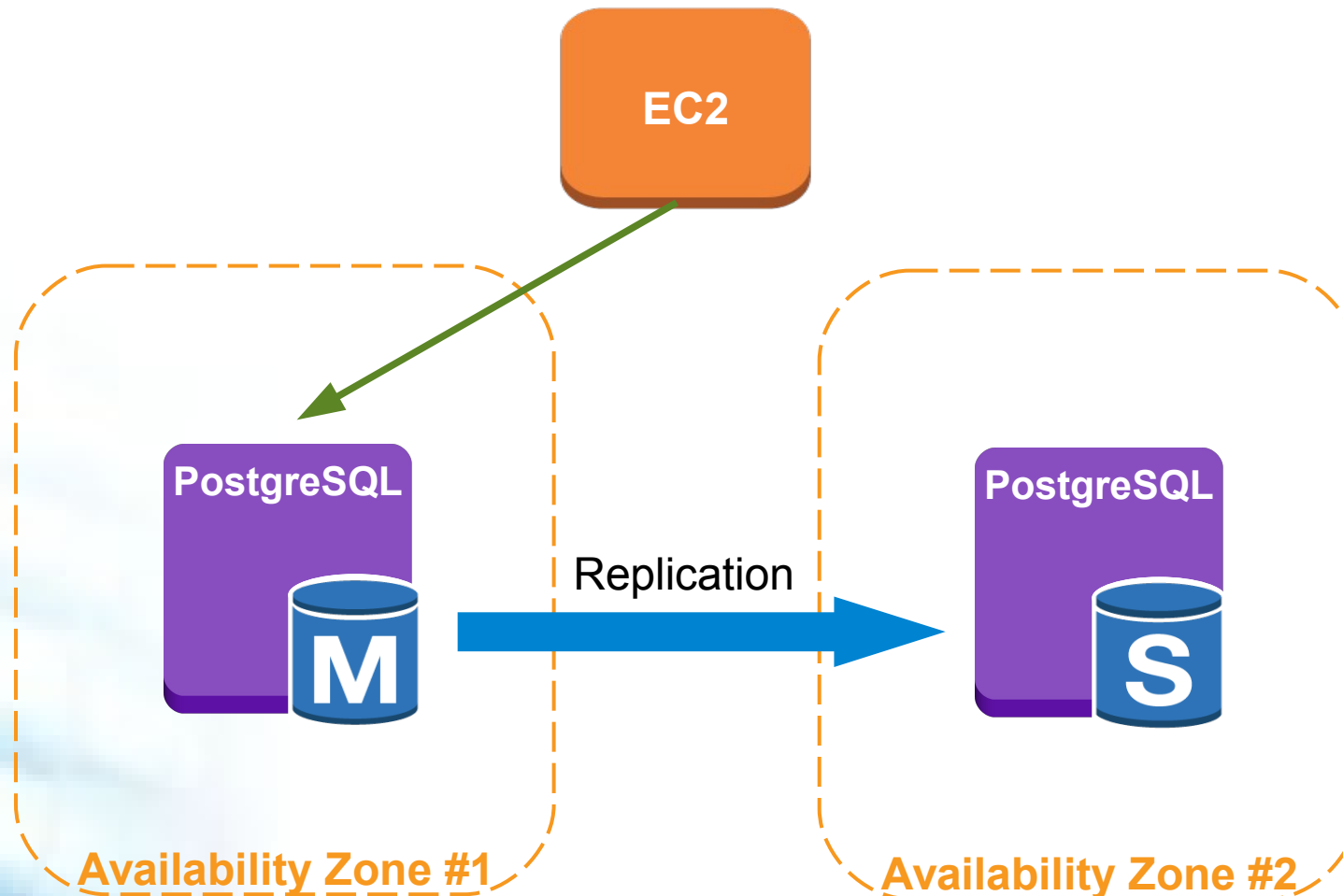
☒ Yes

For databases used in production or pre-production, we recommend the **Multi-AZ** deployment option which offers high availability (99.95% monthly up time **SLA**) and the **Provisioned IOPS** storage option which offers consistent fast performance. These options are chosen as defaults in the next screen.

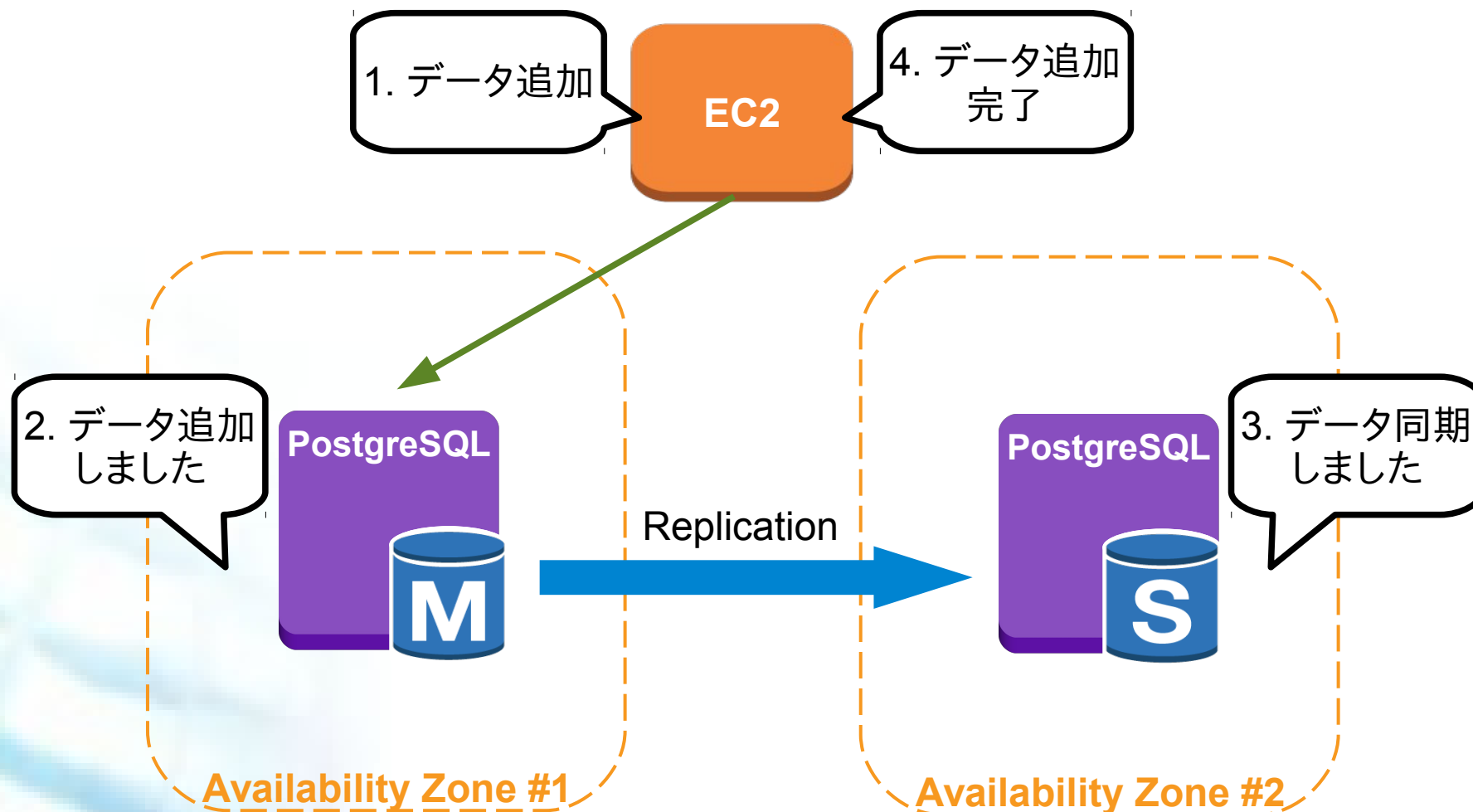
☐ No

[Cancel](#) [Previous](#) [Next Step](#)

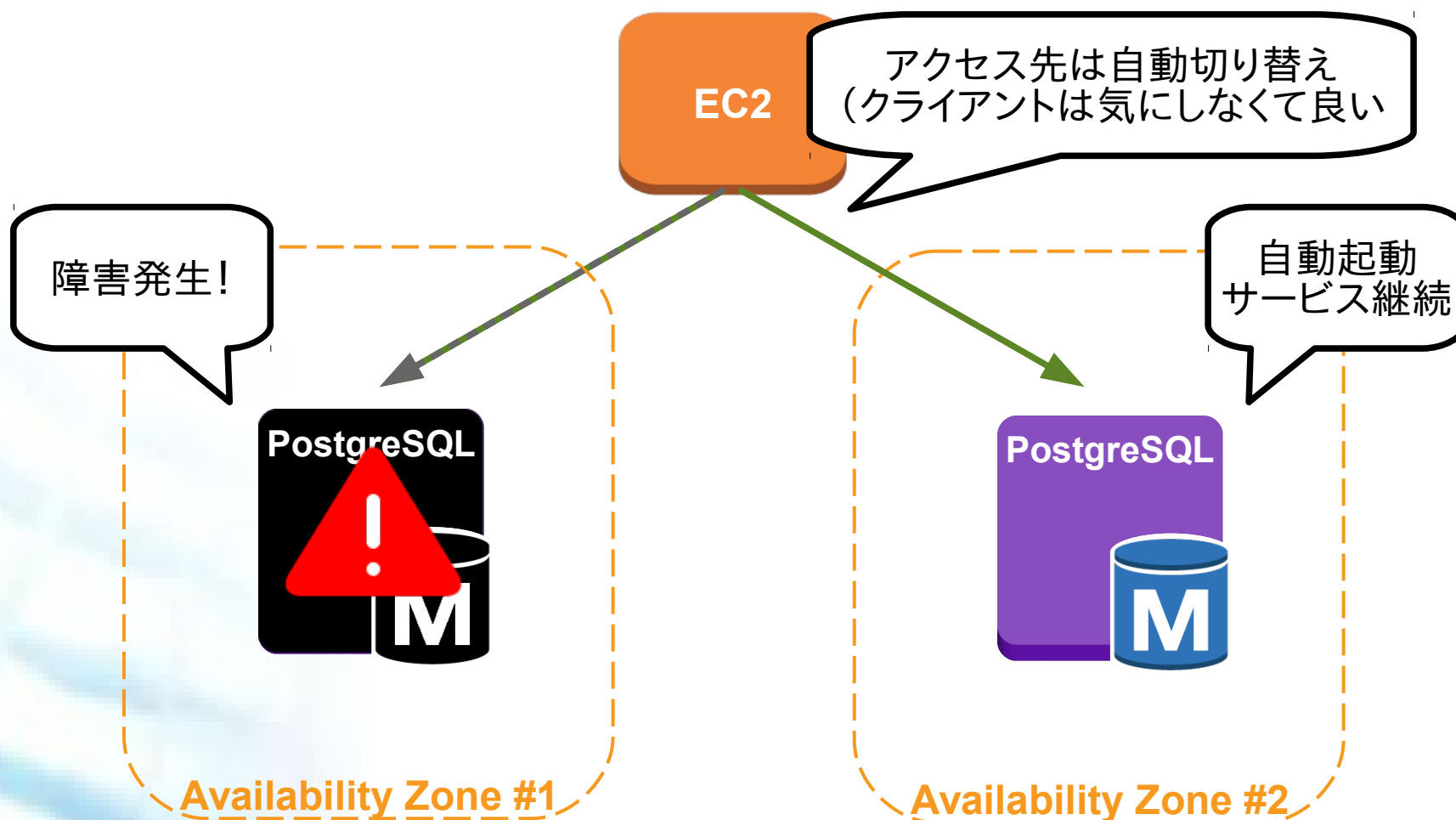
- 耐障害性・可用性を向上させる **AWS RDS の標準機能**
- 異なるアベイラリティゾーン(≒データセンタ)に待機系サーバを作成
- 同期物理レプリケーションと自動フェイルオーバー



- 耐障害性・可用性を向上させる **AWS RDS の標準機能**
- 異なるアベイラリティゾーン(≒データセンタ)に待機系サーバを作成
- **同期物理レプリケーション**と自動フェイルオーバー



- 耐障害性・可用性を向上させる **AWS RDS の標準機能**
- 異なるアベイラリティゾーン(≒データセンタ)に待機系サーバを作成
- 同期物理レプリケーションと**自動フェイルオーバー**



- RDS インスタンス作成時に必要な項目を設定

 Services ▾ Edit ▾

▶

Step 1: Engine Selection

Step 2: Production?

Step 3: DB Instance Details

Step 4: Additional Config

Step 5: Management Options

Step 6: Review

DB Instance Details

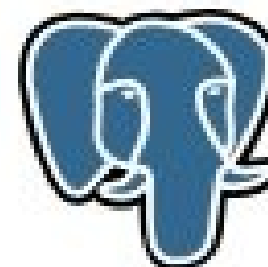
To get started, choose a DB engine below and click Next Step

DB Engine:	postgres
License Model:	postgresql-license ▾
DB Engine Version:	9.3.1 ▾
DB Instance Class:	db.t1.micro ▾
Multi-AZ Deployment:	Yes ▾
Auto Minor Version Upgrade:	☒ Yes ☐ No

Provide the details for your RDS Database Instance.

Allocated Storage:*	5	(Minimum: 5 GB, Maximum: 3072 GB) Higher allocated storage may improve IOPS performance.
Use Provisioned IOPS:	☐	
DB Instance Identifier:*	test-gui	(e.g. mydbinstance)
Master Username:*	sraoss	(e.g. awsuser)
Master Password:*	●●●●●●●●	(e.g. mypassword)

[Cancel](#) [Previous](#) [Next Step](#)



(2014/01/14 時点)
PostgreSQL 9.3.1

- RDS インスタンスのスペックを選択できる
- 後からインスタンスクラスは簡単に変更可能

DB インスタンスクラス	CPU	メモリ
マイクロ	最大 2 ECU	630 MB
スモール	1 ECU × 1 仮想コア	1.7 GB
ミディアム	2 ECU × 1 仮想コア	3.75 GB
ラージ	2 ECU × 2 仮想コア	7.5 GB
エクストララージ	2 ECU × 4 仮想コア	15 GB
ハイメモリエクストララージ	3.25 ECU × 2 仮想コア	17.1 GB
ハイメモリダブルエクストララージ	3.25 ECU × 4 仮想コア	34 GB
ハイメモリクアドラプルエクストララージ	3.25 ECU × 8 仮想コア	68 GB
ハイメモリクラスタエイトエクストララージ	88 ECU (32 ハイパースレッドコア)	244 GB

- DB 名・ポート・ネットワーク構成・パラメータ・セキュリティグループ etc.

 Services ▾ Edit ▾

▶

Step 1: Engine Selection

▶

Step 2: Production?

▶

Step 3: DB Instance Details

▶

Step 4: Additional Config

▶

Step 5: Management Options

▶

Step 6: Review

Additional Config

Provide the optional additional configuration details below.

Database Name: (e.g. mydb)

Database Port:

Choose a VPC:

DB Subnet Group:

Publicly Accessible: ☒ Yes ☐ No

Availability Zone: MultiAZ deployment selected

Option Group:

If you have custom DB Parameter Groups or DB Security Groups you would like to associate with this DB Instance, select them below, otherwise proceed with default settings.

Parameter Group:

VPC Security Group(s):

[Cancel](#) [Previous](#) [Next Step](#)

- 自動バックアップを行う時間帯
- 自動バックアップデータの保存期間は？
- メンテナンスを行う時間帯 (0.5h, 1.0h, 1.5h, ... 3.0h)

 Services ▾ Edit ▾

▶

Step 1: Engine Selection

Step 2: Production?

Step 3: DB Instance Details

Step 4: Additional Config

Step 5: Management Options

Step 6: Review

Management Options

Enabled Automatic Backups: ☒ Yes ☐ No

The number of days for which automated backups are retained.

Backup Retention Period: 1 ▾ days

The daily time range during which automated backups are created if automated backups are enabled

Backup Window: ☒ Select Window ☐ No Preference

Start Time 01 ▾ : 00 ▾ UTC

Duration : 0.5 ▾ hours

The weekly time range (in UTC) during which system maintenance can occur.

Maintenance Window: ☒ Select Window ☐ No Preference

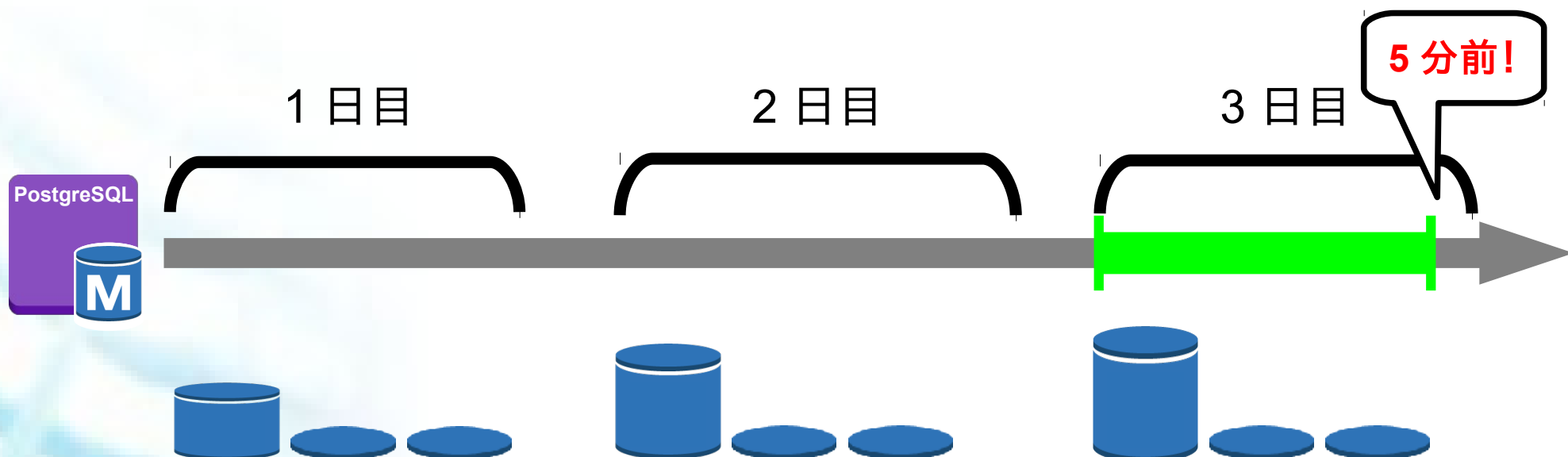
Start Time Monday ▾ 00 ▾ : 00 ▾ UTC

Duration : 0.5 ▾ hours

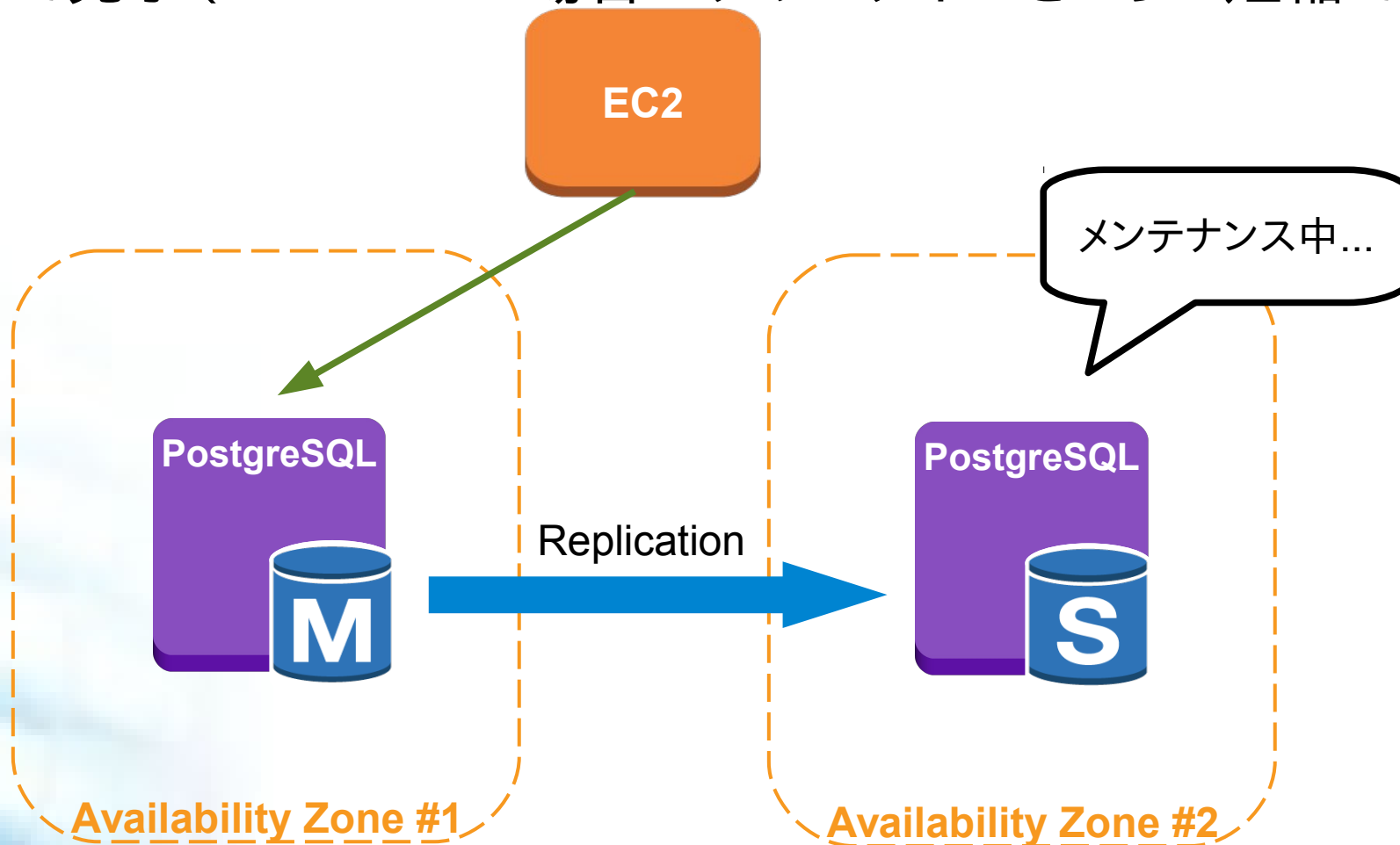
[Cancel](#) [Previous](#) [Next Step](#)

- 1 日 1 回の RDS インスタンスのスナップショット + その日の差分情報
- 取得したバックアップデータを 0 ～ 35 日間保存できる
- 保存期間内の任意の時点に RDS をリカバリできる
- ただし最新の復元時点は現在時刻の **5 分前**
 - 差分情報は 5 分置きに取得されるため

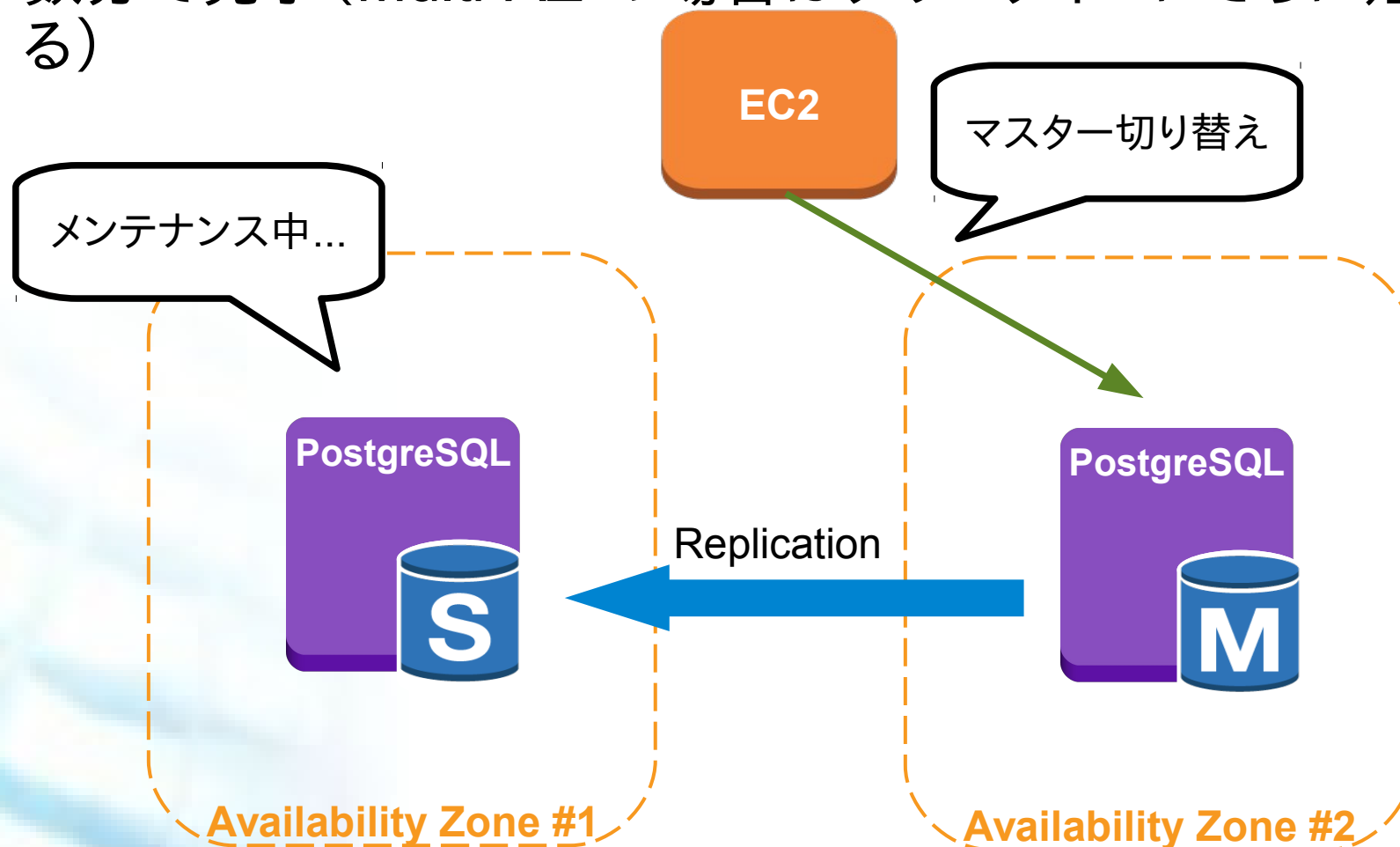
例: Backup Retention Period = 1 days



- AWS 側がソフトウェアパッチを当ててもよい時間を指定
- 数ヶ月に 1 度あるかないか
- RDS インスタンスが再起動することがある
- 数分で完了 (Multi-AZ の場合はダウンタイムをさらに短縮できる)



- AWS 側がソフトウェアパッチを当ててもよい時間を指定
- 数ヶ月に 1 度あるかないか
- RDS インスタンスが再起動することがある
- 数分で完了 (Multi-AZ の場合はダウンタイムにさらに短縮できる)



- PostgreSQL 9.0.14/9.2.5/**9.3.1** でバグ
 - ・ マイナーアップデート (2013/12/05) があった
- RDS for PostgreSQL のメンテナンスあるかも

Text Size: Normal / Large | Donate | Contact |

 PostgreSQL

The world's most advanced open source database.

Home | About | Download | Documentation | Community | Developers | Support | Your account

- » Community
- » Contributors
- » Mailing Lists
 - Subscribe
 - User lists
 - Developer lists
 - pgsql-cluster-hackers
 - pgsql-committers
 - pgsql-hackers
 - pgsql-rrreviewers
 - pgsql-www
 - Regional lists
 - Associations
 - User groups
 - Project lists
 - Inactive lists
- » IRC

Data corruption issues using streaming replication on 9.0.14/9.2.5/9.3.1

From: Christophe Pettus <xof(at)thebuild(dot)com>
To: PostgreSQL-development Hackers <pgsql-hackers(at)postgresql(dot)org>
Subject: Data corruption issues using streaming replication on 9.0.14/9.2.5/9.3.1
Date: 2013-11-18 18:58:26
Message-ID: 0E76EE0A-1740-41B5-88FF-54AA98794532@thebuild.com (view [raw](#) or [flat](#))
Thread: 2013-11-18 18:58:26 from Christophe Pettus <xof(at)thebuild(dot)com>
Lists: [pgsql-hackers](#)

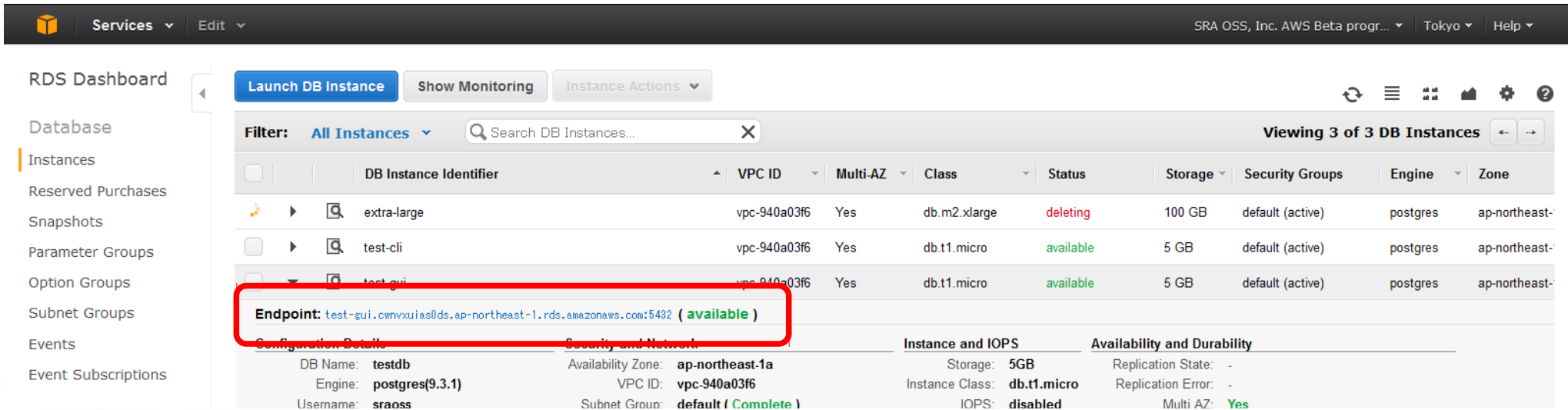
Three times in the last two weeks, we have experience data corruption secondary servers using streaming replication on client systems. The versions involved are 9.0.14, 9.2.5, and 9.3.1. Each incident was separate; two cases they were for the same client (9.0.14 and 9.3.1), one for a different client (9.2.5).

The details of each incident are similar, but not identical.
The details of each incident are:

INCIDENT #1: 9.0.14 -- A new secondary (S1) was initialized using rsync off of an existing, corrupt primary (P1) for the base backup, and using WAL-R for WAL segment shipping. Both the

<http://www.postgresql.org/message-id/0E76EE0A-1740-41B5-88FF-54AA98794532@thebuild.com>

- 簡単に作成できる
- 作成した DB のアクセスは Endpoint から



The screenshot shows the AWS RDS Dashboard. On the left is a navigation menu with options like 'Instances', 'Reserved Purchases', 'Snapshots', etc. The main area displays a table of database instances. The instance 'test-gui' is selected, and its details are shown below the table. The endpoint is highlighted with a red box.

DB Instance Identifier	VPC ID	Multi-AZ	Class	Status	Storage	Security Groups	Engine	Zone
extra-large	vpc-940a03f6	Yes	db.m2.xlarge	deleting	100 GB	default (active)	postgres	ap-northeast-1
test-cli	vpc-940a03f6	Yes	db.t1.micro	available	5 GB	default (active)	postgres	ap-northeast-1
test-gui	vpc-940a03f6	Yes	db.t1.micro	available	5 GB	default (active)	postgres	ap-northeast-1

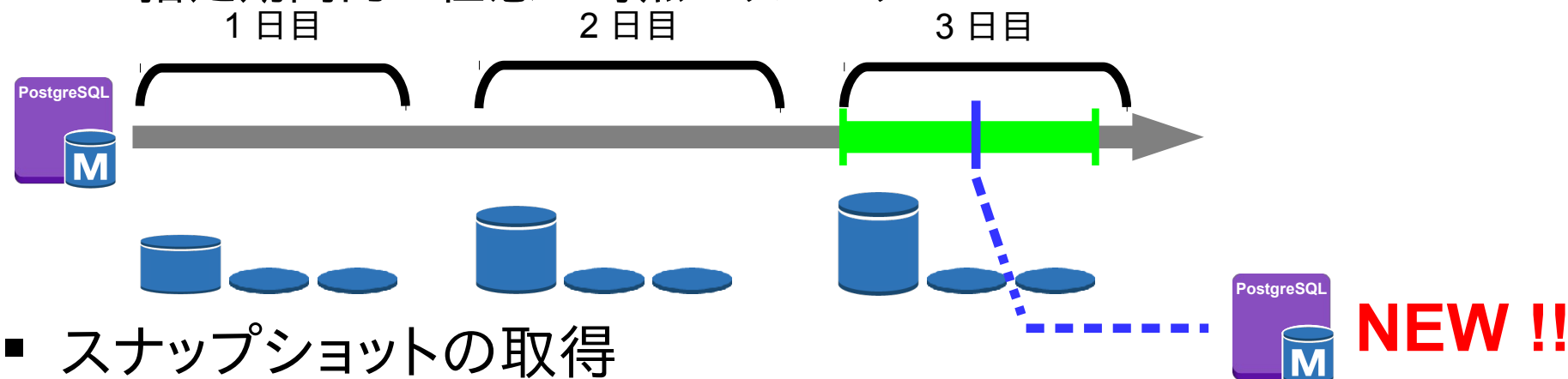
Endpoint: test-gui.cwnvxuias0ds.ap-northeast-1.rds.amazonaws.com:5432 (available)

Configuration Details		Security and Network		Instance and IOPS		Availability and Durability	
DB Name:	testdb	Availability Zone:	ap-northeast-1a	Storage:	5GB	Replication State:	-
Engine:	postgres(9.3.1)	VPC ID:	vpc-940a03f6	Instance Class:	db.t1.micro	Replication Error:	-
Username:	sraoss	Subnet Group:	default (Complete)	IOPS:	disabled	Multi AZ:	Yes

- 100 GB / 1000 IPOS で 15 分程度

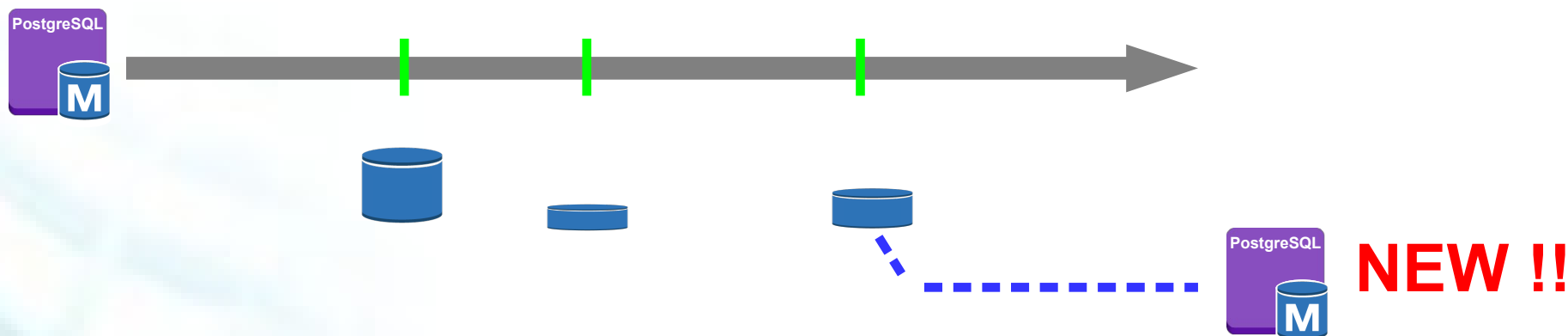
■ 自動バックアップ

- 指定期間内の任意の時点にリカバリ



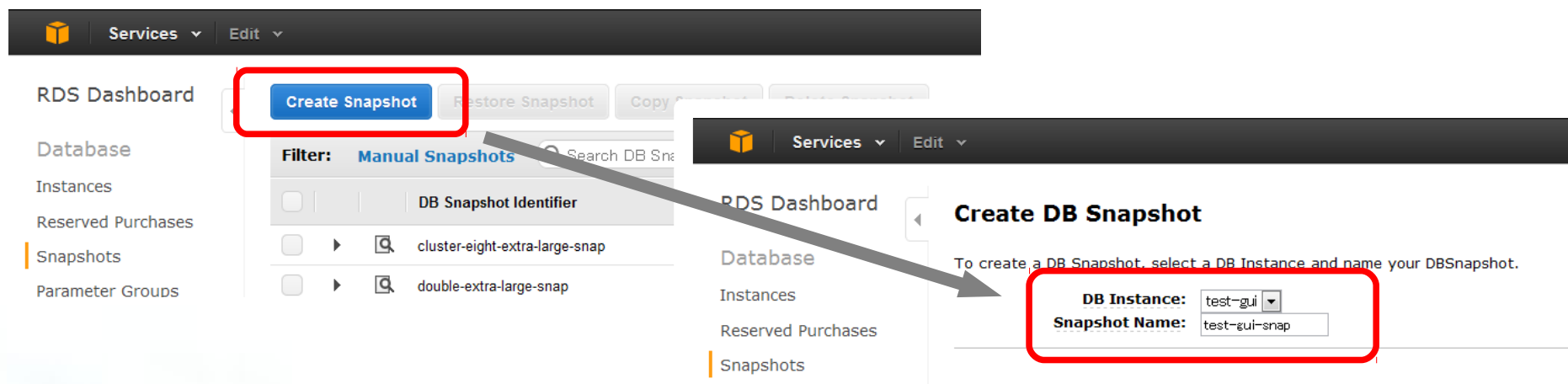
■ スナップショットの取得

- 取得時点に RDS インスタンスをリストア (複数取得=差分取得)



- どちらも**新しいインスタンスを作成することになる**
- リカバリ/リストア時に**インスタンススペックを変更できる**

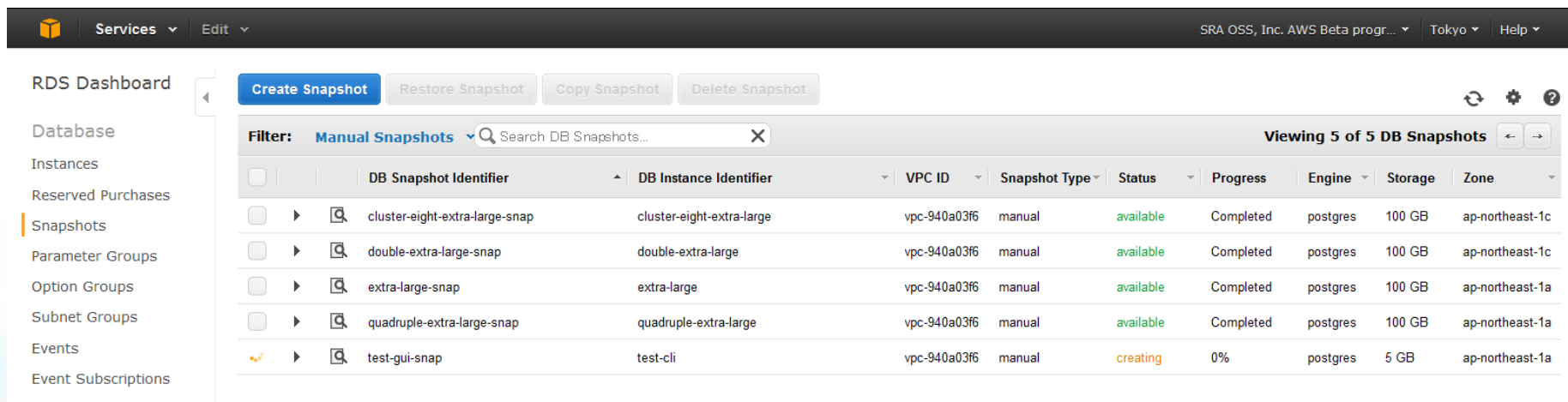
- Create Snapshot
- 取得したい RDS インスタンスを選択してスナップショット名を指定



- スナップショットの取得時間は 7 分程度

DB インスタンスクラス	使用量/ストレージサイズ	取得時間
ハイメモリエクストララージ	15 GB / 100 GB	7 分
ハイメモリダブルエクストララージ	15 GB / 100 GB	9 分
ハイメモリクラスタエイトエクストララージ	15 GB / 100 GB	6 分

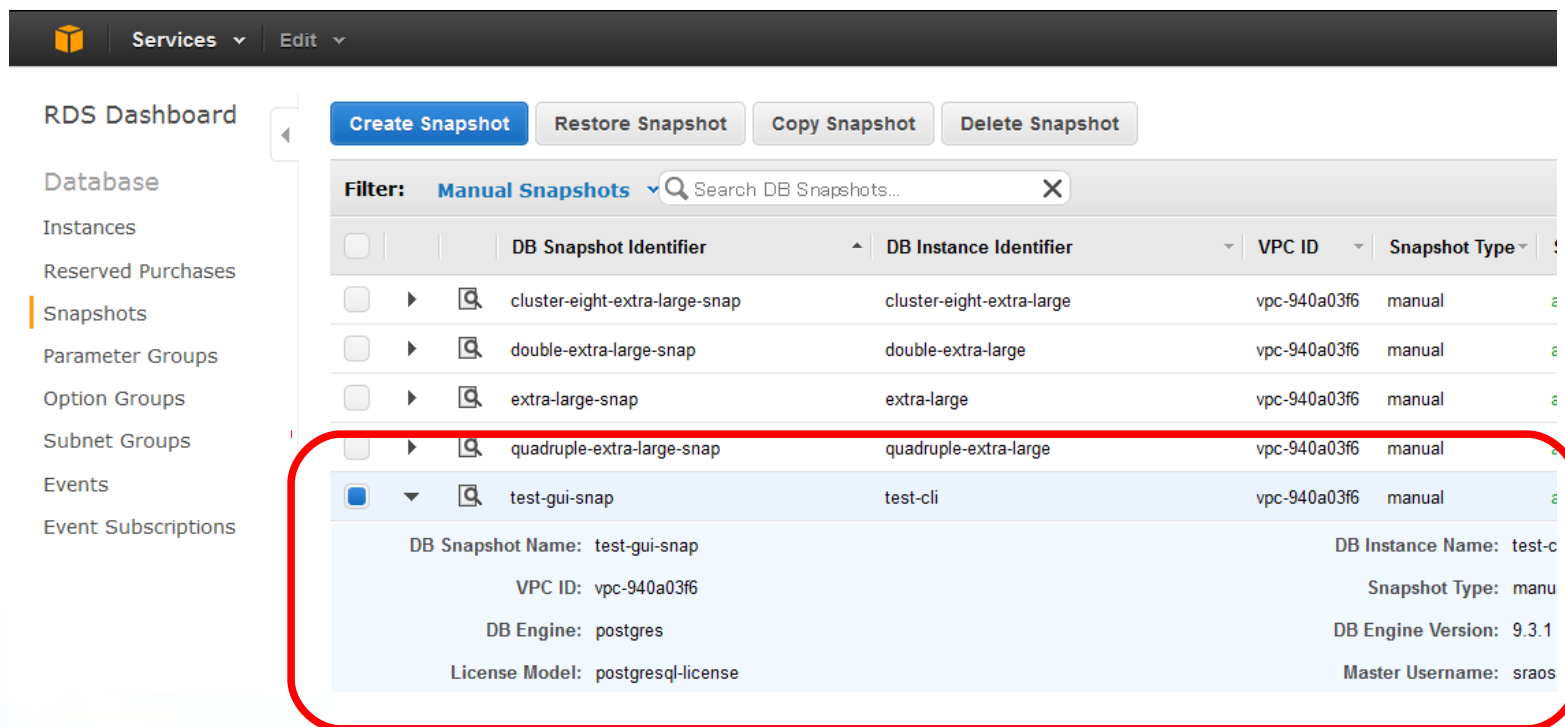
- スナップショットは複数取得可能
- スナップショットのコピーもできる
 - コピー先を異なる AZ に配置してさらに安心
- Multi-AZ を有効にしていないと数分間 DB に接続できなくなる



The screenshot shows the AWS RDS console interface. On the left is a navigation menu with options like 'Instances', 'Reserved Purchases', 'Snapshots', 'Parameter Groups', 'Option Groups', 'Subnet Groups', 'Events', and 'Event Subscriptions'. The 'Snapshots' option is highlighted. The main panel displays a table of database snapshots. At the top of the panel are buttons for 'Create Snapshot', 'Restore Snapshot', 'Copy Snapshot', and 'Delete Snapshot'. Below these is a filter bar set to 'Manual Snapshots' with a search input. The table lists five snapshots, with the last one, 'test-gui-snap', currently in a 'creating' state.

	DB Snapshot Identifier	DB Instance Identifier	VPC ID	Snapshot Type	Status	Progress	Engine	Storage	Zone
☐	cluster-eight-extra-large-snap	cluster-eight-extra-large	vpc-940a03f6	manual	available	Completed	postgres	100 GB	ap-northeast-1c
☐	double-extra-large-snap	double-extra-large	vpc-940a03f6	manual	available	Completed	postgres	100 GB	ap-northeast-1c
☐	extra-large-snap	extra-large	vpc-940a03f6	manual	available	Completed	postgres	100 GB	ap-northeast-1a
☐	quadruple-extra-large-snap	quadruple-extra-large	vpc-940a03f6	manual	available	Completed	postgres	100 GB	ap-northeast-1a
☒	test-gui-snap	test-cli	vpc-940a03f6	manual	creating	0%	postgres	5 GB	ap-northeast-1a

■ リストアしたいスナップショットを選択



RDS Dashboard

Database

Instances

Reserved Purchases

Snapshots

Parameter Groups

Option Groups

Subnet Groups

Events

Event Subscriptions

Create Snapshot Restore Snapshot Copy Snapshot Delete Snapshot

Filter: Manual Snapshots Search DB Snapshots...

	DB Snapshot Identifier	DB Instance Identifier	VPC ID	Snapshot Type
☐	cluster-eight-extra-large-snap	cluster-eight-extra-large	vpc-940a03f6	manual
☐	double-extra-large-snap	double-extra-large	vpc-940a03f6	manual
☐	extra-large-snap	extra-large	vpc-940a03f6	manual
☐	quadruple-extra-large-snap	quadruple-extra-large	vpc-940a03f6	manual
☒	test-gui-snap	test-cli	vpc-940a03f6	manual

DB Snapshot Name: test-gui-snap DB Instance Name: test-c

VPC ID: vpc-940a03f6 Snapshot Type: manu

DB Engine: postgres DB Engine Version: 9.3.1

License Model: postgresql-license Master Username: sraos

- リストア時に**インスタンスのスペックを変更できる!**

 Services ▾ Edit ▾

RDS Dashboard

Database

Instances

Reserved Purchases

Snapshots

Parameter Groups

Option Groups

Subnet Groups

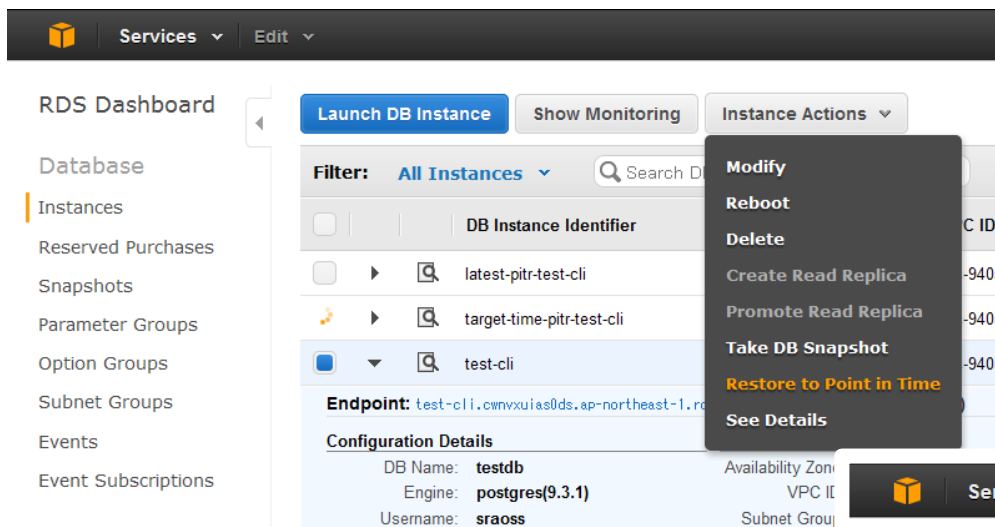
Events

Event Subscriptions

Restore DB Instance

You are creating a new DB Instance from a source DB Instance at a specified time. This new DB Instance will have the default DB Security Group and DB Parameter Groups.

DB Snapshot ID:	test-gui-snap
DB Instance Identifier:*	restore-test-gui-sr (e.g. mydbinstance)
DB Engine:	postgres ▾
License Model:	Postgresql License ▾
DB Instance Class:	db.m1.small ▾
Multi-AZ Deployment:	Yes ▾
Auto Minor Version Upgrade:	☒ Yes ☐ No
Database Name:	restoredb (e.g. mydb)
Database Port:	5432
Storage Type:	Standard ▾ postgres supports provisioned storage from 100 to 3072 GB
Choose a VPC:	Default VPC (vpc-940a03f6) ▾
DB Subnet Group:	default ▾
Publicly Accessible:	☒ Yes ☐ No
Availability Zone:	No Preference ▾
Option Group:	default:postgres-9-3 ▾



- リストアしたい RDS インスタンスを選択

RDS Dashboard

Database

Instances

Reserved Purchases

Snapshots

Parameter Groups

Option Groups

Subnet Groups

Events

Event Subscriptions

Restore DB Instance

You are creating a new DB Instance from a source DB Instance at a specified time. This new DB Instance will have the default DB Security Group and DB Parameter Groups.

Use Latest Restorable Time: ☒ November 15, 2013 2:12:49 PM UTC+9

Use Custom Restore Time: ☐ MMMM d, y : : UTC+9

Source DB Instance: test-cli

DB Instance Identifier: * gui-pitr-test-cli (e.g. mydbinstance)

DB Engine: postgres

License Model: PostgreSQL License

DB Instance Class: db.t1.micro

Multi-AZ Deployment: Yes

Auto Minor Version Upgrade: ☒ Yes ☐ No

Database Port: 5432

Storage Type: Standard postgres supports provisioned storage from 100 to 3072 GB

Choose a VPC: Default VPC (vpc-940a03f6)

DB Subnet Group: default

Publicly Accessible: ☒ Yes ☐ No

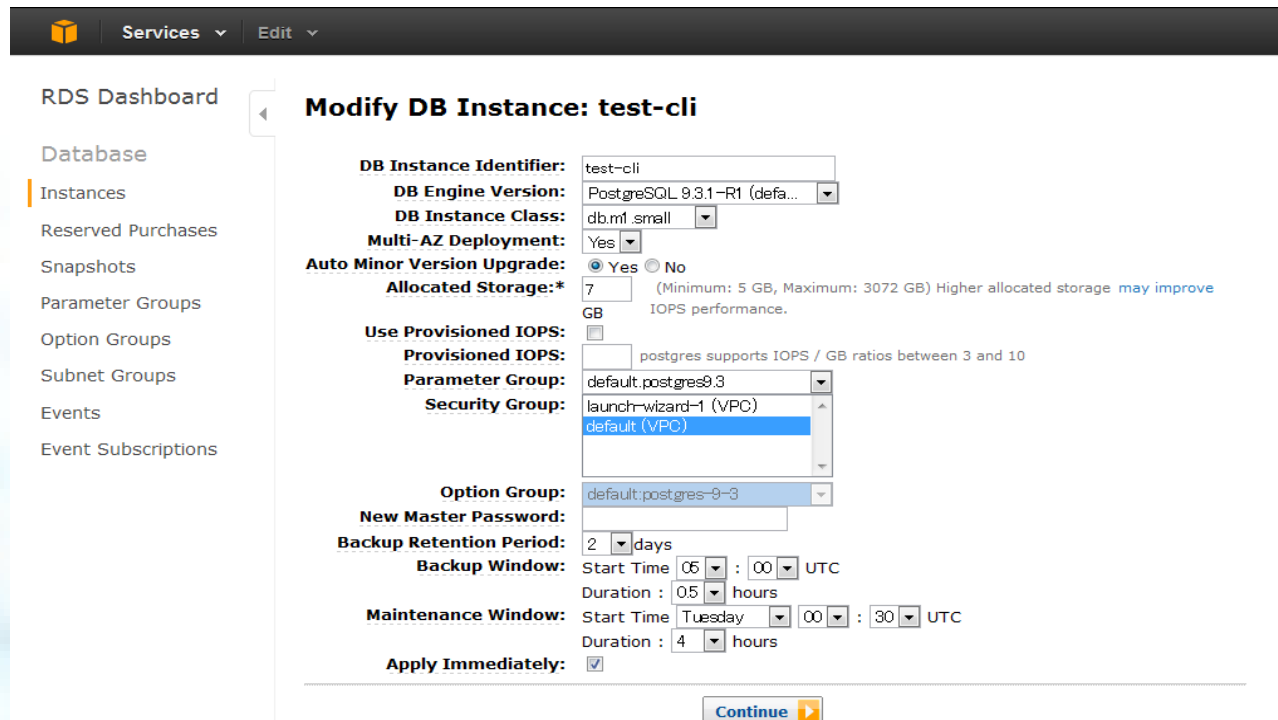
Availability Zone: No Preference MultiAZ deployment selected

Option Group: default:postgres-9-3

- リカバリしたい時刻を選択
- リカバリ時に
インスタンスのスペックを
変更できる!

- スケールアップのタイミング
 - 自動バックアップからのリカバリ
 - スナップショットからのリストア
 - **任意のタイミングで**
- 任意のタイミングでスケールアップ(**RDS インスタンスの変更**)ができる

- 指定の RDS インスタンスの内容を変更できる
- 新たにインスタンスが作成されるわけではない
- 即時変更反映 OR メンテナンスウィンドウで指定した時間帯
- 数分ほど RDS インスタンスは停止する
- **ストレージサイズ**の増加はできても**減少はできない**

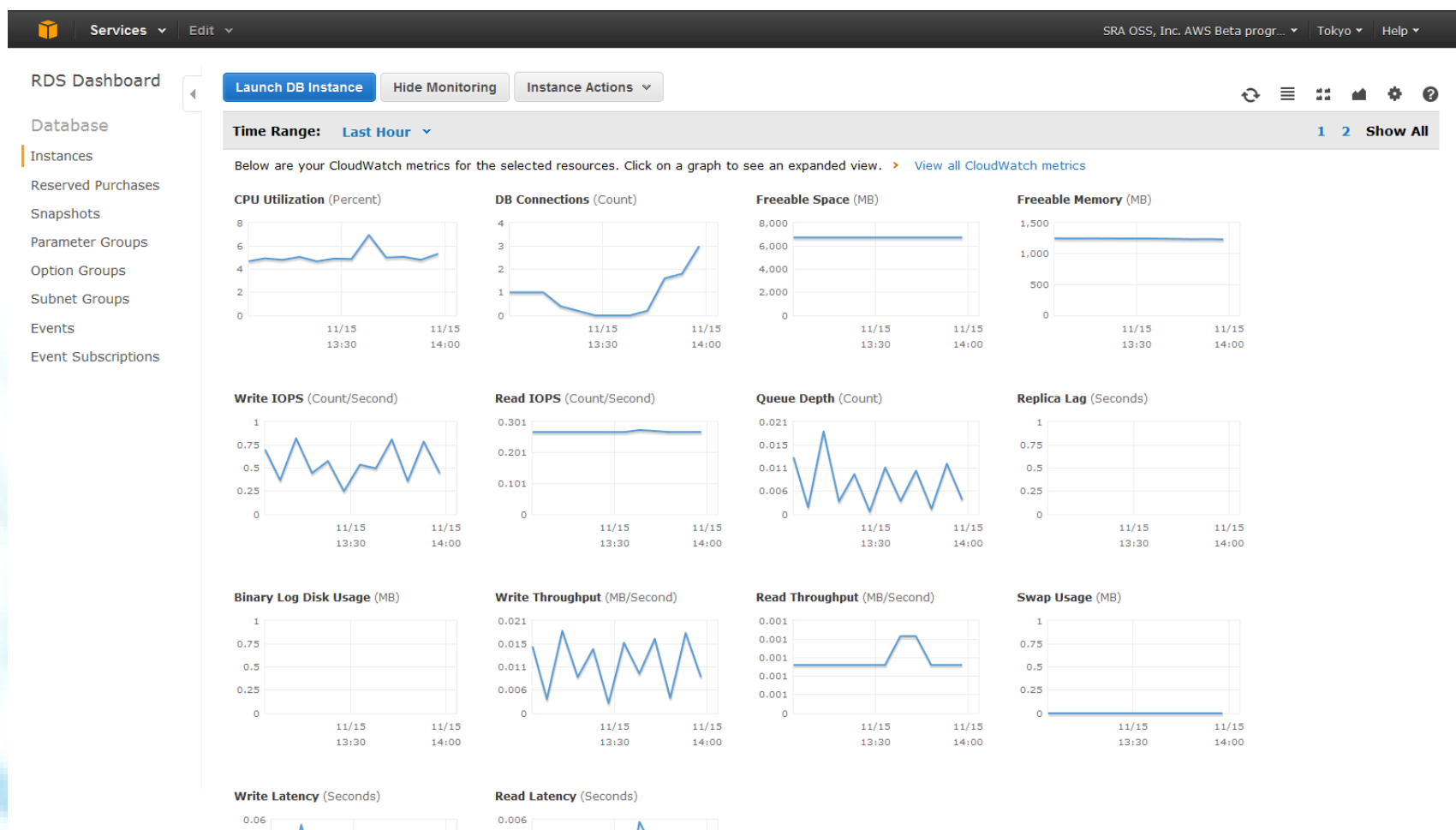


The screenshot shows the 'Modify DB Instance' page in the AWS Management Console for a database instance named 'test-cli'. The left sidebar contains navigation links: RDS Dashboard, Database, Instances (selected), Reserved Purchases, Snapshots, Parameter Groups, Option Groups, Subnet Groups, Events, and Event Subscriptions. The main content area is titled 'Modify DB Instance: test-cli' and contains the following configuration options:

- DB Instance Identifier:** test-cli
- DB Engine Version:** PostgreSQL 9.3.1-R1 (default)
- DB Instance Class:** db.m1.small
- Multi-AZ Deployment:** Yes
- Auto Minor Version Upgrade:** Yes
- Allocated Storage:** 7 GB (Minimum: 5 GB, Maximum: 3072 GB). Higher allocated storage may improve IOPS performance.
- Use Provisioned IOPS:** No
- Provisioned IOPS:** postgres supports IOPS / GB ratios between 3 and 10
- Parameter Group:** default:postgres9.3
- Security Group:** launch-wizard-1 (VPC), default (VPC)
- Option Group:** default:postgres-9-3
- New Master Password:** (empty field)
- Backup Retention Period:** 2 days
- Backup Window:** Start Time: 06:00 UTC, Duration: 0.5 hours
- Maintenance Window:** Start Time: Tuesday 00:30 UTC, Duration: 4 hours
- Apply Immediately:** Yes

A 'Continue' button is located at the bottom right of the form.

- Web コンソールから 2 週間分の RDS インスタンスの統計情報を確認可能
- この統計情報は、**Amazon CloudWatch** (AWS クラウドリソースのモニタリング機能) を使っている



- Web コンソールから 2 週間分の RDS インスタンスの統計情報を確認可能
- この統計情報は、**Amazon CloudWatch** (AWS クラウドリソースのモニタリング機能) を使っている

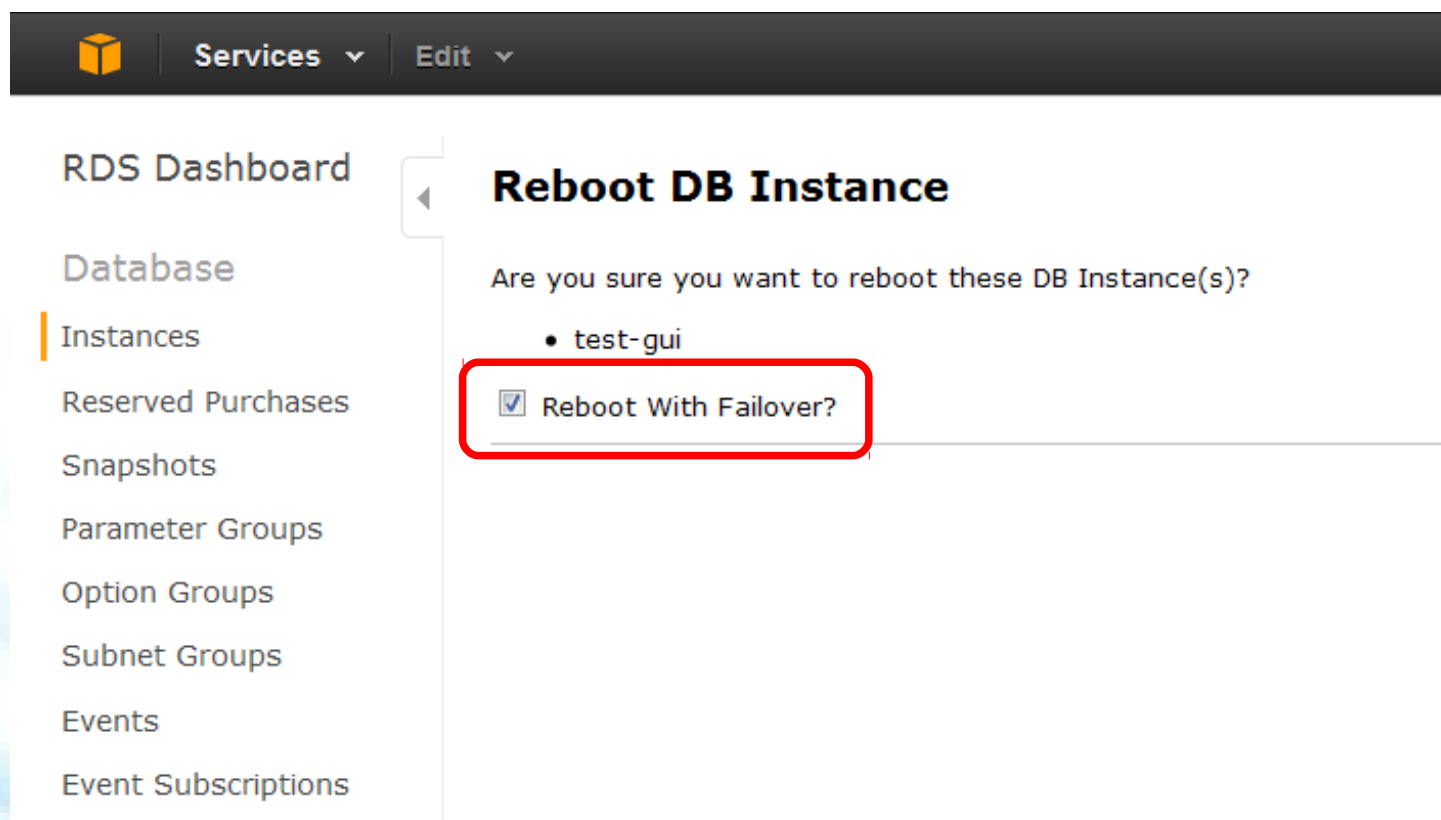
統計情報	内容
CPUUtilization (%)	CPU 使用率
FreeableMemory (Bytes)	利用可能なメモリ量
FreeStorageSpace (Bytes)	利用可能なディスク量
DatabaseConnections (Count)	データベース接続数
ReadIOPS (Count/Second)	1秒辺りの平均ディスク読み込み回数
ReadLatency (Second)	上記の操作の平均所要時間
WriteIOPS (Count/Second)	1秒辺りの平均ディスク書き込み回数
WriteLatency (Second)	上記の操作の平均所要時間

…… など 13 種類

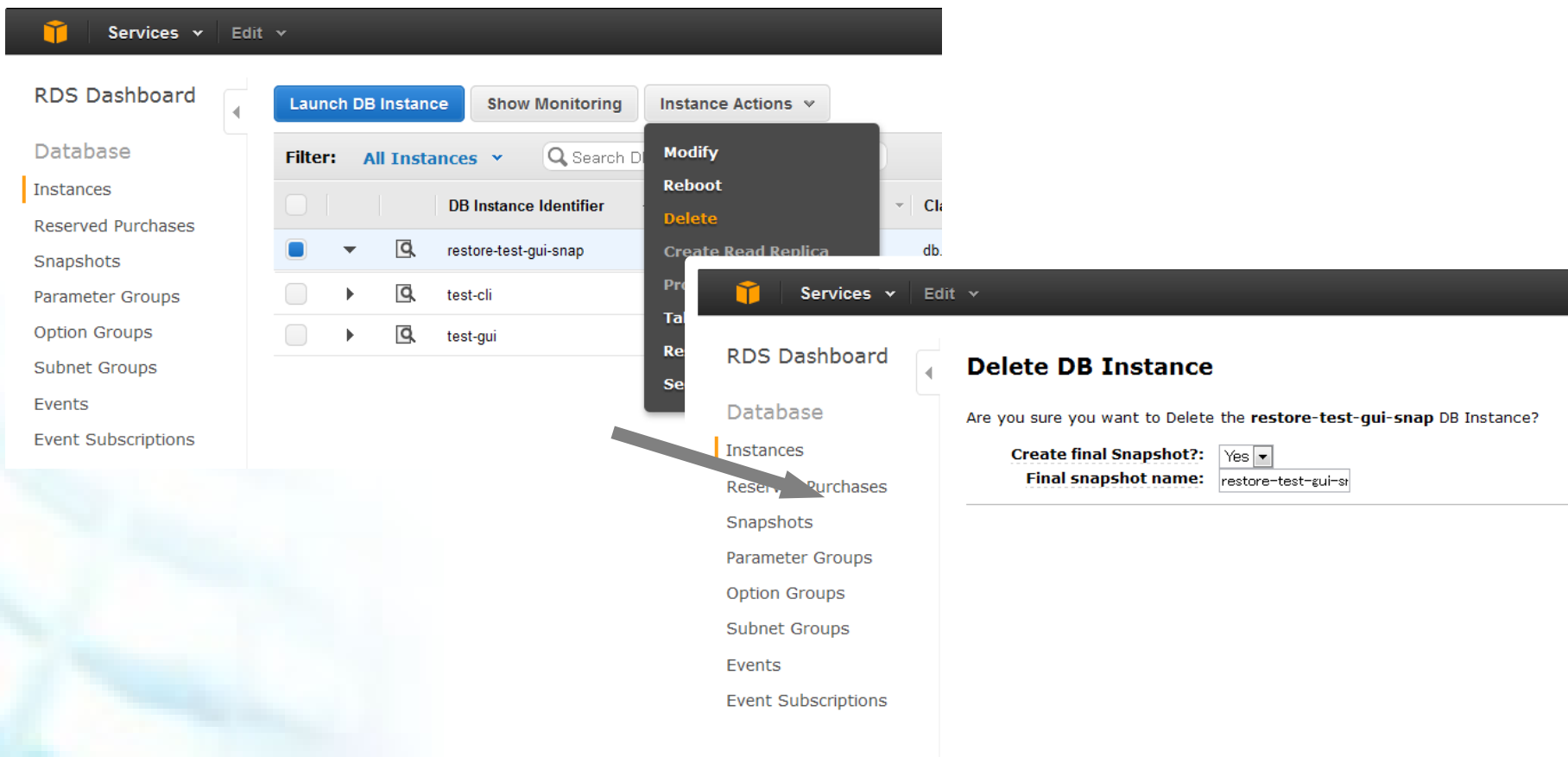
- **Amazon Simple Notification Service** を通して
RDS インスタンスの状態変化をデバイス・アプリに
プッシュ通知

Availability	RDS インスタンスのシャットダウン・再起動
Backup	自動バックアップの開始・終了
Configuration Change	セキュリティグループ、インスタンスのスケールリング、 パスワード変更など
Creation	RDS インスタンス、スナップショットの作成・削除
Failover	フェイルオーバーの開始・完了
Low Storage	ストレージの容量不足
Maintenance	パッチインストールによる オフラインへの移行・オンラインへの復帰
Recovery	データベースインスタンスの復旧
Restoration	ポイント・イン・タイムまたはスナップショットからの RDS インスタンスのリストア

- Multi-AZ を有効にしていると、障害発生時に自動的にスタンバイ側に切り替わりサービス継続
- RDS インスタンスの再起動時にもフェイルオーバーを任意に選択できる



- インスタンスを選択して削除
- 削除時の最終スナップショットを取っておくことも可能
- DB のデータを手元に持ってくる機能は AWS にはないので PostgreSQL のダンプコマンドを使用する



The screenshot illustrates the steps to delete an Amazon RDS instance. It shows the RDS Dashboard with a list of instances. The 'Instance Actions' menu is open, highlighting the 'Delete' option. A secondary screenshot shows the 'Delete DB Instance' confirmation screen, where the user is prompted to create a final snapshot. The 'Create final Snapshot?' checkbox is checked, and the 'Final snapshot name' is set to 'restore-test-gui-snap'.

RDS Dashboard

Database

Instances

Reserved Purchases

Snapshots

Parameter Groups

Option Groups

Subnet Groups

Events

Event Subscriptions

Launch DB Instance

Show Monitoring

Instance Actions

Filter: All Instances

Search DB Instance Identifier

DB Instance Identifier

restore-test-gui-snap

test-cli

test-gui

Modify

Reboot

Delete

Create Read Replica

RDS Dashboard

Database

Instances

Reserved Purchases

Snapshots

Parameter Groups

Option Groups

Subnet Groups

Events

Event Subscriptions

Delete DB Instance

Are you sure you want to Delete the **restore-test-gui-snap** DB Instance?

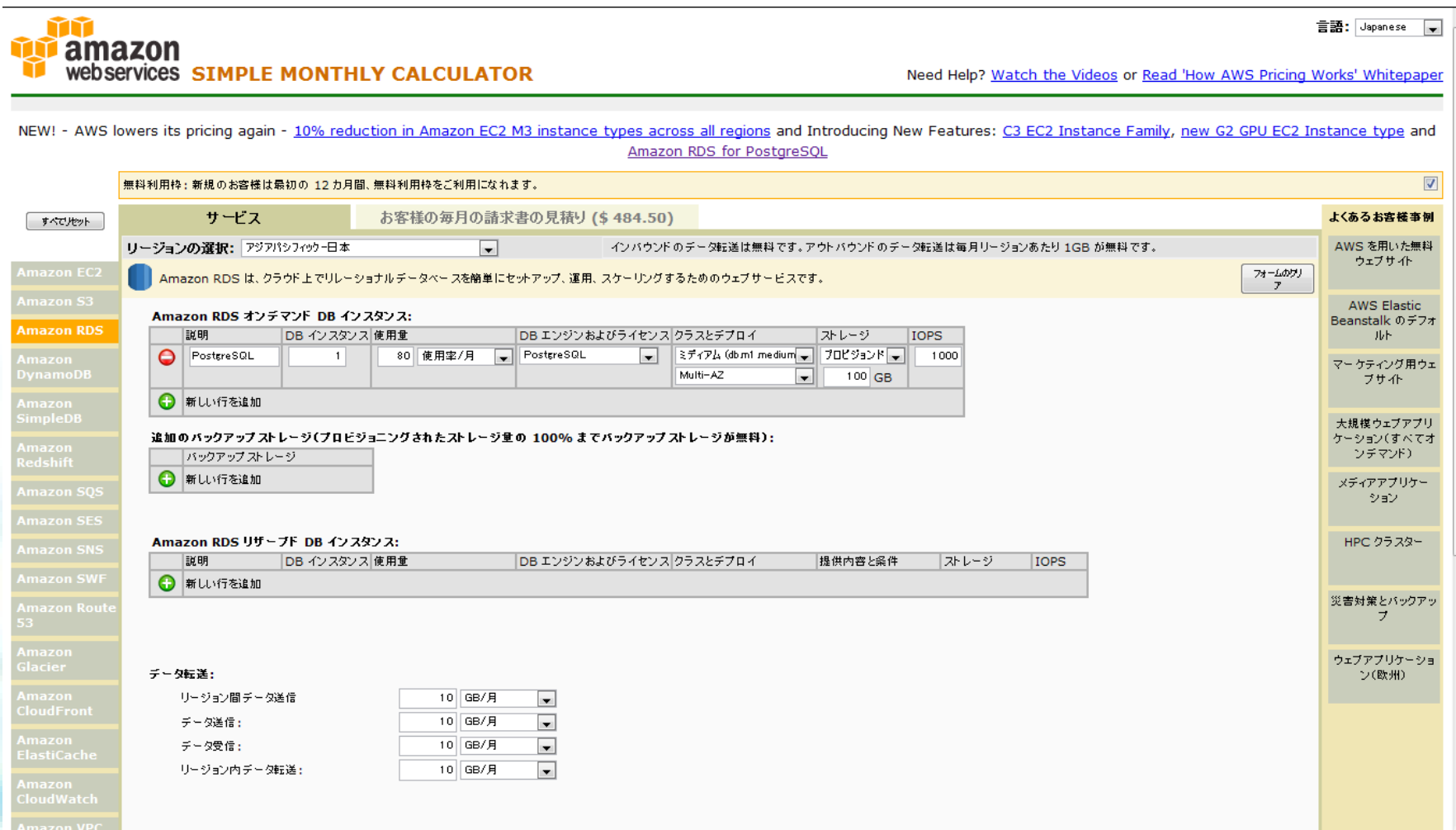
Create final Snapshot?: ☒ Yes

Final snapshot name: restore-test-gui-snap

- 以上の操作は Web コンソールだけでなく、各種 API、スクリプトでも実行可能。
- [AWS SDKs & Tools]
 - <https://aws.amazon.com/jp/tools/>



- 以下のサイトで試算可能
- [Amazon Web Services Simple Monthly Calculator]
 - <http://calculator.s3.amazonaws.com/calc5.html>



amazon web services SIMPLE MONTHLY CALCULATOR

Need Help? [Watch the Videos](#) or [Read 'How AWS Pricing Works' Whitepaper](#)

NEW! - AWS lowers its pricing again - [10% reduction in Amazon EC2 M3 instance types across all regions](#) and Introducing New Features: [C3 EC2 Instance Family](#), [new G2 GPU EC2 Instance type](#) and [Amazon RDS for PostgreSQL](#)

無料利用枠: 新規のお客様は最初の 12 カ月間、無料利用枠をご利用になれます。

サービス お客様の毎月の請求書の見積り (\$ 484.50)

リージョンの選択: アジアパシフィック-日本

インバウンドのデータ転送は無料です。アウトバウンドのデータ転送は毎月リージョンあたり 1GB が無料です。

Amazon RDS は、クラウド上でリレーショナルデータベースを簡単にセットアップ、運用、スケーリングするためのウェブサービスです。

Amazon RDS オンデマンド DB インスタンス:

説明	DB インスタンス	使用量	DB エンジンおよびライセンス	クラスとデプロイ	ストレージ	IOPS
PostgreSQL	1	80 使用率/月	PostgreSQL	ミディアム (db.m1.medium) Multi-AZ	プロビジョンド 100 GB	1000

新しい行を追加

追加のバックアップストレージ(プロビジョニングされたストレージ量の 100% までバックアップストレージが無料):

バックアップストレージ

新しい行を追加

Amazon RDS リザーブド DB インスタンス:

説明	DB インスタンス	使用量	DB エンジンおよびライセンス	クラスとデプロイ	提供内容と条件	ストレージ	IOPS
新しい行を追加							

データ転送:

リージョン間データ送信: 10 GB/月

データ送信: 10 GB/月

データ受信: 10 GB/月

リージョン内データ転送: 10 GB/月

よくあるお客様事例

- AWS を用いた無料ウェブサイト
- AWS Elastic Beanstalk のデフォルト
- マーケティング用ウェブサイト
- 大規模ウェブアプリケーション(すべてオンデマンド)
- メディアアプリケーション
- HPC クラスター
- 災害対策とバックアップ
- ウェブアプリケーション(欧州)

■ [ドキュメント]

- <http://aws.amazon.com/jp/documentation/>
- 各サービスのドキュメント
- 未翻訳のものも多い

■ [AWS SDKs & Tools]

- <http://aws.amazon.com/jp/tools/>
- 各種ツールのインストール・使い方

■ [AWS クラウドサービス活用資料集]

- <http://aws.amazon.com/jp/aws-jp-introduction/>
- スライド資料
- わかりやすい概要、事例紹介など

オープンソースとともに



URL: <http://www.sraoss.co.jp/>
E-mail: sales@sraoss.co.jp
Tel: 03-5979-2701