

FUJITSU Cloud Service for OSS IaaS

Database Service User Guide

Version 1.7

Nov. 22, 2018

FUJITSU LIMITED

K51A-DC-M-012-001E

- The duplication or reproduction of this manual without permission is prohibited.
- Note that the content of this manual may change without prior notice.

Preface

Purpose of This Manual

This document describes the basic operation procedures for building and operating the IaaS Database Service (hereinafter referred to as the database service) provided by FUJITSU Cloud Service for OSS IaaS (hereinafter referred to as IaaS).

This document is intended for Japan East 1/Japan East 2/Japan West 1/Japan West 2 region.

Readers for This Manual

This document is intended for those who are involved in the operation of the database service.

This document is intended for those who have basic knowledge of the following:

- IaaS service operations
- PostgreSQL
- Linux

Organization of Manuals

Refer to the related manuals listed below according to your purposes and methods of use.

| Manual Title | Purposes and Methods of Use |
|-----------------------------------|--|
| Features Handbook | This document explains the functions provided by this service in detail. |
| API User Guide | This document provides instructions on how to use the REST API, including how to build an API execution environment and how to use a sample script that suits the sequence you use. |
| API Reference Manual | Refer to this document for detailed information about how to use the REST API. |
| IaaS Service Portal User Guide | This document explains how to use the functions provided by this service via Service Portal (Web GUI). |
| IaaS Heat Template Specifications | This document explains the format of the Heat Orchestration Template (HOT) that you create in order to use the orchestration function. |
| PostgreSQL 9.6.2 Documentation | Refer to the document "PostgreSQL 9.6.2 Documentation" (http://software.fujitsu.com/jp/manual/manualindex/p17000156.html) |
| PostgreSQL 9.2.4 Documentation | Refer to the document "PostgreSQL 9.2.4 Documentation" (http://software.fujitsu.com/jp/manual/manualfiles/m130028/j2ul1779/01enz000/j2ul-1779-01enz0.pdf) |
| Enterprise Postgres manuals | Refer to the following URL to see how the manuals of Enterprise Postgres are organized: http://software.fujitsu.com/jp/manual/manualindex/p17000156.html For details on clients, refer to the "FUJITSU Enterprise Postgres 9.6 Installation and Setup Guide for Client" included with the Enterprise Postgres Client. |
| Symfoware Server manuals | Refer to the following URL to see how the manuals of Symfoware Server are organized: |

| | |
|---|--|
| | http://swpmsn.soft.fujitsu.com/jp/manual/manualfiles/m140025/j2ul1732/05enz200/j2ul-1732-05enz0.pdf For the manuals of the database server and the database client (Symfoware Server Client), contact the support desk. |
| Database Service User Guide (this document) | This document explains in detail the basic operation procedures of the database service provided by this service. |

Trademarks

- Adobe, the Adobe logo, Acrobat, and Reader are either trademarks or registered trademarks of Adobe Systems Incorporated in the United States and/or other countries.
 - Apache and Tomcat are either trademarks or registered trademarks of The Apache Software Foundation in the United States and/or other countries.
 - Microsoft, Windows, and Windows Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.
 - Linux® is a registered trademark of Linus Torvalds in the United States and/or other countries.
 - Red Hat and Red Hat Enterprise Linux are trademarks of Red Hat, Inc. registered in the United States and/or other countries.
 - UNIX is a registered trademark of The Open Group in the United States and/or other countries.
 - Other company names and product names mentioned in this manual are trademarks or registered trademarks of their respective companies.
- In this manual, the registered trademark symbols (™ or ®) next to system names or product names have been omitted.

Export Administration Regulations

When exporting or giving this document to a third party, be sure to familiarize yourself with the regulations related to export administration valid in your country of residence and the United States, and follow the necessary procedures.

Note

- The content of this manual may change without prior notice.
- The reproduction of this manual without permission is prohibited.
- We do not assume responsibility for any violation of patent rights or any other rights of a third party that may occur due to the use of the data in this manual.

Revision History

| Edition | Date of Update | Location | Overview |
|---------|----------------|--|--|
| 1.0 | Jun. 16, 2017 | All | Initial Version |
| 1.0.1 | Aug. 7, 2017 | Figure 1.2 Multi-AZ System Configuration | Corrected descriptions |
| 1.1 | Dec 4, 2017 | Appendix B: Supported Client | Update supported DB client version |
| 1.2 | Dec 15, 2017 | All | Added descriptions regarding DB engines |
| 1.3 | Mar. 1, 2018 | Appendix E: Roles That Can Execute Each API | Added Cancelling a DB Instance Operation |
| 1.4 | Jun. 29, 2018 | Appendix A: List of DB Parameters That Can Be Changed | Changed specifications |
| 1.5 | Aug 30, 2018 | 3.4 Performance Monitoring Appendix A: List of DB Parameters That Can Be Changed | Added descriptions |
| 1.6 | Sep 25, 2018 | 3.4 Performance Monitoring | Added descriptions |
| 1.7 | Nov 22, 2018 | Appendix A: List of DB Parameters That Can Be Changed Appendix H: Creating a Read Replica | Added descriptions |

Contents

| | |
|--|----|
| Preface | 2 |
| Revision History | 4 |
| Contents | 5 |
| Chapter 1 Operation of the IaaS Database Service | 8 |
| Chapter 2 Creation and Connection of DB Instance..... | 11 |
| 2.1 Preparations | 11 |
| (1) Creating a Network/Subnet..... | 11 |
| (2) Creating a Virtual Router | 12 |
| (3) Creating a Security Group | 12 |
| (4) Creating Security Group Rules | 13 |
| 2.2 Creating a DB Instance | 15 |
| (1) Creating a DB Subnet Group | 15 |
| (2) Creating a DB Parameter Group | 17 |
| (3) Creating a DB Instance | 18 |
| 2.3 Connecting to a DB Instance..... | 23 |
| (1) Checking DB Instance Information | 23 |
| (2) Connecting to a DB Instance | 24 |
| 2.4 Registering an Event Notification | 25 |
| (1) Creating an Event Notification Registration | 25 |
| (2) Checking the Event Notification Registration Information..... | 26 |
| Chapter 3 Operation of DB Instance (Daily Operation) | 27 |
| 3.1 Stopping/Starting a DB Instance..... | 27 |
| (1) Stopping a DB Instance..... | 27 |
| (2) Starting a DB Instance | 28 |
| 3.2 Creating a Backup Manually | 28 |
| (1) Creating a DB snapshot | 28 |
| (2) Referring to a DB Snapshot List..... | 29 |
| 3.3 Checking Database Logs..... | 30 |
| (1) Referring to a DB Log File List | 30 |
| (2) Referring to a DB Log File | 32 |
| 3.4 Performance Monitoring | 33 |
| 3.5 Checking an Event Notification | 33 |
| 3.6 Changing the Encryption Key Used for Transparent Data Encryption | 34 |
| Chapter 4 Changes to DB Instance (When Necessary)..... | 35 |
| 4.1 Changing a Flavor ID | 35 |
| 4.2 Changing Data Disk Size..... | 36 |
| 4.3 Changing from a Single Configuration to a Multi Configuration | 37 |
| (1) Creating a New Multi-Configuration DB Instance | 37 |
| (2) Stopping Business Operations | 37 |
| (3) Extracting Data..... | 37 |
| (4) Inserting Data in a Multi-Configuration DB Instance | 38 |

| | |
|--|----|
| (5) Correcting Application | 38 |
| (6) Resuming Business Operations | 38 |
| 4.4 Changing a DB Instance..... | 39 |
| 4.5 Changing a Database Parameter | 40 |
| (1) Creating a DB Parameter Group | 40 |
| (2) Changing a Parameter Value | 40 |
| (3) Checking the DB Parameter Group Information | 41 |
| (4) Applying a DB Parameter Group | 41 |
| (5) Restarting a DB Instance | 42 |
| Chapter 5 Dealing with Errors (Recovery) | 44 |
| 5.1 Recovering a DB Instance..... | 44 |
| (1) Recovering a DB Instance | 44 |
| (2) Checking DB Instance Recovery | 49 |
| (3) Correcting Application | 49 |
| (4) Deleting a Recovery Source DB Instance | 49 |
| 5.2 Recovering a Multi-Configuration DB Instance | 49 |
| (1) Operation when an Error Occurs..... | 49 |
| (2) Auto Recovery of a DB Instance..... | 49 |
| (3) Operation after Auto Recovery | 50 |
| Chapter 6 Deletion of DB Instance | 51 |
| 6.1 Deleting a DB Instance | 51 |
| (1) Deleting a DB Instance | 51 |
| (2) Deleting a DB Parameter Group | 51 |
| (3) Deleting a DB Subnet Group..... | 52 |
| Appendix A: List of DB Parameters That Can Be Changed | 53 |
| A.1 List of DB Parameters That Can Be Changed | 53 |
| (1) List of DB parameters that can be changed in Enterprise Postgres 9.6 | 54 |
| (2) List of parameters that can be changed in Symfoware Server V12.1..... | 64 |
| A.2 Estimating Memory Usage..... | 71 |
| (1) Base Memory Usage | 71 |
| (2) DB Engine Memory Usage | 71 |
| Appendix B: Supported Client..... | 72 |
| Appendix C: Flavor ID List | 73 |
| Appendix D: Transparent Data Encryption | 74 |
| D.1 Transparent Data Encryption Function | 74 |
| D.2 Encryption Range | 74 |
| D.3 Encrypting a Tablespace..... | 75 |
| D.4 Changing Master Encryption Key | 77 |
| D.5 Changing the Keystore Passphrase | 78 |
| D.6 Backing Up and Restoring/Recovering Database..... | 79 |
| D.7 Introducing Pre-Configured Application | 80 |
| D.8 Other Points to Note | 81 |

| | |
|---|----|
| Appendix E: Roles That Can Execute Each API | 82 |
| Appendix F: Upgrading the DB Engine Version of an Existing DB Instance | 84 |
| (1) Creating the DB Parameter Group for the New DB Engine..... | 85 |
| (2) Creating the DB Instance for the New DB Engine | 85 |
| (3) Stopping Business Operations | 85 |
| (4) Handling Incompatibility | 86 |
| (5) Extracting Data Using the pg_dumpall Command | 86 |
| (6) Inserting Data Using psql Command | 86 |
| (7) Confirming Data Migration..... | 87 |
| (8) Deleting DB Instances of the Old DB Engine Refer to "6.1 Deleting a DB Instance", and delete the DB instance of the old DB engine. | 87 |
| (9) Changing Application Connection Destinations (or FQDNs) | 87 |
| (10) Resuming Business Operations | 88 |
| Appendix G: postgres_fdw | 89 |
| G.1 postgres_fdw | 89 |
| G.2 How to Use..... | 89 |
| Appendix H: Creating a Read Replica | 91 |
| (1) Create the DB Instance | 91 |
| (2) Connect from the DB Client to the DB Instance | 91 |
| (3) Create a Read Replica | 91 |
| (4) Connect from the DB Client to the Read Replica | 93 |
| Appendix I: List of Restrictions | 93 |

This chapter describes the operations of the IaaS Database Service.

This manual is based on the system configuration shown below.

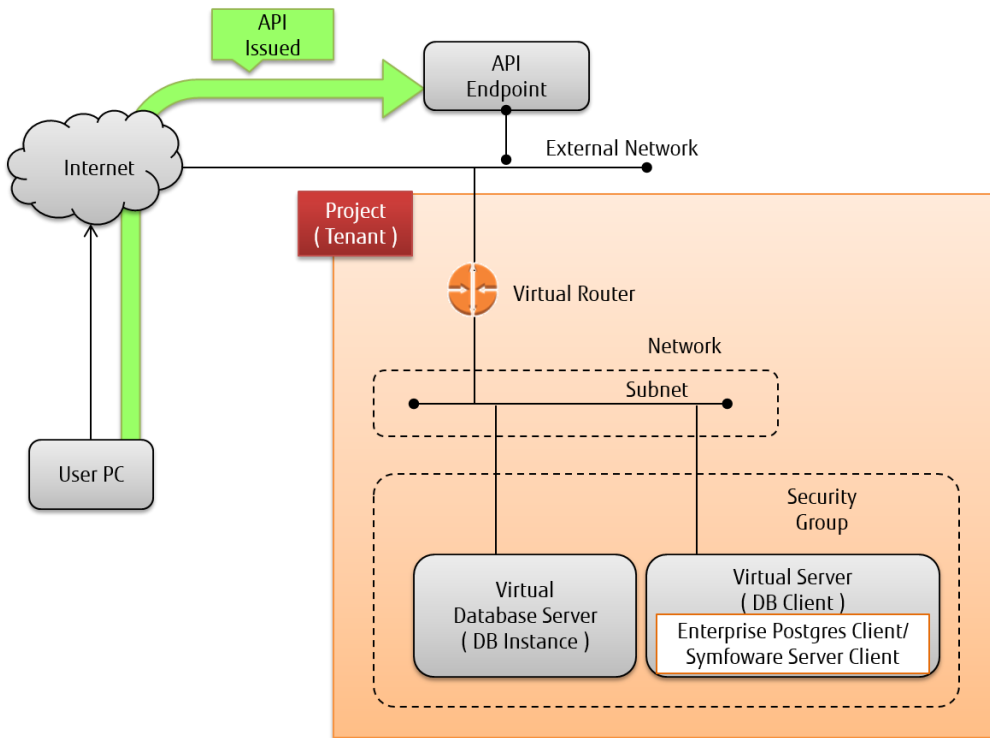


Figure 1.1 System Configuration

With the IaaS Database Service you can create virtual database servers. The virtual database server is a virtual server where a database is created. A virtual database server is referred to as a DB instance hereinafter.

In order to create a DB instance, you must create the following resources in advance.

| Resource | Description |
|--|--|
| Network/Subnet/IP Address | Networks are created in a project in order to create resources such as virtual servers. Subnets are required to manage private IP addresses of resources that are connected to a network, and to configure automatic settings for an IP address with DHCP. |
| Virtual Router | Virtual routers are created to connect external networks to internal networks. They are used during the FQDN name resolution required when connecting to a database virtual server, and for connections from external networks that are performed over the Internet. |
| Security Group | This group functions as a firewall for the associated DB instance. To establish a connection to the database on a DB instance, you must create a rule to permit connection to the DB instance. |
| Network Connectors/Connector Endpoints | In multi-AZ configurations, these are used to connect the networks of different AZs to one another. |
| DB Subnet Group | This group defines a subnet where a DB instance is to be created. In order to create a DB subnet group, you must create the above-mentioned subnet in advance. |
| DB Parameter Group | You can specify DB parameters applicable to DB instances. |

[Reference]

A DB instance belongs to one of the subnets included in the DB subnet group specified during the creation of the DB instance. It belongs to the availability zone specified during the creation of the DB instance.

[Reference]

You can change a subnet that belongs to a DB subnet group. However, a subnet where a DB instance is created cannot be changed.

[Reference]

For Multi-AZ, the system configuration is as follows.

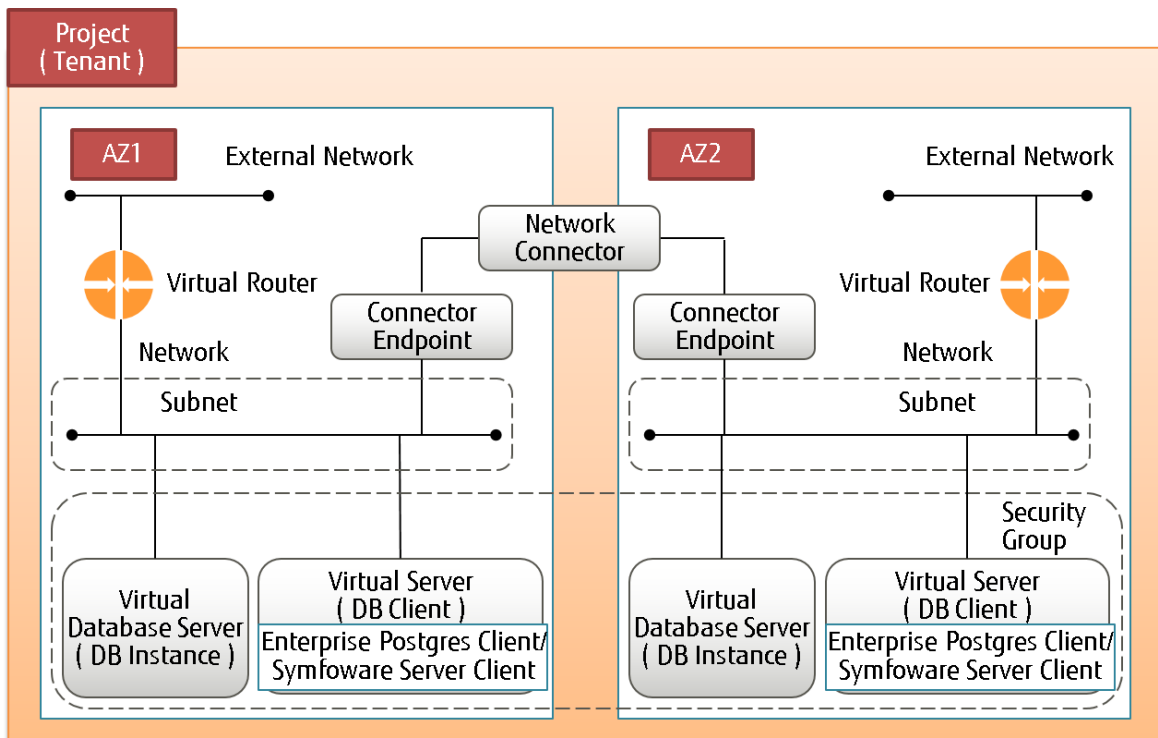


Figure 1.2 Multi-AZ System Configuration

The IaaS Database Service issues and uses the REST API.

The parameters described below are commonly specified when the REST API is issued.

- Request header

| Header | Description |
|--------------|--|
| X-Auth-Token | Specify the token to be obtained during user authentication. |

- Response type

| Header | Description |
|--------------|---|
| Content-Type | For POST/PUT, specify application/json. |

The REST API is issued for the endpoint URL of the database service.

- RESTAPI RequestURI

| URL | Description |
|--|--|
| https://database.[region-name].cloud.global.fujitsu.com/[API_version_information]/[tenant_id]/... | <p>[region-name] Specify the region name. Example: jp-east-1</p> <p>[API_version_information] Specify the API version in the format of "vx.x". Example: v1.0</p> <p>[tenant_id] Specify the project ID. Example: 1234</p> <p><Example of Endpoint URL> https://database.jp-east-1.cloud.global.fujitsu.com/v1.0/1234</p> |

The following chapters describe the procedures to create and connect DB instances, operate DB instances, change DB instances, deal with errors, and to delete DB instances, along with the REST API command examples.

This manual describes all command examples on the assumption that a project ID is set for the environment variable TENANTID.

[Reference]

* Obtaining a TOKEN

In order to issue the REST API to the database service, you must obtain a TOKEN in advance to be used for authentication.

This manual describes all command examples on the assumption that the TOKEN obtained in advance is set for the environment variable TOKEN.

The IaaS Database Service is operated by a user who executes the REST API and the database administrator.

With the FUJITSU cloud service, you can manage privileges by adding a role to the user with the user management service function.

There are four types of preset roles that can execute the REST API of the IaaS Database Service:

- Preset Roles That Can Execute REST API of IaaS Database Service

| Role | Executable API |
|--------------------|---|
| Administrator role | A user with this role can execute all APIs of the IaaS Database Service. |
| System Owner role | A user with this role can execute all APIs of the IaaS Database Service. |
| Operator role | <p>A user with this role can operate all APIs of the IaaS Database Service except for adding and deleting resources.</p> <p>However, APIs required for the recovery of a DB instance can be executed.</p> |
| Observer role | A user with this role can execute the APIs required for the monitoring of the IaaS Database Service. |

For details about the APIs executable by each role, refer to Appendix E.

For details about the roles and the procedures to add a role to users, refer to "11.3 User Management" in "FUJITSU Cloud Service for OSS IaaS Features Handbook".

This chapter describes the basic operations from creation of a DB instance through connection to the database.

The descriptions in this chapter are based on the assumption that the subnet where the DB instance is created and the virtual server (hereinafter referred to as the DB client) to be connected to the database are created in advance.

The operations described in this chapter are performed by users that are assigned with the Administrator role or the System Owner role.

2.1 Preparations

This section describes the preparations required for the creation of a DB instance.

(i) Creating a Network/Subnet

For details about creating a network/subnet, refer to "5.2 Creating a network" to "5.3 Creating a subnet" in "FUJITSU Cloud Service K5 IaaS API User Guide".

The IP address is automatically created when a subnet is created.

The subnet created here will be used when a DB subnet group is created in section 2.2 (1).

Create networks/subnets in all AZs where you want to create DB instances.

[Reference]

The number of IP addresses consumed when a DB instance is created is described below.

Number of Consumed privateIP Addresses for Each Subnet

privateIP addresses are consumed within the subnet network addresses.

The number of consumed privateIP addresses for each subnet is as follows.

| DB Instance Configuration | Permitting Connection via the Internet | Not Permitting Connection via the Internet |
|---------------------------|--|--|
| Single | 9 | 5 |
| Redundant in AZ | 10 | 6 |
| Redundant between AZ | 9 | 5 |

Total Number of Consumed publicIP Addresses

publicIP addresses are consumed only when connection via the Internet is permitted.

publicIP addresses are consumed from the external network.

The number of consumed publicIP addresses is as follows:

For a single configuration: 5

For a redundant configuration: 10

* Permission for Connection via the Internet

To create a DB instance to permit connection via the Internet from external networks, specify "true" for the publiclyAccessible parameter during the creation of the DB instance. For details about the publiclyAccessible parameter, refer to "FUJITSU Cloud Service for OSS IaaS API Reference (Application Platform Service)".

(2) Creating a Virtual Router

The subnets created in (1) must meet the following requirements:

For connections via the Internet from an external network: All subnets are attached to the external network.

For multi-AZ: There is a route from the default gateway of each subnet to the IP of all other subnets.

To make the subnet meet the above-mentioned requirements, create a virtual router.

For details about creating a virtual router, refer to "5.4 Creating a virtual router" to "5.6 Changing the configuration of a virtual router (attaching to a subnet)" in "FUJITSU Cloud Service K5 IaaS API User Guide".

Create virtual routers in all AZs where you want to create DB instances.

For multi-AZ, set a route on the virtual router in each AZ to reach the IP on the other subnet.

For details about routing settings, refer to "1.4.6.8 Update extra route" in "FUJITSU Cloud Service for OSS IaaS API Reference (Network)".

(3) Creating a Security Group

Create a security group to permit connection to the DB instance.

The security group created here is set to the DB instance during the creation of the DB instance.

| Command Example |
|---|
| <pre>SECURITY_GROUP_NAME=[<i>Security group name to be created</i>] ENDPOINT=[<i>Endpoint of the network service</i>] curl -X POST -i \${ENDPOINT}/v2.0/security-groups -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -H "Accept: application/json" -d "{\"security_group\": {\"name\": \"\${SECURITY_GROUP_NAME}\"}}</pre> |
| Example of Execution Results |
| <pre>\$ SECURITY_GROUP_NAME=test-security-group \$ ENDPOINT=https://networking.jp-east-1.cloud.global.fujitsu.com \$ curl -X POST -i \${ENDPOINT}/v2.0/security-groups -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -H "Accept: application/json" -d "{\"security_group\": {\"name\": \"\${SECURITY_GROUP_NAME}\"}}" HTTP/1.1 201 Created (Omitted) { "security_group": { "tenant_id": "XXXXXXXX", "description": "", "name": "test-security-group", "security_group_rules": [{ "remote_group_id": null, "direction": "egress", "remote_ip_prefix": null, "protocol": null, "ethertype": "IPv6", "port_range_max": null, "security_group_id": "XXXXXXXX", "port_range_min": null, "tenant_id": "XXXXXXXX", "id": "XXXXXXXX" }, { "remote_group_id": null, "direction": "egress", "remote_ip_prefix": null, "protocol": null, "ethertype": "IPv4", "port_range_max": null, "security_group_id": "XXXXXXXX", "port_range_min": null, "tenant_id": "XXXXXXXX", "id": "XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX" }] } } } *security_group id is set as SECGRP_ID and used in (4).</pre> |

For details about creating a security group, refer to "5.7 Creating a security group" in "FUJITSU Cloud Service K5 IaaS API User Guide".

(4) Creating Security Group Rules

Specify the following rules for the security group that you set to a DB instance.

| Purpose of Rule | Direction | Protocol | Permission Target* | Port |
|------------------------------------|-----------|----------|-------------------------------------|--|
| Communication from DB client | ingress | tcp | DB client | Port number specified during creation of DB instance (26500 by default) |
| DB instance internal communication | ingress | tcp | Subnet where DB instance is created | Port number specified during creation of DB instance (26500 by default) |
| DB instance internal communication | egress | tcp | Subnet where DB instance is created | Port number specified during creation of DB instance (26500 by default) |

For the connection permission target of the security group, you can select either of the following two types:

- --remote-ip-prefix
 - CIDR of the connection permission target
- --remote-group-id
 - Security group of the connection permission target

| Command Example |
|---|
| <p>CLIENT_CIDR=[<i>CIDR of the DB client (Specify the value created in advance.)</i>]</p> <p>SECGRP_ID=[<i>Security group ID created in (1)</i>]</p> <p>PORT=[<i>Port number specified during DB instance creation (Specify a desired value.)</i>]</p> <p>ENDPOINT=[<i>Endpoint of network service</i>]</p> <pre>curl -X POST -i \${ENDPOINT}/v2.0/security-group-rules -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -H "Accept: application/json" -d "{\"security_group_rule\": {\"direction\": \"ingress\", \"port_range_min\": \"\${PORT}\", \"ethertype\": \"IPv4\", \"port_range_max\": \"\${PORT}\", \"remote_ip_prefix\": \"\${CLIENT_CIDR}\", \"protocol\": \"tcp\", \"security_group_id\": \"\${SECGRP_ID}\"}}</pre> |

| Example of Execution Results |
|--|
| <pre> \$ CLIENT_CIDR=<i>xxx.xxx.xxx.xxx/xx</i> \$ SECGRP_ID=<i>xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx</i> \$ PORT=<i>26500</i> \$ ENDPOINT=<i>https://networking.jp-east-1.cloud.global.fujitsu.com</i> \$ curl -X POST -i \${ENDPOINT}/v2.0/security-group-rules -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -H "Accept: application/json" -d '{"security_group_rule": {"direction": "ingress", "port_range_min": "\${PORT}", "ethertype": "IPv4", "port_range_max": "\${PORT}", "remote_ip_prefix": "\${CLIENT_CIDR}", "protocol": "tcp", "security_group_id": "\${SECGRP_ID}"}}' HTTP/1.1 201 Created (Omitted) </pre> |

The commands shown below must be executed separately for every subnet to be included in the DB subnet group created in section 2.2 (1).

* Section 2.2 describes an example where two subnets are included in the DB subnet group. In this example, the command examples shown below must be executed for each subnet (a total of two times).

| Command Example |
|---|
| <pre> INSTANCE_CIDR=<i>[CIDR of the DB instance (Network ID of the subnet where the DB instance is to be created)]</i> SECGRP_ID=<i>[Security group ID created in (1)]</i> PORT=<i>[Port number specified during DB instance creation (Specify a desired value.)]</i> ENDPOINT=<i>[Endpoint of network service]</i> curl -X POST -i \${ENDPOINT}/v2.0/security-group-rules -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -H "Accept: application/json" -d '{"security_group_rule": {"direction": "ingress", "port_range_min": "\${PORT}", "ethertype": "IPv4", "port_range_max": "\${PORT}", "remote_ip_prefix": "\${INSTANCE_CIDR}", "protocol": "tcp", "security_group_id": "\${SECGRP_ID}"}}' curl -X POST -i \${ENDPOINT}/v2.0/security-group-rules -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -H "Accept: application/json" -d '{"security_group_rule": {"direction": "egress", "port_range_min": "\${PORT}", "ethertype": "IPv4", "port_range_max": "\${PORT}", "remote_ip_prefix": "\${INSTANCE_CIDR}", "protocol": "tcp", "security_group_id": "\${SECGRP_ID}"}}' </pre> |

| Example of Execution Results |
|---|
| <pre> \$ INSTANCE_CIDR=<i>XXX.XXX.XXX.XXX/XX</i> \$ SECGRP_ID=<i>XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX</i> \$ PORT=<i>26500</i> \$ ENDPOINT=<i>https://networking.jp-east-1.cloud.global.fujitsu.com</i> \$ curl -X POST -i \${ENDPOINT}/v2.0/security-group-rules -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -H "Accept: application/json" -d "{\"security_group_rule\": {\"direction\": \"ingress\", \"port_range_min\": \"\${PORT}\", \"ethertype\": \"IPv4\", \"port_range_max\": \"\${PORT}\", \"remote_ip_prefix\": \"\${INSTANCE_CIDR}\", \"protocol\": \"tcp\", \"security_group_id\": \"\${SECGRP_ID}\"}}" HTTP/1.1 201 Created (Omitted) \$ curl -X POST -i \${ENDPOINT}/v2.0/security-group-rules -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -H "Accept: application/json" -d "{\"security_group_rule\": {\"direction\": \"egress\", \"port_range_min\": \"\${PORT}\", \"ethertype\": \"IPv4\", \"port_range_max\": \"\${PORT}\", \"remote_ip_prefix\": \"\${INSTANCE_CIDR}\", \"protocol\": \"tcp\", \"security_group_id\": \"\${SECGRP_ID}\"}}" HTTP/1.1 201 Created (Omitted) </pre> |

2.2 Creating a DB Instance

This section describes the procedure to create a DB instance.

(1) Creating a DB Subnet Group

Create a DB subnet group that consists of subnets in AZ1 and AZ2.

The DB subnet group created here is set to the DB instance during the creation of the DB instance.

| Command Example |
|--|
| <pre> SUBNET1=<i>[ID of the AZ1 subnet where the DB instance is to be created (Specify the value created in advance.)]</i> SUBNET2=<i>[ID of the AZ2 subnet where the DB instance is to be created (Specify the value created in advance.)]</i> SUBNETGROUPID=<i>[DB subnet group ID (Specify a desired value.)]</i> SUBNETGROUPNAME=<i>[DB subnet group name (Specify a desired value.)]</i> ENDPOINT=<i>[Endpoint of the database service]</i> curl -X POST -i \${ENDPOINT}/v1.0/\${TENANTID}/subnetgroups -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d "{\"subnetgroup\": {\"name\": \"\${SUBNETGROUPNAME}\", \"id\": \"\${SUBNETGROUPID}\", \"subnetds\": [{\"subnetId\": \"\${SUBNET1}\", {\"subnetId\": \"\${SUBNET2}\"}]}}"}" </pre> |

| Example of Execution Results |
|--|
| <pre> \$ SUBNET1=<i>YYYYYY-YYY-YYY-YYY-YYYYYYYY</i> \$ SUBNET2=<i>ZZZZZZZ-ZZZZ-ZZZZ-ZZZZ-ZZZZZZZZZZZ</i> \$ SUBNETGROUPID=<i>test-subnetgroup-id</i> \$ SUBNETGROUPNAME=<i>test-subnetgroup-name</i> \$ ENDPOINT=<i>https://database.jp-east-1.cloud.global.fujitsu.com</i> \$ curl -X POST -i \${ENDPOINT}/v1.0/\${TENANTID}/subnetgroups -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d {"subnetgroup": {"name": "\${SUBNETGROUPNAME}", "id": "\${SUBNETGROUPID}", "subnetds": [{"subnetId": "\${SUBNET1}"}, {"subnetId": "\${SUBNET2}"}]}} HTTP/1.1 200 OK (Omitted) </pre> |

Make sure that the creation of the DB subnet group has been completed.

Execute the information reference API of the DB subnet group as shown below and make sure that "status" shows "Available".

| Command Example |
|--|
| <pre> SUBNETGROUPID=<i>[ID of the DB subnet group created above]</i> ENDPOINT=<i>[Endpoint of database service]</i> curl -X GET -i \${ENDPOINT}/v1.0/\${TENANTID}/subnetgroups/\${SUBNETGROUPID} -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" </pre> |
| Example of Execution Results |
| <pre> \$ SUBNETGROUPID=<i>test-subnetgroup-id</i> \$ ENDPOINT=<i>https://database.jp-east-1.cloud.global.fujitsu.com</i> \$ curl -X GET -i \${ENDPOINT}/v1.0/\${TENANTID}/subnetgroups/\${SUBNETGROUPID} -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" HTTP/1.1 200 OK (Omitted) {"subnetgroup":{"name":"test-subnetgroup-name","id":"test-subnetgroup-id","status":"Available","subnetds":[{"subnetId":"YYYYYYYY-YYYY-YYYY- YYYY-YYYYYYYYYYYY"}, {"subnetId":"ZZZZZZZ-ZZZZ-ZZZZ-ZZZZ-ZZZZZZZZZZZ"}],"created":"YYYY-MM-DDThh:mm:ssZ","description":null}} </pre> |

(2) Creating a DB Parameter Group

Create a DB parameter group to be applied to the DB instance.

The DB parameter group created here is applied to the database parameters.

For details about changing the parameters, refer to section 4.5.

When creating the DB parameter group, specify the DB parameter group based on the DB engine version of the DB instance to create.

| engine | engineVersion | parameterGroupFamily |
|--------------------|---------------|-------------------------|
| enterprisepostgres | 9.6 | enterprisepostgres_v9.6 |
| symfoware | 12.1 | symfoware_v12.1 |

[Reference]

.....

If the DB engine version is omitted when creating a DB instance, the default DB engine version is selected. As of November 2017, the default version is "enterprisepostgres 9.6". Therefore, select "enterprisepostgres 9.6".

.....

| Command Example |
|--|
| <pre>PARAMG_ID=[DB parameter group ID (Specify a desired value.)] PARAMG_NAME=[DB parameter group name (Specify a desired value.)] PARAMG_FAMILY=[DB parameter group family] ENDPOINT=[Endpoint of the database service] curl -X POST -i \${ENDPOINT}/v1.0/\${TENANTID}/parametergroups -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d {"parametergroup": {"parameterGroupFamily": "\${PARAMG_FAMILY}", "name": "\${PARAMG_NAME}", "id": "\${PARAMG_ID}"}}</pre> |
| Example of Execution Results |
| <pre>\$ PARAMG_ID=test-paramg-id \$ PARAMG_NAME=test-paramg-name \$ PARAMG_FAMILY="enterprisepostgres_v9.6" \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X POST -i \${ENDPOINT}/v1.0/\${TENANTID}/parametergroups -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d {"parametergroup": {"parameterGroupFamily": "\${PARAMG_FAMILY}", "name": "\${PARAMG_NAME}", "id": "\${PARAMG_ID}"}} HTTP/1.1 200 OK (Omitted) {"parametergroup":{"id":"test-paramg-id","description":null,"name":"test-paramg-name","parameterGroupFamily":"enterprisepostgres_v9.6"}}</pre> |

(3) Creating a DB Instance

Create a DB instance.

The options to be specified when API is executed are described below.

- AZ name (availabilityZone)

Among the public availability zones, select one of the AZ names as the DB instance creation destination.

- DB instance ID (id)/name (name)

Specify a desired value for the DB instance ID and name.

- Security group ID (securityGroupIds)/DB subnet group ID (subnetGroupId)/DB parameter group ID (parameterGroupId)

Specify the security group ID, DB subnet group ID, and DB parameter group ID that are created in advance.

- Flavor ID (flavorRef)

Specify an appropriate ID according to Appendix C.

- Database administrator name/administrator password

Specify a desired value for the database administrator name and password.

- Port number

Specify the database port number.

Specify the same value as the port number specified in the security group rule in section 2.1 (2).

- Data disk size (volume)

Specify the disk size of the DB instance in GB. Specify a desired value from 10 to 10240.

- Automatic backup retention period (backupRetentionPeriod)

For the automatic backup retention period, specify the number of generations for which to retain automatic backups. Specify a desired value from 0 to 10.

Specifying "1" or more for the automatic backup retention period enables automatic backup. Specifying "0" disables automatic backup.

If you specify "1" or more, the first backup operation will be performed after the DB instance is created. Until backup is completed, "status" remains "Build".

When backup is completed, "status" changes to "Active".

- Automatic backup time (preferredBackupWindow)

For the automatic backup time, specify a time period during which daily automatic backup is performed.

At the start time of the period specified above, automatic backup starts.

Depending on the data volume, backup creation may continue beyond the end time of the period specified above.

We recommend that the automatic backup time be set in a period with low workload, because the performance of issuing SQL statements lowers during backup creation.

- Automatic maintenance enabled/disabled (autoMaintenance)/Automatic maintenance time (preferredMaintenanceWindow)

With automatic maintenance, the application of DB instance patches and changes to DB instances that require a restart cause the DB instance to restart.

For details about changes to DB instances that require a restart, refer to section 4.4.

Specify "true" or "false" to enable/disable automatic maintenance.

If this parameter is not specified, "true" is specified by default. Specifying "false" disables automatic maintenance.

For the automatic maintenance time, specify a time period during which weekly automatic maintenance is performed.

At the start time of the period on the day of the week specified above, maintenance starts automatically.

SQL statements cannot be issued during automatic maintenance, so set the automatic maintenance time in a period with the lowest workload, and change the time when necessary. For details about the procedures to make such changes, refer to section 4.4.

- DB engine (engine) and version (engineVersion)

For the DB engine and version, the latest engine and version are selected by default.

As of November 2017, the following DB engine versions can be created.

| engine | engineVersion |
|------------------------------|---------------|
| enterprisepostgres (Default) | 9.6 (Default) |
| symfoware | 12.1 |

As support for a DB engine version (engineVersion) ends five years after it is initially published, use of the latest DB engines and versions is recommended.

You can select either single-configuration or multi-configuration for the DB instance configuration by setting other options. The following example shows a case where a single-configuration DB instance is created.

| Command Example |
|---|
| <p>AZ=[AZ name to be specified as the creation destination of the DB instance (Select any of the public availability zones.)]</p> <p>INSTANCEID=[DB instance ID (Specify a desired value.)]</p> <p>INSTANCENAME=[DB instance name (Specify a desired value.)]</p> <p>SUBNETGROUPID=[ID of the DB subnet group created in (1)]</p> <p>FLAVOR=[Flavor ID (Select any of the public flavor IDs.)]</p> <p>MASTERUSERNAME=[Database administrator name (Specify a desired value.)]</p> <p>MASTERUSERPASSWORD=[Database administrator password (Specify a desired value.)]</p> <p>PORT=[Port number specified during DB instance creation (Specify the same value as the PORT value specified in section 2.1 (2).)]</p> <p>SECGRP_ID=[Security group ID created in section 2.1 (1)]</p> <p>PARAMG_ID=[ID of the DB parameter group created in (2)]</p> <p>SIZE=[Data disk size [GB] (Specify a desired value from 10 to 10240.)]</p> <p>BACKUP_RETENTION_PERIOD=[Backup retention period [day] (Specify a desired value from 0 to 10.)]</p> <p>PREFERRED_BACKUP_WINDOW=[Backup time [UTC] (Specify a desired period of time for 30 min. or more.)]</p> <p>AUTO_MAINTENANCE=[Automatic maintenance enabled/disabled (Specify true or false.)]</p> <p>PREFERRED_MAINTENANCE_WINDOW=[Maintenance time [UTC] (Specify a desired day of the week and period of time for 30 min. or more.)]</p> <p>ENDPOINT=[Endpoint of database service]</p> <pre>curl -X POST -i \${ENDPOINT}/v1.0/\${TENANTID}/instances -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d '{"instance": {"multiAZ": "false", "multi": "false", "availabilityZone": "\${AZ}", "name": "\${INSTANCENAME}", "subnetGroupId": "\${SUBNETGROUPID}", "masterUserName": "\${MASTERUSERNAME}", "publiclyAccessible": "false", "flavorRef": "\${FLAVOR}", "masterUserPassword": "\${MASTERUSERPASSWORD}", "port": \${PORT}, "volume": {"type": "MI", "size": \${SIZE}}, "preferredMaintenanceWindow": "\${PREFERRED_MAINTENANCE_WINDOW}", "parameterGroupId": "\${PARAMG_ID}", "securityGroupIds": [{"securityGroupId": "\${SECGRP_ID}"], "backupRetentionPeriod": \${BACKUP_RETENTION_PERIOD}, "id": "\${INSTANCEID}", "preferredBackupWindow": "\${PREFERRED_BACKUP_WINDOW}", "autoMaintenance": "\${AUTO_MAINTENANCE}"}}</pre> |

```

Example of Execution Results

$ AZ=jp-east-1a
$ INSTANCEID=test-instance-id
$ INSTANCENAME=test-instance-name
$ SUBNETGROUPID=test-subnetgroup-id
$ FLAVOR=i101
$ MASTERUSERNAME="masterusername"
$ MASTERUSERPASSWORD="masteruserpassword"
$ PORT=26500
$ SECGRP_ID=xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
$ PARAMG_ID=test-paramg-id
$ SIZE=10
$ BACKUP_RETENTION_PERIOD=5
$ PREFERRED_BACKUP_WINDOW="16:00-16:30"
$ AUTO_MAINTENANCE=true
$ PREFERRED_MAINTENANCE_WINDOW="Sat:18:00-Sat:18:30"
$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com

$ curl -X POST -i ${ENDPOINT}/v1.0/${TENANTID}/instances -H "X-Auth-Token: ${TOKEN}" -H "Content-Type: application/json" -d '{"instance":
{"multiAZ": "false", "multi": "false", "availabilityZone": "${AZ}", "name": "${INSTANCENAME}", "subnetGroupId":
"${SUBNETGROUPID}", "masterUserName": "${MASTERUSERNAME}", "publiclyAccessible": "false", "flavorRef": "${FLAVOR}",
"masterUserPassword": "${MASTERUSERPASSWORD}", "port": ${PORT}, "volume": {"type": "M1", "size": ${SIZE}},
"preferredMaintenanceWindow": "${PREFERRED_MAINTENANCE_WINDOW}", "parameterGroupId": "${PARAMG_ID}", "securityGroupIds":
[{"securityGroupId": "${SECGRP_ID}"], "backupRetentionPeriod": ${BACKUP_RETENTION_PERIOD}, "id": "${INSTANCEID}",
"preferredBackupWindow": "${PREFERRED_BACKUP_WINDOW}", "autoMaintenance": "${AUTO_MAINTENANCE}}"'

HTTP/1.1 202 Accepted
(Omitted)

```

After the above-mentioned command is executed, the creation of the DB instance is completed. Then "status" changes to "Active". For details about the DB instance status, refer to "FUJITSU Cloud Service for OSS IaaS API Reference (Application Platform Service)".

* Example of Creating a Multi-Configuration DB Instance

To create a multi-configuration DB instance, specify "true" for both the multi parameter and the multiAZ parameter of the DB instance creation API. The relationship between each parameter and the DB instance configuration is as follows.

| multi Parameter | multiAZ Parameter | DB Instance Configuration |
|-----------------|-------------------|--|
| false | false | A single-configuration DB instance is created. |
| true | false | Two DB instances are created in the same AZ and made redundant. |
| false | true | Unavailable. |
| true | true | With one unit for each of the two AZs included in the specified DB subnet group, a total of two DB instances are created and made redundant. |

The example below shows the commands that create a multi-configuration DB instance with multi=true and multiAZ=true.

* For the flavor ID, specify an appropriate value according to Appendix C.

| Command Example |
|---|
| <p>AZ=[AZ name to be specified as the creation destination of the DB instance (Select any of the public availability zones.)]</p> <p>INSTANCEID=[DB instance ID (Specify a desired value.)]</p> <p>INSTANCENAME=[DB instance name (Specify a desired value.)]</p> <p>SUBNETGROUPID=[ID of DB subnet group created in (1)]</p> <p>FLAVOR=[Flavor ID (Select any of the public flavor IDs.)]</p> <p>MASTERUSERNAME=[Database administrator name (Specify a desired value.)]</p> <p>MASTERUSERPASSWORD=[Database administrator password (Specify a desired value.)]</p> <p>PORT=[Port number specified during DB instance creation (Specify the same value as the PORT value specified in section 2.1 (2).)]</p> <p>SECGRP_ID=[Security group ID created in section 2.1 (1)]</p> <p>PARAMG_ID=[ID of the DB parameter group created in (2)]</p> <p>SIZE=[Data disk size [GB] (Specify a desired value from 10 to 10240.)]</p> <p>BACKUP_RETENTION_PERIOD=[Backup retention period [day] (Specify a desired value from 0 to 10.)]</p> <p>PREFERRED_BACKUP_WINDOW=[Backup time [UTC] (Specify a desired period of time for 30 min. or more.)]</p> <p>AUTO_MAINTENANCE=[Automatic maintenance enabled/disabled (Specify true or false.)]</p> <p>PREFERRED_MAINTENANCE_WINDOW=[Maintenance time [UTC] (Specify a desired day of the week and period of time for 30 min. or more.)]</p> <p>ENDPOINT=[Endpoint of database service]</p> <pre> curl -X POST -i \${ENDPOINT}/v1.0/\${TENANTID}/instances -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d '{"instance": {"multiAZ": "true", "multi": "true", "availabilityZone": "\${AZ}", "name": "\${INSTANCENAME}", "subnetGroupId": "\${SUBNETGROUPID}", "masterUserName": "\${MASTERUSERNAME}", "publiclyAccessible": "false", "flavorRef": "\${FLAVOR}", "masterUserPassword": "\${MASTERUSERPASSWORD}", "port": \${PORT}, "volume": {"type": "M1", "size": \${SIZE}}, "preferredMaintenanceWindow": "\${PREFERRED_MAINTENANCE_WINDOW}", "parameterGroupId": "\${PARAMG_ID}", "securityGroupIds": [{"securityGroupId": "\${SECGRP_ID}"], "backupRetentionPeriod": \${BACKUP_RETENTION_PERIOD}, "id": "\${INSTANCEID}", "preferredBackupWindow": "\${PREFERRED_BACKUP_WINDOW}", "autoMaintenance": "\${AUTO_MAINTENANCE}"}' </pre> |

2.3 Connecting to a DB Instance

This section describes the procedure to connect to a DB instance.

(1) Checking DB Instance Information

Check the status of the DB instance to connect to. In this case, check the following two points:

- Confirm that the DB instance status is Active.
- Confirm the FQDN required for connection to the database.

| Command Example |
|---|
| <pre> INSTANCEID=[ID of the DB instance created in section 2.2 (3)] ENDPOINT=[Endpoint of the database service] curl -X GET -i \${ENDPOINT}/v1.0/\${TENANTID}/instances/\${INSTANCEID} -H "X-Auth-Token: \${TOKEN}" </pre> |
| Example of Execution Results |
| <pre> \$ INSTANCEID= test-instance-id \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X GET -i \${ENDPOINT}/v1.0/\${TENANTID}/instances/\${INSTANCEID} -H "X-Auth-Token: \${TOKEN}" HTTP/1.1 200 OK X-Fcx-Endpoint-Request: EXECUTED_REQ016533324_200 Date: Tue, 12 Jul 2016 02:31:33 GMT Server: Apache/2.2.15 (CentOS) Pragma: no-cache Expires: Thu, 01 Jan 1970 00:00:00 GMT Cache-Control: no-cache, no-store Content-Type: application/json;charset=UTF-8 Content-Length: 1601 {"instance":{"volume":{"size":10,"type":"M1"},"id":"test-instance-id","name":"test-instance- name","description":null,"multiAZ":false,"multi":false,"port":26500,"preferredBackupWindow":"17:50- 18:20","preferredMaintenanceWindow":"mon:01:46- mon:02:16","preferredRecoveryTime":{"applyImmediately":true,"time":null},"securityGroupIds":[{"securityGroupId":"XXXXXXXX-XXXX-XXXX-XXXX- XXXXXXXXXXXX"}],"parameterGroupId":"DefaultGroup- 12.1","backupRetentionPeriod":0,"autoMinorVersionUpgrade":true,"engineVersion":"9.6","engineMinorVersion":0,"autoMaintenance":true,"availabilit yZone":"XXXXXXXX","subnetGroupId":"test-subnetgroup- id","publiclyAccessible":false,"engine":"enterprisepostgres","masterUserName":"masterusername","characterSet":"UTF8","collate":"C","created":"2016- 07- 12T02:09:01Z","flavor":{"id":"1101","links":[{"href":"http://XXXXXXXX/v1.0/XXXXXXXX/flavors/1101","rel":"SELF"}, {"href":"http://XXXXXXXX/XXXXXXXX/flavo rs/1101","rel":"BOOKMARK"}]},"links":[{"href":"http://XXXXXXXX/v1.0/XXXXXXXX/instances/test-instance- id","rel":"SELF"}, {"href":"http://XXXXXXXX/XXXXXXXX/instances/test-instance- id","rel":"BOOKMARK"}],"status":"Active","updated":null,"privateAddress":"XXXXXXXX.XXX.XXX","privatelp":"XX.XX.XX.XX","publicAddress":null,"publi clp":null,"subPrivatelp":null,"subPubliclp":null,"pendingModifiedValues":null,"readReplicaDBInstanceIdentifiers":[],"readReplicaSrcDBInstanceIdentifi er":null,"secondaryAvailabilityZone":null,"replicaStatus":null}} </pre> |

To connect to the database, use the FQDN provided for each DB instance.

The FQDN is described in privateAddress in the API response shown above. Make a note of this FQDN to be used in (2).

* The FQDN confirmed in this section and the port number and database administrator name/password specified during creation of the DB instance in section 2.2 (3) will be used to connect to the database in (2). The user who created the DB instance must hand over the information to the database administrator.

(2) Connecting to a DB Instance

The operations described in this section are performed by the database administrator.

This section describes an example of the procedure to connect to the database on a DB instance.

Connect to the database from a DB client where the DB engine client (Enterprise Postgres Client or Symfoware Server Client) is installed.

For details about obtaining DB engine clients, refer to Appendix B.

For details about DB engine client functions, refer to the DB engine (Enterprise Postgres or Symfoware Server) manuals.

* When you use pgAdmin, refer to the DB engine manuals.

(For the server information specified for pgAdmin, use the parameters confirmed in (1).

For "Host", specify the FQDN confirmed in (1).)

| Command Example |
|---|
| <pre>CLIENT_DIR=<i>[Directory path where the DB engine client is installed]</i> export PATH=\${CLIENT_DIR}/bin:\${PATH} export LD_LIBRARY_PATH=\${CLIENT_DIR}/lib:\${LD_LIBRARY_PATH} FQDN=<i>[Address in the FQDN confirmed in (1)]</i> PORT=<i>[Port confirmed in (1)]</i> MASTERUSERNAME=<i>[Database administrator name confirmed in (1)]</i> SQL_STR=<i>[SQL statement to be issued]</i> psql -p \${PORT} -U \${MASTERUSERNAME} -h \${FQDN} -d postgres -c "\${SQL_STR}"</pre> |
| Example of Execution Results |
| <pre>\$ CLIENT_DIR="<i>XXXXXXXXXXXXXXXXXX</i>" \$ export PATH=\${CLIENT_DIR}/bin:\${PATH} \$ export LD_LIBRARY_PATH=\${CLIENT_DIR}/lib:\${LD_LIBRARY_PATH} \$ FQDN="<i>XXXXXXXX.XXX.XXX</i>" \$ PORT="<i>26500</i>" \$ MASTERUSERNAME="<i>masterusername</i>" \$ SQL_STR="<i>CREATE DATABASE testdb;</i>" \$ psql -p \${PORT} -U \${MASTERUSERNAME} -h \${FQDN} -d postgres -c "\${SQL_STR}" Password for user masterusername: CREATE DATABASE</pre> |

To encrypt the data to be stored in the database, use the transparent data encryption function.

To use the transparent data encryption function, no initial setting procedures are required during DB instance creation.

For details about the transparent data encryption function, refer to Appendix D.

2.4 Registering an Event Notification

This section describes the procedure to register event notifications for a DB instance.

Registering event notifications allows you to check the events that occurred in the DB instance.

By checking events, you can detect errors in the DB instance.

(1) Creating an Event Notification Registration

Register an event notification for the DB instance created in section 2.2.

| Command Example |
|---|
| <pre>EVENTSUB_ID=[<i>Event notification registration ID (Specify a desired value.)</i>] SOURCEID=[<i>ID of the DB instance created in section 2.2 (3)</i>] EVENTSUB_NAME=[<i>Event notification registration name (Specify a desired value.)</i>] SOURCE_TYPE=[<i>Resource type for event notification (Specify db-instance or db-snapshot.)</i>] ENDPOINT=[<i>Endpoint of the database service</i>] curl -X POST -i \${ENDPOINT}/v1.0/\${TENANTID}/eventnotifications -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d {"eventnotification": {"id": "\${EVENTSUB_ID}", "sourceIds": [{"sourceId": "\${SOURCEID}"}], "enabled": "false", "name": "\${EVENTSUB_NAME}", "sourceType": "\${SOURCE_TYPE}"}}</pre> |
| Example of Execution Results |
| <pre>\$ EVENTSUB_ID=test-eventsub-id \$ SOURCEID=test-instance-id \$ EVENTSUB_NAME=test-eventsub-name \$ SOURCE_TYPE=db-instance \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X POST -i \${ENDPOINT}/v1.0/\${TENANTID}/eventnotifications -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d {"eventnotification": {"id": "\${EVENTSUB_ID}", "sourceIds": [{"sourceId": "\${SOURCEID}"}], "enabled": "false", "name": "\${EVENTSUB_NAME}", "sourceType": "\${SOURCE_TYPE}"}} HTTP/1.1 200 OK (Omitted)</pre> |

(2) Checking the Event Notification Registration Information

Confirm that the event notification has been registered.

Confirm that "sourceId" in the API response indicates the ID of the DB instance to which the event notification is registered.

| Command Example |
|---|
| <pre>EVENTSUB_ID=[Event notification registration ID created in (1)] ENDPOINT=[Endpoint of the database service] curl -X GET -i \${ENDPOINT}/v1.0/\${TENANTID}/eventnotifications/\${EVENTSUB_ID} -H "X-Auth-Token: \${TOKEN}"</pre> |
| Example of Execution Results |
| <pre>\$ EVENTSUB_ID=test-eventsub-id \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X GET -i \${ENDPOINT}/v1.0/\${TENANTID}/eventnotifications/\${EVENTSUB_ID} -H "X-Auth-Token: \${TOKEN}" HTTP/1.1 200 OK (Omitted) { "eventSubscription": {"enabled": false, "sourceType": "db-instance", "id": "test-eventsub-id", "name": "test-eventsub- name", "eventCategories": null, "sourceIds": [{"sourceId": "test-instance-id"}], "description": null, "status": "Active", "created": "YYYY-MM-DDThh:mm:ssZ"} }</pre> |

The operations described in this chapter are performed by users that are assigned with the Administrator role, the System Owner role, or the Operator role.

The log check operations in section 3.3 can be performed also by users that are assigned with the Observer role.

3.1 Stopping/Starting a DB Instance

This section describes the procedure to stop/start a DB instance.

Perform this operation according to the operation policy. (Example: To prevent additional cost for the virtual server charges while the DB instance is not used.)

(1) Stopping a DB Instance

Stop a DB instance.

While a DB instance is stopped, SQL connection to the database is unavailable.

When a DB instance is stopped, SQL connection is disconnected. Before stopping a DB instance, be sure to stop the application.

For a DB instance that has been stopped, no charge for the virtual server is required.

| Command Example |
|--|
| <pre>INSTANCEID=[<i>ID of the DB instance created in advance (For details about creating a DB instance, refer to section 2.2 (3).)</i>] ENDPOINT=[<i>Endpoint of the database service</i>] curl -X POST -i \${ENDPOINT}/v1.0/\${TENANTID}/instances/\${INSTANCEID}/action -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d '{"action": {"stop": ""}}'</pre> |
| Example of Execution Results |
| <pre>\$ INSTANCEID=test-instance-id \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X POST -i \${ENDPOINT}/v1.0/\${TENANTID}/instances/\${INSTANCEID}/action -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d '{"action": {"stop": ""}}' HTTP/1.1 202 Accepted (Omitted)</pre> |

After the above-mentioned command is executed, the shutdown process is completed. Then "status" changes to "Shutdown". You can check the DB instance status by using the information reference API of the DB instance. (Refer to "FUJITSU Cloud Service for OSS IaaS API Reference (Application Platform Service)" or to section 2.3 (1).)

(2) Starting a DB Instance

Start a DB instance that is in shutdown status.

| Command Example |
|---|
| <pre>INSTANCEID=[ID of the DB instance created in advance (For details about creating a DB instance, refer to section 2.2 (3).)] ENDPOINT=[Endpoint of the database service] curl -X POST -i \${ENDPOINT}/v1.0/\${TENANTID}/instances/\${INSTANCEID}/action -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d '{"action": {"start": ""}}'</pre> |
| Example of Execution Results |
| <pre>\$ INSTANCEID=test-instance-id \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X POST -i \${ENDPOINT}/v1.0/\${TENANTID}/instances/\${INSTANCEID}/action -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d '{"action": {"start": ""}} HTTP/1.1 202 Accepted (Omitted)</pre> |

You can check the DB instance status by using the information reference API of the DB instance. (Refer to "FUJITSU Cloud Service for OSS IaaS API Reference (Application Platform Service)" or to section 2.3 (1).)

3.2 Creating a Backup Manually

This section describes the procedure to manually create a backup of a DB instance.

(1) Creating a DB snapshot

Create a backup of a DB instance manually.

| Command Example |
|--|
| <pre>INSTANCEID=[ID of the DB instance created in advance (For details about creating a DB instance, refer to section 2.2 (3).)] DESCRIPTION=[DB snapshot description (Specify as desired.)] SNAPSHOT_ID=[DB snapshot ID (Specify a desired value.)] SNAPSHOT_NAME=[DB snapshot name (Specify a desired value.)] ENDPOINT=[Endpoint of the database service] curl -X POST -i \${ENDPOINT}/v1.0/\${TENANTID}/snapshots -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d '{"snapshot": {"instanceId": "\${INSTANCEID}", "description": "\${DESCRIPTION}", "name": "\${SNAPSHOT_NAME}", "id": "\${SNAPSHOT_ID}"}}'</pre> |

| Example of Execution Results |
|--|
| <pre> \$ INSTANCEID=test-instance-id \$ DESCRIPTION="this is sample" \$ SNAPSHOT_ID=test-snapshot-id \$ SNAPSHOT_NAME=test-snapshot-name \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X POST -i \${ENDPOINT}/v1.0/\${TENANTID}/snapshots -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d {"snapshot": {"instanceId": "\${INSTANCEID}", "description": "\${DESCRIPTION}", "name": "\${SNAPSHOT_NAME}", "id": "\${SNAPSHOT_ID}"}} HTTP/1.1 202 Accepted (Omitted) {"snapshot":{"instanceId":"test-instance-id","id":"test-snapshot-id","name":"test-snapshot- name","snapshotType":"manual","status":"In_Progress","created":null,"description":"this is sample"}} </pre> |

(2) Referring to a DB Snapshot List

Make sure that the creation of the DB snapshot has been completed.

Execute the list reference API of the DB snapshot as shown below and make sure that "status" shows "Available".

The time required to complete the creation of a DB snapshot varies depending on the data volume.

| Command Example |
|--|
| <pre> ENDPOINT=[Endpoint of the database service] curl -X GET -i \${ENDPOINT}/v1.0/\${TENANTID}/snapshots?snapshotType=manual -H "X-Auth-Token: \${TOKEN}" </pre> |
| Example of Execution Results |
| <pre> \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X GET -i \${ENDPOINT}/v1.0/\${TENANTID}/snapshots?snapshotType=manual -H "X-Auth-Token: \${TOKEN}" HTTP/1.1 200 OK (Omitted) {"snapshots":[{"instanceId":"test-instance-id","id":"XXXXXXXX","name":"XXXXXXXX","snapshotType":"manual","status":"Available"}]} </pre> |

[Reference]

We recommend that you delete DB snapshots that are no longer necessary, in order to prevent additional cost for the snapshot/automatic backup charges. A command example of the deletion API for the DB snapshot is shown below.

| Command Example |
|---|
| <pre>SNAPSHOT_ID=[ID of the unnecessary DB snapshot] ENDPOINT=[Endpoint of the database service] curl -X DELETE -i \${ENDPOINT}/v1.0/\${TENANTID}/snapshots/\${SNAPSHOT_ID} -H "X-Auth-Token: \${TOKEN}"</pre> |
| Example of Execution Results |
| <pre>\$ SNAPSHOT_ID=test-snapshot-id \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X DELETE -i \${ENDPOINT}/v1.0/\${TENANTID}/snapshots/\${SNAPSHOT_ID} -H "X-Auth-Token: \${TOKEN}" HTTP/1.1 202 Accepted</pre> |

You can check whether the DB snapshot has been deleted by using the list reference API of the DB snapshot. In the list reference API of the DB snapshot, confirm that the deleted DB snapshot is not shown in the list.

3.3 Checking Database Logs

You can use the database logs when a database error occurs or to check the operation of the application to be created.

(i) Referring to a DB Log File List

Obtain the database log file list.

| Command Example |
|---|
| <pre>INSTANCEID=[ID of the DB instance created in advance (For details about creating a DB instance, refer to section 2.2 (3).)] LIMIT=[The number of log files to be displayed (Specify the desired value from 20 to 100.)] ENDPOINT=[Endpoint of the database service] curl -X GET -i \${ENDPOINT}/v1.0/\${TENANTID}/logfiles/\${INSTANCEID}?limit=\${LIMIT} -H "X-Auth-Token: \${TOKEN}"</pre> |

| Example of Execution Results |
|---|
| <pre> \$ INSTANCEID=test-instance-id \$ LIMIT=100 \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X GET -i \${ENDPOINT}/v1.0/\${TENANTID}/logfiles/\${INSTANCEID}?limit=\${LIMIT} -H "X-Auth-Token: \${TOKEN}" HTTP/1.1 200 OK (Omitted) {"describeDBLogFiles":[{"lastWritten":XXXXXXXX,"logFileName":"postgresql.log-YYYY-MM-DD_XXXXXX","size":XXXX},{"lastWritten":XXXXXXXX,"logFileName":"postgresql.log-YYYY-MM-DD_XXXXXX","size":XXXX},{"lastWritten":XXXXXXXX,"logFileName":"postgresql.log-YYYY-MM-DD_XXXXXX","size":XXXX}]} </pre> |

[Reference]

To refer to a DB log file for a specific date, specify the filelastwritten parameter to view a list of DB log files written on the specified date and later.

| Command Example |
|--|
| <pre> INSTANCEID=[ID of the DB instance created in advance (For details about creating a DB instance, refer to section 2.2 (3).)] LIMIT=[The number of log files to be displayed (Specify the desired value from 20 to 100.)] FILELASTWRITTEN=[Date/time when the log file was written last (POSIX time stamp)] ENDPOINT=[Endpoint of the database service] curl -X GET -i \${ENDPOINT}/v1.0/\${TENANTID}/logfiles/\${INSTANCEID}?limit=\${LIMIT}&filelastwritten=\${FILELASTWRITTEN} -H "X-Auth-Token: \${TOKEN}" </pre> |
| Example of Execution Results |
| <pre> \$ INSTANCEID=test-instance-id \$ LIMIT=100 \$ FILELASTWRITTEN=1485907200 \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X GET -i \${ENDPOINT}/v1.0/\${TENANTID}/logfiles/\${INSTANCEID}?limit=\${LIMIT}&filelastwritten=\${FILELASTWRITTEN} -H "X-Auth-Token: \${TOKEN}" HTTP/1.1 200 OK (Omitted) {"describeDBLogFiles":[{"lastWritten":XXXXXXXX,"logFileName":"postgresql.log-YYYY-MM-DD_XXXXXX","size":XXXX},{"lastWritten":XXXXXXXX,"logFileName":"postgresql.log-YYYY-MM-DD_XXXXXX","size":XXXX}]} </pre> |

(2) Referring to a DB Log File

From the DB log file list obtained in (1), refer to the DB log file to be checked.

| Command Example |
|---|
| <pre>INSTANCEID=[ID of the DB instance created in advance (For details about creating a DB instance, refer to section 2.2 (3).)] LOGFILENAME=[DB log file name checked in (1)] LIMIT=[The number of lines to be displayed (Specify the desired value from 20 to 100.)] ENDPOINT=[Endpoint of the database service] curl -X GET -i \${ENDPOINT}/v1.0/\${TENANTID}/logfiles/\${INSTANCEID}/\${LOGFILENAME}?limit=\${LIMIT} -H "X-Auth-Token: \${TOKEN}"</pre> |
| Example of Execution Results |
| <pre>\$ INSTANCEID=test-instance-id \$ LOGFILENAME=postgresql.log-YYYY-MM-DD_XXXXXX \$ LIMIT=100 \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X GET -i \${ENDPOINT}/v1.0/\${TENANTID}/logfiles/\${INSTANCEID}/\${LOGFILENAME}?limit=\${LIMIT} -H "X-Auth-Token: \${TOKEN}" HTTP/1.1 200 OK (Omitted) {"downloadDBLogFilePortion":{"additionalDataPending":false,"logFileData":{"LOG: XXXXXXXX\nLOG: XXXXXXXX\nLOG: XXXXXXXX\nLOG: XXXXXXXX\n","marker":"-1"}}</pre> |

[Reference]

Using "marker" for pagination allows you to specify the position from which the log file is to be obtained.

For "marker", specify the value returned by the reference API of the DB log file the previous time.

If "marker" is specified, the DB log file is displayed from the next line of "marker".

(If "marker" is not specified, the DB log file is displayed from the top.)

| Command Example |
|---|
| <pre>INSTANCEID=[ID of the DB instance created in advance (For details about creating a DB instance, refer to section 2.2 (3).)] LOGFILENAME=[DB log file name checked in (1)] LIMIT=[The number of lines to be displayed (Specify the desired value from 20 to 100.)] MARKER=[Character string to specify the position from which the list is to be obtained (Log is displayed from the next line of the specified marker.)] ENDPOINT=[Endpoint of the database service] curl -X GET -i \${ENDPOINT}/v1.0/\${TENANTID}/logfiles/\${INSTANCEID}/\${LOGFILENAME}?marker=\${MARKER}&limit=\${LIMIT} -H "X-Auth- Token: \${TOKEN}"</pre> |

| Example of Execution Results |
|---|
| <pre> \$ INSTANCEID=test-instance-id \$ LOGFILENAME=postgresql.log-YYYY-MM-DD_XXXXXX \$ LIMIT=20 \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X GET -i \${ENDPOINT}/v1.0/\${TENANTID}/logfiles/\${INSTANCEID}/\${LOGFILENAME}?limit=\${LIMIT} -H "X-Auth-Token: \${TOKEN}" HTTP/1.1 200 OK (Omitted) {"downloadDBLogFilePortion":{"additionalDataPending":false,"logFileData":"LOG: XXXXXXXX\nLOG: XXXXXXXX\nLOG: XXXXXXXX\nLOG: XXXXXXXX\n\n","marker":"20"}} *Specify this marker for "MARKER" to refer to the DB log file. \$ MARKER=20 \$ curl -X GET -i \${ENDPOINT}/v1.0/\${TENANTID}/logfiles/\${INSTANCEID}/\${LOGFILENAME}?marker=\${MARKER}&limit=\${LIMIT} -H "X-Auth-Token: \${TOKEN}" HTTP/1.1 200 OK (Omitted) {"downloadDBLogFilePortion":{"additionalDataPending":false,"logFileData":"LOG: XXXXXXXX\nLOG: XXXXXXXX\nLOG: XXXXXXXX\nLOG: XXXXXXXX\n\n","marker":"null"}} *The DB log file is displayed from the next line of the previous result. </pre> |

3.4 Performance Monitoring

Using the monitoring service, you can obtain the performance information of DB instances.

Regarding the disk capacity of the database virtual server, you need to check it periodically. We recommend you to keep 20% (available disk size) of the disk capacity.

If the disk capacity is exhausted, it will become necessary to perform recovery or re-create databases.

When the available disk capacity becomes low, consider expanding the disk size following "Chapter 4.2 Changing Data Disk Size". In addition, when the disk usage ratio exceeds 80%, an email notification with the event category "low storage" will be sent by the event notification function. If you receive such a notification, check the available disk capacity and consider expanding the disk size.

For details about obtaining performance information, refer to "Chapter 4: Monitoring Service" in "FUJITSU Cloud Service for OSS IaaS API Reference (Management Administration)".

For details about the monitored items of the database service to be obtained, refer to "A.7 Lists of Monitored Items" in "FUJITSU Cloud Service for OSS IaaS Features Handbook".

3.5 Checking an Event Notification

Event notifications are sent to the FUJITSU Cloud Service for OSS contractors and the administrators by email.

Take appropriate recovery measures for the DB instance as necessary. For details about DB instance recovery, refer to Chapter 5.

3.6 Changing the Encryption Key Used for Transparent Data Encryption

Using the transparent data encryption function (refer to Appendix D) allows you to encrypt the data to be stored in the database. We recommend that the master encryption key and the keystore passphrase be changed periodically while the transparent data encryption function is used.

The operations described in this chapter are performed by users that are assigned with the Administrator role or the System Owner role.

4.1 Changing a Flavor ID

Change the flavor as needed for operation.

* For the flavor ID, specify an appropriate value according to Appendix C.

In the example below, changes in the DB instance are applied immediately. If immediate application is enabled, the DB instance restarts after the API is executed.

| Command Example |
|--|
| <p>INSTANCEID=[<i>ID of the DB instance created in advance (For details about creating a DB instance, refer to section 2.2 (3).)</i>]</p> <p>FLAVOR=[<i>Flavor ID (Select any of the public flavor IDs.)</i>]</p> <p>APPLY_IMMEDIATELY=[<i>Immediate application of DB instance enabled/disabled (Select true or false)</i>]</p> <p>ENDPOINT=[<i>Endpoint of the database service</i>]</p> <pre>curl -X PUT -i \${ENDPOINT}/v1.0/\${TENANTID}/instances/\${INSTANCEID} -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d '{"instance": {"flavorRef": "\${FLAVOR}", "applyImmediately": "\${APPLY_IMMEDIATELY}"}}</pre> |
| Example of Execution Results |
| <pre>\$ INSTANCEID=test-instance-id \$ FLAVOR=i102 \$ APPLY_IMMEDIATELY=true \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X PUT -i \${ENDPOINT}/v1.0/\${TENANTID}/instances/\${INSTANCEID} -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d '{"instance": {"flavorRef": "\${FLAVOR}", "applyImmediately": "\${APPLY_IMMEDIATELY}"}}</pre> <p>HTTP/1.1 202 Accepted</p> <p>(Omitted)</p> |

4.2 Changing Data Disk Size

Change the data disk size as needed for operation.

When changing the data disk size, it is only possible to increase the size. You cannot reduce the size.

When changing the data disk size of a DB instance that has not been automatically backed up, create a DB snapshot in advance.

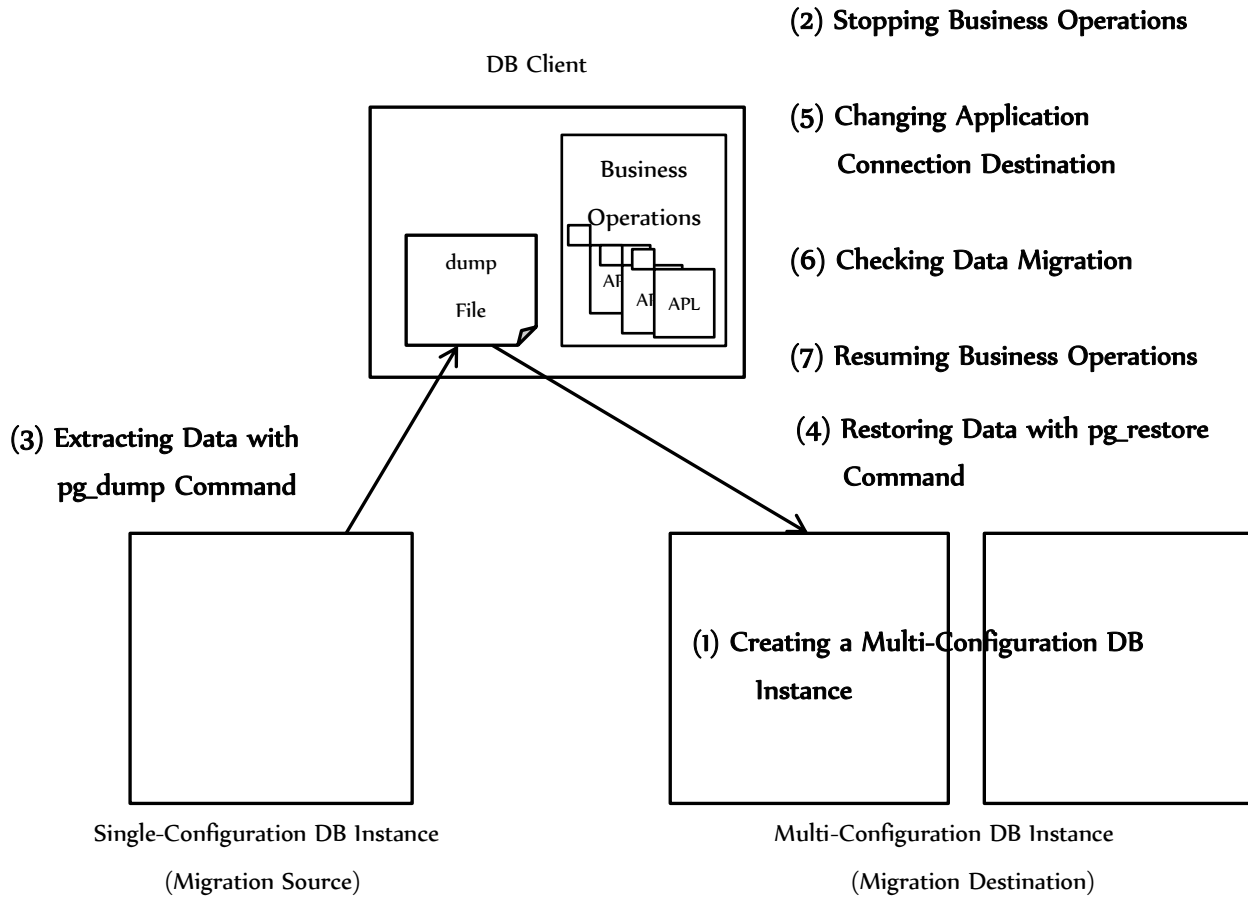
For details about creating a DB snapshot, refer to section 3.2.

In the example below, changes in the DB instance are applied immediately. If immediate application is enabled, the DB instance restarts after the API is executed.

| Command Example |
|--|
| <p>INSTANCEID=<i>[ID of the DB instance created in advance (For details about creating a DB instance, refer to section 2.2 (3).)]</i></p> <p>SIZE=<i>[Data disk size [GB] (Specify a desired value from 10 to 10240.)]</i></p> <p>APPLY_IMMEDIATELY=<i>[Immediate application of DB instance enabled/disabled (Select true or false)]</i></p> <p>ENDPOINT=<i>[Endpoint of the database service]</i></p> <pre>curl -X PUT -i \${ENDPOINT}/v1.0/\${TENANTID}/instances/\${INSTANCEID} -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d '{"instance": {"volume": {"size": \${SIZE}}, "applyImmediately": "\${APPLY_IMMEDIATELY}"}'</pre> |
| Example of Execution Results |
| <pre>\$ INSTANCEID=test-instance-id \$ SIZE=20 \$ APPLY_IMMEDIATELY=true \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X PUT -i \${ENDPOINT}/v1.0/\${TENANTID}/instances/\${INSTANCEID} -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d '{"instance": {"volume": {"size": \${SIZE}}, "applyImmediately": "\${APPLY_IMMEDIATELY}"}' HTTP/1.1 202 Accepted (Omitted)</pre> |

4.3 Changing from a Single Configuration to a Multi Configuration

This section describes the procedure to change a single-configuration DB instance to a multi-configuration DB instance. This procedure changes the IP address/FQDN.



(1) Creating a New Multi-Configuration DB Instance

Create a DB instance by specifying the following options for the DB instance creation API:

- multi
- multiAZ (for redundancy between AZ)

For a command example, refer to "Example of Creating a Multi-Configuration DB Instance" in section 2.2 (3).

(2) Stopping Business Operations

Stop business operations for the single-configuration DB instance that is in operation.

(3) Extracting Data

Perform this process on a DB client where the DB engine client (Enterprise Postgres Client or Symfoware Server Client) is installed.

Obtain the database data and schema from the migration source DB instance by executing the command shown below.

Execute this process the same number of times as the number of databases used in the migration source.

Example: Extracting data and schema from the database (in custom archive format)

| Command Example |
|--|
| <pre> CLIENT_DIR=[<i>Directory path where the DB engine client is installed</i>] export PATH=\${CLIENT_DIR}/bin:\${PATH} export LD_LIBRARY_PATH=\${CLIENT_DIR}/lib:\${LD_LIBRARY_PATH} FQDN=[<i>Connection destination of the migration source DB instance (For details about checking the FQDN, refer to section 2.3 (1).)</i>] PORT=[<i>Port number of the migration source DB instance (For details about checking the port number, refer to section 2.3 (1).)</i>] MASTERUSERNAME=[<i>Database administrator name for the migration source DB instance</i>] DB=[<i>DB name of the backup target</i>] pg_dump -h \${FQDN} -p \${PORT} -U \${MASTERUSERNAME} --format=custom \${DB} > K5DB_dump.custom </pre> |

(4) Inserting Data in a Multi-Configuration DB Instance

Perform this process on a DB client where the DB engine client is installed.

For the multi-configuration DB instance created in (1), restore the schema and data extracted in (3).

Example: Inserting data and schema

| Command Example |
|---|
| <pre> CLIENT_DIR=[<i>Directory path where the DB engine client is installed</i>] export PATH=\${CLIENT_DIR}/bin:\${PATH} export LD_LIBRARY_PATH=\${CLIENT_DIR}/lib:\${LD_LIBRARY_PATH} FQDN=[<i>Connection destination of the migration destination DB instance (For details about checking the FQDN, refer to section 2.3 (1).)</i>] PORT=[<i>Port number of the migration destination DB instance (For details about checking the port number, refer to section 2.3 (1).)</i>] MASTERUSERNAME=[<i>Database administrator name for the migration destination DB instance</i>] pg_restore -h \${FQDN} -p \${PORT} -U \${MASTERUSERNAME} -C -d postgres < K5DB_dump.custom </pre> |

(5) Correcting Application

Change the application connection destination of the database to the new multi-configuration DB instance created in (1).

(6) Resuming Business Operations

Resume your business operations.

- For details about the `pg_dump` command and `pg_restore` command, refer to the following sections in the "PostgreSQL 9.6.2 Documentation" or the "PostgreSQL 9.2.4 Documentation":
 - "VI. Reference" - "II. PostgreSQL Client Applications" - "`pg_dump`"
 - "VI. Reference" - "II. PostgreSQL Client Applications" - "`pg_restore`"
 - To obtain all database data and schema in one batch in step (3), use the `pg_dumpall` command. For details about the `pg_dumpall` command, refer to the following section in the "PostgreSQL 9.6.2 Documentation" or the "PostgreSQL 9.2.4 Documentation":
 - "VI. Reference" - "II. PostgreSQL Client Applications" - "`pg_dumpall`"
 - If the dump file is obtained as a SQL script file in step (3), restore the data using the `psql` command in step (4). For details about how to use the `psql` command, refer to the following section in the "PostgreSQL 9.6.2 Documentation" or the "PostgreSQL 9.2.4 Documentation":
 - "VI. Reference" - "II. PostgreSQL Client Applications" - "`psql`"
-

4.4 Changing a DB Instance

Using the DB instance change API allows you to also change the DB instance configuration for items other than those described in this chapter, when necessary.

For details about the parameters that can be changed, refer to "FUJITSU Cloud Service for OSS IaaS API Reference (Application Platform Service)".

When changing a DB instance, the application of changes may require the DB instance to get restarted, depending on the parameter.

The changes that require DB instances to get restarted when they are applied are shown below.

- When any of the following parameters is changed:

- `flavorRef`
- `volume`
- `multi`
- `multiAZ`
- `port`
- `engineVersion`

If "true" is specified for the `applyImmediately` parameter during execution of the DB instance change API, the DB instance restarts after execution of the API and the change can be applied immediately.

To disable immediate application of changes, specify "false" for the `applyImmediately` parameter. In this case, changes in a DB instance are applied when automatic maintenance is performed or when the DB instance is restarted manually.

For details about the DB instance change API, refer to "FUJITSU Cloud Service for OSS IaaS API Reference (Application Platform Service)" or section 4.1 and 4.2.

4.5 Changing a Database Parameter

This section describes how to make changes to the database parameters.

For parameter changes, the following two methods are available:

- (A) Create a new DB parameter group and apply it to the DB instance.
- (B) Change an existing DB parameter group that has already been applied to the DB instance.

(A) can change the parameters of individual DB instances.

(B) can change the parameters of all DB instances to which the target DB parameter group is applied.

For (A), perform all the steps from (1) to (5) below. For (B), perform steps (2), (3), and (5) below.

(1) Creating a DB Parameter Group

For details about the DB instance group creation API, refer to "FUJITSU Cloud Service for OSS IaaS API Reference (Application Platform Service)" or section 2.2 (2).

(2) Changing a Parameter Value

Change the parameter values of the DB parameter group. Change the parameters as needed for operation.

For details about the parameters that can be changed, refer to Appendix A.

The example below shows a case in which the "shared_buffers" parameter is changed.

| Command Example |
|---|
| <pre>PARAMG_ID=[ID of the DB parameter group created in advance] ENDPOINT=[Endpoint of the database service] curl -X PUT -i \${ENDPOINT}/v1.0/\${TENANTID}/parametergroups/\${PARAMG_ID} -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d "{\"parametergroup\": {\"parameters\": [{\"applyMethod\": \"pending-reboot\", \"name\": \"shared_buffers\", \"value\": \"10000\"}]}}"</pre> |
| Example of Execution Results |
| <pre>\$ PARAMG_ID=test-paramg-id \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X PUT -i \${ENDPOINT}/v1.0/\${TENANTID}/parametergroups/\${PARAMG_ID} -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d "{\"parametergroup\": {\"parameters\": [{\"applyMethod\": \"pending-reboot\", \"name\": \"shared_buffers\", \"value\": \"10000\"}]}}"</pre> <p>HTTP/1.1 200 OK</p> <p>(Omitted)</p> <pre>{\"parameterGroup\":{\"id\":\"XXXXXXXX\",\"description\":null,\"name\":\"XXXXXXXX\", \"parameters\":[{\"name\":\"shared_buffers\",\"value\":\"10000\",\"allowedValues\":\"\", \"applyMethod\":\"pending-reboot\", \"applyType\":\"static\", \"dataType\":\"integer\", \"description\":\"(8kB) Sets the number of shared memory buffers used by the server.\", \"minimumEngineVersion\":\"12.1\", \"source\":\"user\", \"isModifiable\":true}], \"parameterGroupFamily\":\"enterprisepostgres_v9.6\", \"appliedInstances\":null}}</pre> |

[Reference]

Though the SET command of PostgreSQL can also change parameters, the parameters changed by the SET command are applied only to the session, and parameter value in the DB parameter group are not affected.

(3) Checking the DB Parameter Group Information

Check the DB parameter group to confirm that the changes to the DB parameter group in step (2) have been applied.

| Command Example |
|--|
| <pre>PARAMG_ID=[ID of the DB parameter group changed in (2)] ENDPOINT=[Endpoint of the database service] curl -X GET -i \${ENDPOINT}/v1.0/\${TENANTID}/parametergroups/\${PARAMG_ID} -H "X-Auth-Token: \${TOKEN}"</pre> |
| Example of Execution Results |
| <pre>\$ PARAMG_ID=test-paramg-id \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X GET -i \${ENDPOINT}/v1.0/\${TENANTID}/parametergroups/\${PARAMG_ID} -H "X-Auth-Token: \${TOKEN}" HTTP/1.1 200 OK (Omitted) {"parameterGroup":{"id":"XXXXXXXX","description":null,"name":"XXXXXXXX","parameters":{"(Omitted) ,{"name":"shared_buffers","value":"10000", (Omitted) }},"parameterGroupFamily":"enterprisepostgres_v9.6","appliedInstances":null}}</pre> |

[Reference]

The values currently configured in a DB instance can be confirmed by connecting to that instance and executing the "SHOW ALL" SQL command.

For details on how to execute SQL commands, refer to section 2.3 (2).

(4) Applying a DB Parameter Group

Apply the new DB parameter group created above to an existing DB instance.

| Command Example |
|--|
| <p>INSTANCEID=[<i>ID of the DB instance created in advance (For details about creating a DB instance, refer to section 2.2 (3).)</i>]</p> <p>PARAMG_ID=[<i>ID of the DB parameter group created in (1)</i>]</p> <p>ENDPOINT=[<i>Endpoint of the database service</i>]</p> <pre>curl -X PUT -i \${ENDPOINT}/v1.0/\${TENANTID}/instances/\${INSTANCEID} -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d '{"instance": {"parameterGroupId": "\${PARAMG_ID}"}}</pre> |
| Example of Execution Results |
| <pre>\$ INSTANCEID=test-instance-id \$ PARAMG_ID=test-paramg-id \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X PUT -i \${ENDPOINT}/v1.0/\${TENANTID}/instances/\${INSTANCEID} -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d '{"instance": {"parameterGroupId": "\${PARAMG_ID}"}}</pre> <p>HTTP/1.1 202 Accepted</p> <p>(Omitted)</p> |

[Reference]

.....

The values currently configured for a DB instance can be confirmed by connecting to that instance and executing the "SHOW ALL" SQL command.

For details on how to execute SQL commands, refer to section 2.3 (2).

.....

(5) Restarting a DB Instance

When you change a parameter with "applyMethod" set to "pending-reboot", restart the DB instance.
 (For details about "applyMethod", refer to Appendix A.)

| Command Example |
|---|
| <p>INSTANCEID=[<i>ID of the DB instance created in advance (For details about creating a DB instance, refer to section 2.2 (3).)</i>]</p> <p>ENDPOINT=[<i>Endpoint of the database service</i>]</p> <pre>curl -X POST -i \${ENDPOINT}/v1.0/\${TENANTID}/instances/\${INSTANCEID}/action -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d '{"action": {"reboot": ""}}</pre> |
| Example of Execution Results |
| <pre>\$ INSTANCEID=test-instance-id \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X POST -i \${ENDPOINT}/v1.0/\${TENANTID}/instances/\${INSTANCEID}/action -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d '{"action": {"reboot": ""}}</pre> <p>HTTP/1.1 202 Accepted</p> <p>(Omitted)</p> |

[Reference]

.....
The values currently configured for a DB instance can be confirmed by connecting to that instance and executing the "SHOW ALL" SQL command.

For details on how to execute SQL commands, refer to section 2.3 (2).
.....

[Reference]

.....
If a DB instance is restarted with incorrect parameter values configured, the following event notification will be issued.

| source-type | category | message |
|-------------|----------------------|--|
| db-instance | configuration change | Modification of db parameters failed. See DB logfiles. |

In this case, the DB instance has been restarted, but the parameters that were configured incorrectly and the specification of "pending-reboot" for the "applyMethod" parameter have not been reflected.

A log regarding the incorrect parameters is output to the DB log file "start_error.log" file.

Refer to "start_error.log" and identify the incorrect parameters.

For details on viewing DB log files, refer to section 3.3 (2). The relevant log can be viewed by specifying "start_error.log" for "LOGFILENAME".

After identifying the incorrect parameters, refer to section 4.5 (2) and change their values.

If the value of the "applyMethod" parameter is "pending-reboot", refer to section 4.5 (5) and reboot the DB instance.
.....

The operations described in this chapter are performed by users that are assigned with the Administrator role, the System Owner role, or the Operator role.

5.1 Recovering a DB Instance

This section describes the procedure to recover a DB instance.

When a DB instance is recovered, the database is restored.

When a DB instance is backed up or restored, the master encryption key and the automatic open key used for the transparent data encryption function (refer to Appendix D) are also backed up or restored automatically.

(1) Recovering a DB Instance

There are two patterns for the procedure to recover a DB instance.

Recover a DB instance by using either Pattern 1 or Pattern 2 according to the operation policy.

Pattern 1: Recovering a DB Instance from a DB Snapshot

Recover a DB instance from a DB Snapshot created in advance.

With this pattern, the DB instance is recovered to the status of the "created" time for the DB snapshot.

With DB instance recovery from a DB snapshot, a new DB instance is created aside from the DB instance from which the DB snapshot is obtained. Because a new DB instance is used, a new FQDN/IP is created and you will be charged for this instance separately from the DB instance from which the DB snapshot is obtained.

| Command Example |
|---|
| <p>AZ=[AZ name to be specified as the creation destination of the DB instance (Specify the same value as with the recovery source DB instance.)]</p> <p>SNAPSHOTID=[ID of the backup source snapshot (Select any of the IDs created in advance. For details on how to check a snapshot ID, refer to section 3.2 (2)).]</p> <p>RESTOREINSTID=[ID of the DB instance to be created for restoration (Specify a desired value.)]</p> <p>SUBNETGROUPID=[ID of the DB subnet group created in advance (Specify the same value as with the recovery source DB instance.)]</p> <p>FLAVOR=[Flavor ID (Specify the same value as with the recovery source DB instance.)]</p> <p>MASTERUSERNAME=[Database administrator name (Specify the same value as with the recovery source DB instance.)]</p> <p>MASTERUSERPASSWORD=[Database administrator password (Specify the same value as with the recovery source DB instance.)]</p> <p>PORT=[Port number specified during DB instance creation (Specify the same value as with the recovery source DB instance.)]</p> <p>SECGRP_ID=[ID of the security group created in advance (Specify the same value as with the recovery source DB instance.)]</p> <p>PARAMG_ID=[ID of the DB parameter group created in advance (Specify the same value as with the recovery source DB instance.)]</p> <p>SIZE=[Data disk size [GB] (Specify the same value as with the recovery source DB instance.)]</p> <p>BACKUP_RETENTION_PERIOD=[Backup retention period [day] (Specify the same value as with the recovery source DB instance.)]</p> <p>PREFERRED_BACKUP_WINDOW=[Backup time [UTC] (Specify the same value as with the recovery source DB instance.)]</p> <p>AUTO_MAINTENANCE=[Automatic maintenance enabled/disabled (Specify the same value as with the recovery source DB instance.)]</p> <p>PREFERRED_MAINTENANCE_WINDOW=[Maintenance time [UTC] (Specify the same value as with the recovery source DB instance.)]</p> <p>ENDPOINT=[Endpoint of the database service]</p> <pre>curl -X POST -i \${ENDPOINT}/v1.0/\${TENANTID}/instances -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d "{\"action\": { \"restoresnapshot\": \"\", \"instance\": { \"multiAZ\": \"false\", \"multi\": \"false\", \"availabilityZone\": \"\${AZ}\", \"subnetGroup\": { \"subnetGroup\": \"\", \"masterUserName\": \"\${MASTERUSERNAME}\", \"publiclyAccessible\": \"false\", \"flavorRef\": \"\${FLAVOR}\", \"masterUserPassword\": \"\${MASTERUSERPASSWORD}\", \"port\": \${PORT}, \"volume\": { \"type\": \"MI\", \"size\": \${SIZE}}, \"preferredMaintenanceWindow\": \"\${PREFERRED_MAINTENANCE_WINDOW}\", \"parameterGroup\": \"\${PARAMG_ID}\", \"securityGroups\": [{ \"securityGroup\": \"\${SECGRP_ID}\" }], \"backupRetentionPeriod\": \${BACKUP_RETENTION_PERIOD}, \"id\": \"\${RESTOREINSTID}\", \"preferredBackupWindow\": \"\${PREFERRED_BACKUP_WINDOW}\", \"autoMaintenance\": \"\${AUTO_MAINTENANCE}\", \"snapshot\": { \"id\": \"\${SNAPSHOTID}\" } } } }</pre> |

* For MASTERUSERPASSWORD, specify the same value as that specified for "masterUserPassword" on the date and time at which the snapshot specified for SNAPSHOTID was created.

If you specify "1" or more for BACKUP_RETENTION_PERIOD, the first backup operation will be performed after the DB instance is recovered. Until the backup is completed, "status" remains as "Build". When backup is completed, "status" changes to "Active".

Example of Execution Results

```
$ AZ=jp-east-1a
$ SNAPSHOTID=test-snapshot-id
$ RESTOREINSTID=test-restore-instance-id
$ SUBNETGROUPID=test-subnetgroup-id
$ FLAVOR=i101
$ MASTERUSERNAME="masterusername"
$ MASTERUSERPASSWORD="masteruserpassword"
$ PORT=26500
$ SECGRP_ID=xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx
$ PARAMG_ID=test-paramg-id
$ SIZE=10
$ BACKUP_RETENTION_PERIOD=5
$ PREFERRED_BACKUP_WINDOW="16:00-16:30"
$ AUTO_MAINTENANCE=true
$ PREFERRED_MAINTENANCE_WINDOW="Sat:18:00-Sat:18:30"
$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com
$ curl -X POST -i ${ENDPOINT}/v1.0/${TENANTID}/instances -H "X-Auth-Token: ${TOKEN}" -H "Content-Type: application/json" -d
"{\"action\": {\"restoresnapshot\": \"\"}, \"instance\": {\"multiAZ\": \"false\", \"multi\": \"false\", \"availabilityZone\": \"${AZ}\", \"subnetGroup\":
\"${SUBNETGROUPID}\", \"masterUserName\": \"${MASTERUSERNAME}\", \"publiclyAccessible\": \"false\", \"flavorRef\": \"${FLAVOR}\",
\"masterUserPassword\": \"${MASTERUSERPASSWORD}\", \"port\": ${PORT}, \"volume\": {\"type\": \"M1\", \"size\": ${SIZE}},
\"preferredMaintenanceWindow\": \"${PREFERRED_MAINTENANCE_WINDOW}\", \"parameterGroup\": \"${PARAMG_ID}\", \"securityGroups\":
[\"securityGroup\": \"${SECGRP_ID}\"], \"backupRetentionPeriod\": ${BACKUP_RETENTION_PERIOD}, \"id\": \"${RESTOREINSTID}\",
\"preferredBackupWindow\": \"${PREFERRED_BACKUP_WINDOW}\", \"autoMaintenance\": \"${AUTO_MAINTENANCE}\", \"snapshot\": {\"id\":
\"${SNAPSHOTID}\"}}}"
HTTP/1.1 202 Accepted
(Omitted)
```

Pattern 2: Recovering a DB Instance with Point-in-Time Recovery

You can specify a point in time during the backup retention period and set the DB instance to be recovered to the status of the specified time.

The latest period of time available for restoration is normally the past five minutes from the current time.

During DB instance recovery with point-in-time recovery, a new DB instance is created aside from the DB instance of the recovery source. Because a new DB instance is used, a new FQDN/IP is created and you will be charged for the instance separately from the DB instance of the recovery source.

| Command Example |
|---|
| <p>AZ=[AZ name to be specified as the creation destination of the DB instance (Specify the same value as with the recovery source DB instance.)]</p> <p>INSTANCEID=[ID of the recovery source DB instance (The one created in advance. For details about creating a DB instance, refer to section 2.2 (3).)]</p> <p>PITR_INSTID=[ID of the DB instance to be created with point-in-time recovery (Specify a desired value.)]</p> <p>USE_LATEST_RESTORABLE_TIME=[Restoration to the latest restorable point enabled/disabled (Specify true or false.)]</p> <p>SUBNETGROUPID=[ID of the DB subnet group created in advance (Specify the same value as with the recovery source DB instance.)]</p> <p>FLAVOR=[Flavor ID (Specify the same value as with the recovery source DB instance.)]</p> <p>MASTERUSERNAME=[Database administrator name (Specify the same value as with the recovery source DB instance.)]</p> <p>MASTERUSERPASSWORD=[Database administrator password (Specify the same value as with the recovery source DB instance.)]</p> <p>PORT=[Port number specified during DB instance creation (Specify the same value as with the recovery source DB instance.)]</p> <p>SECGRP_ID=[ID of the security group created in advance (Specify the same value as with the recovery source DB instance.)]</p> <p>PARAMG_ID=[ID of the DB parameter group created in advance (Specify the same value as with the recovery source DB instance.)]</p> <p>SIZE=[Data disk size [GB] (Specify the same value as with the recovery source DB instance.)]</p> <p>BACKUP_RETENTION_PERIOD=[Backup retention period [day] (Specify the same value as with the recovery source DB instance.)]</p> <p>PREFERRED_BACKUP_WINDOW=[Backup time [UTC] (Specify the same value as with the recovery source DB instance.)]</p> <p>AUTO_MAINTENANCE=[Automatic maintenance enabled/disabled (Specify the same value as with the recovery source DB instance.)]</p> <p>PREFERRED_MAINTENANCE_WINDOW=[Maintenance time [UTC] (Specify the same value as with the recovery source DB instance.)]</p> <p>ENDPOINT=[Endpoint of the database service]</p> <pre>curl -X POST -i \${ENDPOINT}/v1.0/\${TENANTID}/instances/\${INSTANCEID} -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d "{\"action\": {\"restoretointime\": \"\", \"restore\": {\"useLatestRestorableTime\": \"\${USE_LATEST_RESTORABLE_TIME}\"}, \"instance\": {\"multiAZ\": \"false\", \"multi\": \"false\", \"availabilityZone\": \"\${AZ}\", \"publiclyAccessible\": \"false\", \"subnetGroupId\": \"\${SUBNETGROUPID}\", \"masterUserName\": \"\${MASTERUSERNAME}\", \"flavorRef\": \"\${FLAVOR}\", \"masterUserPassword\": \"\${MASTERUSERPASSWORD}\", \"port\": \${PORT}, \"volume\": {\"type\": \"M1\", \"size\": \${SIZE}}, \"preferredMaintenanceWindow\": \"\${PREFERRED_MAINTENANCE_WINDOW}\", \"parameterGroupId\": \"\${PARAMG_ID}\", \"securityGroupIds\": [{\"securityGroupId\": \"\${SECGRP_ID}\"}], \"backupRetentionPeriod\": \${BACKUP_RETENTION_PERIOD}, \"id\": \"\${PITR_INSTID}\", \"preferredBackupWindow\": \"\${PREFERRED_BACKUP_WINDOW}\", \"autoMaintenance\": \"\${AUTO_MAINTENANCE}\"}}<!--" </pre--> </pre> |

* If you specify "restoretointime", for MASTERUSERPASSWORD specify the same value as that of "masterUserPassword" of the instance at the specified time. If you specify "true" for "useLatestRestorableTime", specify the current "masterUserPassword" of the DB instance.

If you specify "1" or more for BACKUP_RETENTION_PERIOD, the first backup operation will be performed after the DB instance is

recovered. Until the backup is completed, "status" remains as "Build". When backup is completed, "status" changes to "Active".

```
Example of Execution Results

$ AZ=jp-east-1a
$ INSTANCEID=test-instance-id
$ PITR_INSTID=test-pitr-instance-id
$ USE_LATEST_RESTOREABLE_TIME=true
$ SUBNETGROUPID=test-subnetgroup-id
$ FLAVOR=1101
$ MASTERUSERNAME="masterusername"
$ MASTERUSERPASSWORD="masteruserpassword"
$ PORT=26500
$ SECGRP_ID=xxxxxxxx-xxxx-xxxx-xxxxxxxxxxxx
$ PARAMG_ID=test-paramg-id
$ SIZE=10
$ BACKUP_RETENTION_PERIOD=5
$ PREFERRED_BACKUP_WINDOW="16:00-16:30"
$ AUTO_MAINTENANCE=true
$ PREFERRED_MAINTENANCE_WINDOW="Sat:18:00-Sat:18:30"
$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com

$ curl -X POST -i ${ENDPOINT}/v1.0/${TENANTID}/instances/${INSTANCEID} -H "X-Auth-Token: ${TOKEN}" -H "Content-Type: application/json" -d "{\"action\": {\"restoretointime\": \"\"}, \"restore\": {\"useLatestRestoreableTime\": \"${USE_LATEST_RESTOREABLE_TIME}\"}, \"instance\": {\"multiAZ\": \"false\", \"multi\": \"false\", \"availabilityZone\": \"${AZ}\", \"publiclyAccessible\": \"false\", \"subnetGroupId\": \"${SUBNETGROUPID}\", \"masterUserName\": \"${MASTERUSERNAME}\", \"flavorRef\": \"${FLAVOR}\", \"masterUserPassword\": \"${MASTERUSERPASSWORD}\", \"port\": ${PORT}, \"volume\": {\"type\": \"MI\", \"size\": ${SIZE}}, \"preferredMaintenanceWindow\": \"${PREFERRED_MAINTENANCE_WINDOW}\", \"parameterGroupId\": \"${PARAMG_ID}\", \"securityGroupIds\": [{\"securityGroupId\": \"${SECGRP_ID}\"}], \"backupRetentionPeriod\": ${BACKUP_RETENTION_PERIOD}, \"id\": \"${PITR_INSTID}\", \"preferredBackupWindow\": \"${PREFERRED_BACKUP_WINDOW}\", \"autoMaintenance\": \"${AUTO_MAINTENANCE}\"}"

HTTP/1.1 202 Accepted
(Omitted)
```


(2) Checking DB Instance Recovery

Immediately after (1) is executed, "status" of the recovered DB instance is "Build". After confirming that "status" has changed to "Active" by using the information reference API of the recovered DB instance, perform the procedure described in (3).

For details about the DB instance information reference API, refer to "FUJITSU Cloud Service for OSS IaaS API Reference (Application Platform Service)" or section 2.3 (1).

The time required to complete the DB instance recovery process varies depending on the data volume.

If "status" shows "Error", recovery has failed. Delete the DB instance indicated with "Error" and execute the procedure from (1) again.

For details about deleting a DB instance, refer to "FUJITSU Cloud Service for OSS IaaS API Reference (Application Platform Service)" or section 6.1 (1).

(3) Correcting Application

Change the application connection destination of the database to the DB instance created in (1).

(4) Deleting a Recovery Source DB Instance

Delete a recovery source DB instance that is no longer necessary.

For details about deleting a DB instance, refer to section 6.1 (1).

5.2 Recovering a Multi-Configuration DB Instance

This section describes the procedure to recover a multi-configuration DB instance.

(1) Operation when an Error Occurs

If an error occurs on a multi-configuration DB instance, the operation is switched to a single-system operation.

It takes approximately two minutes for the system to change to single-system operation after an error occurs.

During this period, SQL connection is disabled. Thus, create an application with which SQL is retried for two minutes or more.

During degenerated single-system operation and during redundancy recovery, the DB instance status is "Degenerated".

(2) Auto Recovery of a DB Instance

When a multi-configuration DB instance is switched to single-system operation, auto recovery is activated.

Auto recovery starts at the time set for "preferredRecoveryTime" that is specified for each DB instance.

Immediate recovery is specified by default.

To specify preferredRecoveryTime, use the DB instance change API.

Set the time according to the operation policy, as the performance of issuing SQL statements lowers during auto recovery.

Redundancy recovery may take more than 30 minutes, depending on the data volume.

(3) Operation after Auto Recovery

If an error occurs on the primary DB instance and the auto recovery is performed, the operation of the primary and secondary DBs are reversed after the auto recovery. In this situation, the DB instance status is "Switched".

In the "Switched" status, the redundancy of DB instances has been recovered and operation can continue.

To reverse operation of the primary and secondary DBs from the "Switched" status again, perform a failover using the DB instance restart API.

During this period, SQL statements cannot be issued. Be careful when you execute restart.

| Command Example |
|---|
| <pre>INSTANCEID=[ID of a multi-configuration DB instance created in advance (For details about creating a DB instance, refer to section 2.2 (3).)] ENDPOINT=[Endpoint of the database service] curl -X POST -i \${ENDPOINT}/v1.0/\${TENANTID}/instances/\${INSTANCEID}/action -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d '{"action": {"failover": "true", "reboot": ""}}'</pre> |
| Example of Execution Results |
| <pre>\$ INSTANCEID=test-multi-instance-id \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X POST -i \${ENDPOINT}/v1.0/\${TENANTID}/instances/\${INSTANCEID}/action -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d '{"action": {"failover": "true", "reboot": ""}}' HTTP/1.1 202 Accepted (Omitted)</pre> |

The operations described in this chapter are performed by users that are assigned with the Administrator role or the System Owner role.

6.1 Deleting a DB Instance

This section describes the procedure to delete a DB instance.

(1) Deleting a DB Instance

Delete a DB instance.

| Command Example |
|---|
| <pre> INSTANCEID=[ID of a DB instance created in advance (For details about creating a DB instance, refer to section 2.2 (3).)] ENDPOINT=[Endpoint of database service] curl -X DELETE -i \${ENDPOINT}/v1.0/\${TENANTID}/instances/\${INSTANCEID} -H "X-Auth-Token: \${TOKEN}" </pre> |
| Example of Execution Results |
| <pre> \$ INSTANCEID=test-instance-id \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X DELETE -i \${ENDPOINT}/v1.0/\${TENANTID}/instances/\${INSTANCEID} -H "X-Auth-Token: \${TOKEN}" HTTP/1.1 202 Accepted </pre> |

Before performing the procedure in (2), make sure that the DB instance has been deleted using the information reference API of the DB instance.

Execute the information reference API of the DB instance to confirm that "status" of the DB instance is "Deleted" or that the information of the DB instance cannot be obtained ("404" is returned for the request status).

For details about the DB instance information reference API, refer to "FUJITSU Cloud Service for OSS IaaS API Reference (Application Platform Service)" or section 2.3 (1).

(2) Deleting a DB Parameter Group

Delete a DB parameter group. When a DB parameter group is being used by a DB instance, the DB parameter group cannot be deleted. ("400" is returned for the response status.)

Delete a DB parameter group only when it is not being used by any DB instances.

| Command Example |
|---|
| <pre> PARAMG_ID=[ID of a DB parameter group created in advance (For details about creating a DB parameter group, refer to section 2.2 (2).)] ENDPOINT=[Endpoint of database service] curl -X DELETE -i \${ENDPOINT}/v1.0/\${TENANTID}/parametergroups/\${PARAMG_ID} -H "X-Auth-Token: \${TOKEN}" </pre> |

| Example of Execution Results |
|--|
| <pre> \$ PARAMG_ID=test-paramg-id \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X DELETE -i \${ENDPOINT}/v1.0/\${TENANTID}/parametergroups/\${PARAMG_ID} -H "X-Auth-Token: \${TOKEN}" HTTP/1.1 200 OK </pre> |

(3) Deleting a DB Subnet Group

Delete a DB subnet group. When a DB subnet group is being used by a DB instance, the DB subnet group cannot be deleted. ("400" is returned for the response status.)

Delete a DB subnet group only when it is not being used by any DB instance.

| Command Example |
|--|
| <pre> SUBNETGROUPID=[ID of a DB subnet group created in advance (For details about creating a DB subnet group, refer to section 2.2 (1).)] ENDPOINT=[Endpoint of database service] curl -X DELETE -i \${ENDPOINT}/v1.0/\${TENANTID}/subnetgroups/\${SUBNETGROUPID} -H "X-Auth-Token: \${TOKEN}" </pre> |
| Example of Execution Results |
| <pre> \$ SUBNETGROUPID=test-subnetgroup-id \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X DELETE -i \${ENDPOINT}/v1.0/\${TENANTID}/subnetgroups/\${SUBNETGROUPID} -H "X-Auth-Token: \${TOKEN}" HTTP/1.1 200 OK </pre> |

Appendix A: List of DB Parameters That Can Be Changed

A.1 List of DB Parameters That Can Be Changed

A list of parameters that can be specified for the database is shown below.

You can change the database parameters as described in section 4.5.

Note that the parameters with "ismodifiable" set to "f" cannot be changed.

For a parameter with "appplymethod" set to "pending-reboot", changes are applied when the DB instance is restarted.

For a parameter with "appplymethod" set to "immediate", changes are applied immediately after the parameter value change API is issued.

For details about the parameters, refer to the Enterprise Postgres manuals or the Symfoware Server manuals and specify appropriate values according to the operation policy.

* When changing a parameter with "allowedvalues" set to a value that is enclosed in both double and single quotation marks as "'...'", the value specified with the parameter value change API must also be enclosed in both double and single quotation marks as "'...'".

* In the tables below, "-" (hyphen) represents "no limit".

* In the tables below, items in yellow are parameters whose use will affect the amount of memory used.

To avoid running out of memory, estimate the amount of memory that will be used before configuring these items.

For details on estimating memory use, refer to Appendix A.2.

* In the tables below, the "max_connections" parameter specifies the number of connections which can be connected to the database at the same time.

As each database service also uses a connection, be sure to add these connections to the value of this parameter. Add one connection when using a single configuration, and add three connections when using a redundant configuration.

When changing the following parameter value of DB parameter group, specify only the numerical value in the unit shown below. Because when you specify the unit together it becomes abnormal.

| name | unit |
|------------------------|----------------|
| authentication_timeout | seconds |
| maintenance_work_mem | KB |
| shared_buffers | blocks of 8 KB |
| temp_buffers | blocks of 8 KB |
| work_mem | KB |
| bgwrite_delay | milliseconds |
| wal_writer_delay | milliseconds |
| checkpoint_timeout | seconds |
| checkpoint_warning | seconds |
| effective_cache_size | blocks of 8 KB |
| autovacuum_naptime | seconds |
| deadlock_timeout | milliseconds |

(I) List of DB parameters that can be changed in Enterprise Postgres 9.6

| name | default value | min | max | allowedvalues | apppmethod | ismodifi able |
|--------------------------------|--------------------------------------|-----|---------|--|----------------|------------------|
| archive_mode | on | - | - | {on,off} | pending-reboot | f |
| default_transaction_isolation | "read committed" | - | - | {serializable','repeatable read''','read committed''','read uncommitted''} | immediate | t |
| bonjour | off | - | - | - | pending-reboot | f |
| bonjour_name | " | - | - | - | pending-reboot | f |
| krb_caseins_users | true | - | - | - | pending-reboot | f |
| default_text_search_config | 'pg_catalog.english' | - | - | - | pending-reboot | t |
| data_directory | '/userdata/data' | - | - | - | pending-reboot | f |
| listen_addresses | '*' | - | - | - | pending-reboot | f |
| dynamic_library_path | '\$libdir' | - | - | - | pending-reboot | f |
| ssl | off | - | - | {on,off} | pending-reboot | t |
| hot_standby | off | - | - | {on,off} | pending-reboot | f |
| hot_standby_feedback | off | - | - | {on,off} | pending-reboot | f |
| krb_server_keyfile | " | - | - | - | pending-reboot | f |
| local_preload_libraries | " | - | - | - | pending-reboot | f |
| logging_collector | on | - | - | {on,off} | pending-reboot | f |
| max_connections | 100 | 7 | 262143 | - | pending-reboot | t |
| max_files_per_process | 1000 | 25 | - | - | pending-reboot | t |
| max_locks_per_transaction | 64 | 10 | 1000000 | - | pending-reboot | t |
| max_pred_locks_per_transaction | 64 | 10 | 2000000 | - | pending-reboot | t |
| max_prepared_transactions | 0 | 0 | 1000000 | - | pending-reboot | t |
| log_checkpoints | off | - | - | {on,off} | immediate | t |
| max_wal_senders | 12 | - | - | - | pending-reboot | f |
| shared_buffers | 4096 | 16 | 7602176 | - | pending-reboot | t |
| shared_preload_libraries | 'pgx_datamasking,block_alter_system' | - | - | - | pending-reboot | f |
| ssl_ca_file | " | - | - | - | pending-reboot | f |
| ssl_ciphers | 'HIGH:MEDI-UM:+3DES:!aNULL' | - | - | - | pending-reboot | f |
| ssl_crl_file | " | - | - | - | pending-reboot | f |
| ssl_prefer_server_ciphers | on | - | - | - | pending-reboot | f |
| ssl_ecdh_curve | 'prime256v1' | - | - | - | pending-reboot | f |

| name | default value | min | max | allowedvalues | applymethod | ismodifiable |
|--------------------------------|---|-----------|------------|------------------------|----------------|--------------|
| superuser_reserved_connections | 5 | - | - | - | pending-reboot | f |
| synchronous_standby_names | " | - | - | - | pending-reboot | f |
| syslog_ident | 'postgres' | - | - | - | pending-reboot | f |
| tablespace_path_prefix | '/userdata/tblspc' | - | - | - | pending-reboot | f |
| enable_indexonlyscan | on | - | - | {on,off} | pending-reboot | t |
| temp_file_limit | -1 | -1 | 2147483647 | - | pending-reboot | t |
| timezone_abbreviations | 'Default' | - | - | - | pending-reboot | f |
| ssl_key_file | '/opt/dbaas/ssl_keys/server.key' | - | - | - | pending-reboot | f |
| max_standby_archive_delay | 30000 | -1 | 2147483647 | - | immediate | t |
| track_activity_query_size | 1024 | 100 | 102400 | - | pending-reboot | t |
| unix_socket_directory | " | - | - | - | pending-reboot | f |
| unix_socket_group | " | - | - | - | pending-reboot | f |
| hba_file | '/userdata/data/pg_hba.conf' | - | - | - | pending-reboot | f |
| ident_file | '/userdata/data/pg_ident.conf' | - | - | - | pending-reboot | f |
| autovacuum_freeze_max_age | 200000000 | 100000000 | 200000000 | - | pending-reboot | t |
| autovacuum_max_workers | 3 | 1 | 1000 | - | pending-reboot | t |
| unix_socket_permissions | 0777 | - | - | - | pending-reboot | f |
| wal_buffers | -1 | -1 | 7602176 | - | pending-reboot | t |
| wal_level | 'replica' | - | - | {replica,logical} | pending-reboot | t |
| external_pid_file | " | - | - | - | pending-reboot | f |
| track_counts | on | - | - | {on,off} | immediate | t |
| archive_command | "/opt/dbaas/client_config/local_backup_script.sh %p %f" | - | - | - | pending-reboot | f |
| keystore_location | '/userdata/data' | - | - | - | pending-reboot | f |
| ssl_cert_file | '/opt/dbaas/ssl_keys/server.crt' | - | - | - | pending-reboot | f |
| archive_timeout | 300 | - | - | - | pending-reboot | f |
| autovacuum_vacuum_threshold | 50 | 0 | 2147483647 | - | immediate | t |
| backslash_quote | safe_encoding | - | - | {safe_encoding,on,off} | immediate | t |

| name | default value | min | max | allowedvalues | applymethod | ismodifiable |
|------------------------------|---------------|-----|------------------------|---|-------------|--------------|
| bgwriter_delay | 200 | 10 | 10000 | - | immediate | t |
| bgwriter_lru_maxpages | 100 | 0 | 1000 | - | immediate | t |
| bgwriter_lru_multiplier | 2.0 | 0 | 10 | - | immediate | t |
| bgwriter_flush_after | 64 | 0 | 256 | - | immediate | t |
| bytea_output | 'hex' | - | - | {escape,hex} | immediate | t |
| check_function_bodies | on | - | - | {on,off} | immediate | t |
| vacuum_freeze_min_age | 50000000 | 0 | 100000000 | - | immediate | t |
| checkpoint_completion_target | 0.5 | 0 | 1 | - | immediate | t |
| checkpoint_timeout | 3600 | 30 | 3600 | - | immediate | t |
| checkpoint_warning | 30 | 0 | 2147483647 | - | immediate | t |
| checkpoint_flush_after | 32 | 0 | 256 | - | immediate | t |
| client_encoding | sql_ascii | - | - | - | immediate | t |
| client_min_messages | notice | - | - | {debug5,debug4,debug3,debug2,debug1,log,notice,warning,error} | immediate | t |
| commit_delay | 0 | 0 | 100000 | - | immediate | t |
| commit_siblings | 5 | 0 | 1000 | - | immediate | t |
| constraint_exclusion | partition | - | - | {partition,on,off} | immediate | t |
| cpu_index_tuple_cost | 0.005 | 0 | 1.797690000000001e+308 | - | immediate | t |
| cpu_operator_cost | 0.0025 | 0 | 1.797690000000001e+308 | - | immediate | t |
| cpu_tuple_cost | 0.01 | 0 | 1.797690000000001e+308 | - | immediate | t |
| cursor_tuple_fraction | 0.1 | 0 | 1 | - | immediate | t |
| db_user_namespace | off | - | - | {on,off} | immediate | f |
| deadlock_timeout | 1000 | 1 | 2147483647 | - | immediate | t |

| name | default value | min | max | allowedvalues | applymethod | ismodifiable |
|--------------------------------|---------------|-----|------------|---------------|-------------|--------------|
| debug_pretty_print | on | - | - | {on,off} | immediate | t |
| debug_print_parse | off | - | - | {on,off} | immediate | t |
| debug_print_plan | off | - | - | {on,off} | immediate | t |
| debug_print_rewritten | off | - | - | {on,off} | immediate | t |
| default_statistics_target | 100 | 1 | 10000 | - | immediate | t |
| default_tablespace | " | - | - | - | immediate | f |
| default_transaction_deferrable | off | - | - | {on,off} | immediate | t |
| full_page_writes | on | - | - | {on,off} | immediate | t |
| default_transaction_read_only | off | - | - | {on,off} | immediate | t |
| default_with_oids | off | - | - | {on,off} | immediate | t |
| effective_cache_size | 16384 | 1 | 2147483647 | - | immediate | t |
| effective_io_concurrency | 1 | 0 | 1000 | - | immediate | t |
| enable_bitmapscan | on | - | - | {on,off} | immediate | t |
| enable_hashagg | on | - | - | {on,off} | immediate | t |
| enable_hashjoin | on | - | - | {on,off} | immediate | t |
| enable_indexscan | on | - | - | {on,off} | immediate | t |
| enable_material | on | - | - | {on,off} | immediate | t |
| enable_mergejoin | on | - | - | {on,off} | immediate | t |
| enable_nestloop | on | - | - | {on,off} | immediate | t |
| enable_seqscan | on | - | - | {on,off} | immediate | t |
| enable_sort | on | - | - | {on,off} | immediate | t |
| enable_tidscan | on | - | - | {on,off} | immediate | t |
| escape_string_warning | on | - | - | {on,off} | immediate | t |
| exit_on_error | off | - | - | {on,off} | immediate | f |
| extra_float_digits | 0 | -15 | -3 | - | immediate | t |
| from_collapse_limit | 8 | 1 | 2147483647 | - | immediate | t |
| fsync | on | - | - | {on,off} | immediate | t |
| geqo_effort | 5 | 1 | 10 | - | immediate | t |
| geqo_generations | 0 | 0 | 2147483647 | - | immediate | t |
| geqo_pool_size | 0 | 0 | 2147483647 | - | immediate | t |

| name | default value | min | max | allowedvalues | applymethod | ismodifiable |
|-----------------------------|----------------------------------|-----|----------------|--|-------------|--------------|
| geqo_seed | 0.0 | 0 | 1 | - | immediate | t |
| geqo_selection_bias | 2.0 | 1.5 | 2 | - | immediate | t |
| geqo_threshold | 12 | 2 | 2147483 647 | - | immediate | t |
| intervalstyle | 'postgres' | - | - | {postgres,postgres_verbose,sql_standard,iso_8601} | immediate | t |
| join_collapse_limit | 8 | 1 | 2147483 647 | - | immediate | t |
| lc_messages | 'en_US.UTF-8' | - | - | - | immediate | t |
| lc_monetary | 'en_US.UTF-8' | - | - | - | immediate | t |
| lc_numeric | 'en_US.UTF-8' | - | - | - | immediate | t |
| lc_time | 'en_US.UTF-8' | - | - | - | immediate | t |
| geqo | on | - | - | {on,off} | immediate | t |
| lo_compat_privileges | off | - | - | {on,off} | immediate | f |
| log_autovacuum_min_duration | -1 | -1 | 2147483 647 | - | immediate | t |
| log_connections | off | - | - | {on,off} | immediate | t |
| log_destination | 'stderr' | - | - | - | immediate | f |
| log_directory | 'pg_log' | - | - | - | immediate | f |
| log_disconnections | off | - | - | {on,off} | immediate | t |
| log_duration | off | - | - | {on,off} | immediate | t |
| log_error_verbosity | default | - | - | {terse,default,verbose} | immediate | t |
| log_executor_stats | off | - | - | {on,off} | immediate | t |
| log_file_mode | 0600 | - | - | - | immediate | f |
| log_filename | 'postgresql.log-%Y-%m-%d_%H%M%S' | - | - | {postgresql.log.%Y-%m-%d,postgresql.log.%Y-%m-%d-%H} | immediate | t |
| log_hostname | off | - | - | {on,off} | immediate | t |
| log_line_prefix | " | - | - | - | immediate | t |
| log_lock_waits | off | - | - | {on,off} | immediate | t |
| log_min_duration_statement | -1 | -1 | 2147483 647 | - | immediate | t |
| log_min_error_statement | error | - | - | {debug5,debug4,debug3,debug | immediate | t |

| name | default value | min | max | allowedvalues | applymethod | ismodifiable |
|-----------------------------|---------------|------|------------------------|--|-------------|--------------|
| ent | | | | 2,debug1,info,notice,warning,error,log,fatal,panic} | | |
| log_min_messages | warning | - | - | {debug5,debug4,debug3,debug2,debug1,info,notice,warning,error,log,fatal,panic} | immediate | t |
| log_planner_stats | off | - | - | {on,off} | immediate | t |
| log_replication_commands | off | - | - | {on,off} | immediate | t |
| log_rotation_age | 1440 | 1 | 1440 | - | immediate | t |
| log_rotation_size | 10240 | 0 | 2097151 | - | immediate | t |
| log_statementa | 'none' | - | - | {none,ddl,mod,all} | immediate | t |
| log_statement_stats | off | - | - | {on,off} | immediate | t |
| log_temp_files | -1 | -1 | 2147483647 | - | immediate | t |
| log_truncate_on_rotation | on | - | - | {on,off} | immediate | f |
| maintenance_work_mem | 16384 | 1024 | 2147483647 | - | immediate | t |
| log_parser_stats | off | - | - | {on,off} | immediate | t |
| max_stack_depth | 2048 | 100 | 9216 | - | immediate | t |
| max_standby_streaming_delay | 30000 | -1 | 2147483647 | - | immediate | t |
| password_encryption | on | - | - | {on,off} | immediate | f |
| quote_all_identifiers | off | - | - | {on,off} | immediate | t |
| random_page_cost | 4.0 | 0 | 1.797690000000001e+308 | - | immediate | t |
| restart_after_crash | on | - | - | {on,off} | immediate | f |
| seq_page_cost | 1.0 | 0 | 1.79769e+308 | - | immediate | t |
| session_replication_role | 'origin' | - | - | {origin,replica,local} | immediate | t |
| sql_inheritance | on | - | - | {on,off} | immediate | t |
| standard_conforming_strings | on | - | - | {on,off} | immediate | t |
| statement_timeout | 0 | 0 | 2147483647 | - | immediate | t |
| stats_temp_directory | 'pg_stat_tmp' | - | - | - | immediate | f |

| name | default value | min | max | allowedvalues | applymethod | ismodifiable |
|-------------------------------------|---------------|-------|------------|--|----------------|--------------|
| synchronize_seqscans | on | - | - | {on,off} | immediate | t |
| synchronous_commit | on | - | - | {local,on,off,remote_write,remote_apply} | immediate | t |
| syslog_facility | 'LOCAL0' | - | - | - | immediate | f |
| tcp_keepalives_count | 0 | 0 | 2147483647 | - | immediate | t |
| tcp_keepalives_idle | 0 | 0 | 2147483647 | - | immediate | t |
| tcp_keepalives_interval | 0 | 0 | 2147483647 | - | immediate | t |
| temp_buffers | 1024 | 0 | 1073741823 | - | immediate | t |
| temp_tablespaces | " | - | - | - | immediate | f |
| track_activities | on | - | - | {on,off} | immediate | t |
| timezone | UTC | - | - | - | immediate | t |
| track_functions | none | - | - | {none,pl,all} | immediate | t |
| track_io_timing | off | - | - | {on,off} | immediate | t |
| track_commit_timestamp | off | - | - | {on,off} | pending-reboot | t |
| transform_null_equals | off | - | - | {on,off} | immediate | t |
| array_nulls | on | - | - | {on,off} | immediate | t |
| authentication_timeout | 60 | 1 | 600 | - | immediate | t |
| autovacuum | on | - | - | {on,off} | immediate | t |
| autovacuum_analyze_scale_factor | 0.1 | 0 | 100 | - | immediate | t |
| autovacuum_analyze_threshold | 50 | 0 | 2147483647 | - | immediate | t |
| autovacuum_vacuum_cost_delay | 20 | -1 | 100 | - | immediate | t |
| autovacuum_naptime | 60 | 1 | 2147483 | - | immediate | t |
| autovacuum_multixact_freeze_max_age | 400000000 | 10000 | 2000000000 | - | pending-reboot | t |
| autovacuum_vacuum_cost_delay | 20 | -1 | 100 | - | immediate | t |
| autovacuum_vacuum_cost_limit | -1 | -1 | 10000 | - | immediate | t |
| autovacuum_vacuum_scale_factor | 0.2 | 0 | 100 | - | immediate | t |

| name | default value | min | max | allowedvalues | applymethod | ismodifiable |
|-----------------------------------|---------------------|-----|------------|------------------------|----------------|--------------|
| autovacuum_work_mem | -1 | -1 | 2147483647 | - | immediate | t |
| update_process_title | on | - | - | - | immediate | f |
| vacuum_cost_delay | 0 | 0 | 100 | - | immediate | t |
| vacuum_cost_limit | 200 | 1 | 10000 | - | immediate | t |
| vacuum_cost_page_directory | 20 | 0 | 10000 | - | immediate | t |
| vacuum_cost_page_hits | 1 | 0 | 10000 | - | immediate | t |
| vacuum_cost_page_miss | 10 | 0 | 10000 | - | immediate | t |
| vacuum_defer_cleanup_age | 0 | 0 | 1000000 | - | immediate | t |
| vacuum_freeze_table_age | 150000000 | 0 | 200000000 | - | immediate | t |
| vacuum_multixact_freeze_min_age | 5000000 | 0 | 10000000 | - | immediate | t |
| vacuum_multixact_freeze_table_age | 150000000 | 0 | 200000000 | - | immediate | t |
| wal_keep_segments | 64 | - | - | - | immediate | f |
| wal_receiver_status_interval | 10 | 0 | 2147483 | - | immediate | f |
| wal_sync_method | fdasync | - | - | - | immediate | f |
| wal_writer_delay | 200 | 1 | 10000 | - | immediate | t |
| wal_writer_flush_after | 128 | 0 | 2147483647 | - | immediate | t |
| work_mem | 1024 | 64 | 2147483647 | - | immediate | t |
| xmlbinary | 'base64' | - | - | {base64,hex} | immediate | t |
| xmloption | 'content' | - | - | {content,document} | immediate | t |
| datestyle | '''iso,mdy''' | - | - | - | immediate | t |
| search_path | '''\$user,public''' | - | - | - | immediate | t |
| log_timezone | UTC | - | - | - | immediate | f |
| row_security | on | - | - | {on,off} | immediate | t |
| huge_pages | 'try' | - | - | {try,on,off} | pending-reboot | t |
| dynamic_shared_memory_type | 'posix' | - | - | {posix,sysv,mmap,none} | pending-reboot | t |
| replacement_sort_tuples | 150000 | 0 | 2147483647 | - | immediate | t |
| max_worker_processes | 8 | 0 | 1000 | - | pending-reboot | t |

| name | default value | min | max | allowedvalues | applymethod | ismodifiable |
|-------------------------------------|---------------|-----|------------------------|------------------|----------------|--------------|
| s | | | | | | |
| max_parallel_workers_per_gather | 0 | 0 | 1024 | - | immediate | t |
| old_snapshot_threshold | -1 | -1 | 86400 | - | pending-reboot | t |
| backend_flush_after | 0 | 0 | 256 | - | immediate | t |
| wal_compression | off | - | - | {on,off} | immediate | t |
| wal_log_hints | off | - | - | - | pending-reboot | f |
| max_wal_size | 64 | 2 | 2147483647 | - | immediate | t |
| min_wal_size | 5 | 2 | 2147483647 | - | immediate | t |
| wal_sender_timeout | 60000 | 0 | 2147483647 | - | immediate | f |
| wal_receiver_timeout | 60000 | 0 | 2147483647 | - | immediate | f |
| wal_retrieve_retry_interval | 5000 | 1 | 2147483647 | - | immediate | f |
| max_replication_slots | 6 | 6 | 262143 | - | pending-reboot | t |
| parallel_tuple_cost | 0.1 | 0 | 1.797690000000001e+308 | - | immediate | t |
| parallel_setup_cost | 1000 | 0 | 1.797690000000001e+308 | - | immediate | t |
| min_parallel_relation_size | 1024 | 0 | 715827882 | - | immediate | t |
| force_parallel_mode | off | - | - | {on,off,regress} | immediate | t |
| syslog_sequence_numbers | on | - | - | - | immediate | f |
| syslog_split_messages | on | - | - | - | immediate | f |
| cluster_name | " | - | - | - | pending-reboot | f |
| lock_timeout | 0 | 0 | 2147483647 | - | immediate | t |
| idle_in_transaction_session_timeout | 0 | 0 | 2147483647 | - | immediate | t |
| gin_fuzzy_search_limit | 0 | 0 | 2147483647 | - | immediate | t |
| gin_pending_list_limit | 4096 | 64 | 2147483 | - | immediate | t |

| name | default value | min | max | allowedvalues | applymethod | ismodifiable |
|-----------------------------|---------------|-----|-----|---------------|-------------|--------------|
| | | | 647 | | | |
| session_preload_libraries | " | - | - | - | immediate | f |
| operator_precedence_warning | off | - | - | {on,off} | immediate | f |

(2) List of parameters that can be changed in Symfoware Server V12.1

| name | default value | min | max | allowedvalues | applymethod | ismodifiable |
|--------------------------------|-------------------------------------|-----|------------|---|----------------|--------------|
| archive_mode | on | - | - | {on,off} | pending-reboot | f |
| default_transaction_isolation | ""read committed"" | - | - | {serializable",repeatable read""",read committed""",read uncommitted""} | immediate | t |
| bonjour | off | - | - | - | pending-reboot | f |
| bonjour_name | "" | - | - | - | pending-reboot | f |
| krb_caseins_users | true | - | - | - | pending-reboot | f |
| default_text_search_config | 'pg_catalog.english' | - | - | - | pending-reboot | t |
| data_directory | "/userdata/data' | - | - | - | pending-reboot | f |
| listen_addresses | ""* | - | - | - | pending-reboot | f |
| dynamic_library_path | '\$libdir' | - | - | - | pending-reboot | f |
| ssl | off | - | - | {on,off} | pending-reboot | t |
| hot_standby | off | - | - | {on,off} | pending-reboot | f |
| hot_standby_feedback | off | - | - | {on,off} | pending-reboot | f |
| krb_server_keyfile | "" | - | - | - | pending-reboot | f |
| krb_srvname | "" | - | - | - | pending-reboot | f |
| local_preload_libraries | "" | - | - | - | pending-reboot | f |
| logging_collector | on | - | - | {on,off} | pending-reboot | f |
| max_connections | 100 | - | - | - | pending-reboot | t |
| max_files_per_process | 1,000 | 25 | - | - | pending-reboot | t |
| max_locks_per_transaction | 64 | 10 | 2147483647 | - | pending-reboot | t |
| max_pred_locks_per_transaction | 64 | 10 | 2147483647 | - | pending-reboot | t |
| max_prepared_transactions | 0 | - | - | - | pending-reboot | t |
| log_checkpoints | off | - | - | {on,off} | immediate | t |
| max_wal_senders | 12 | - | - | - | pending-reboot | f |
| replication_timeout | 60000 | - | - | - | pending-reboot | t |
| shared_buffers | 4096 | - | - | - | pending-reboot | t |
| shared_preload_libraries | "" | - | - | - | pending-reboot | f |
| ssl_ca_file | "" | - | - | - | pending-reboot | f |
| ssl_ciphers | 'ALL:!ADH:!LOW:!EXP:!MD5:@STRENGTH' | - | - | - | pending-reboot | f |
| ssl_crl_file | "" | - | - | - | pending-reboot | f |
| superuser_reserved_connections | 5 | - | - | - | pending-reboot | f |

| name | default value | min | max | allowedvalues | applymethod | ismodifiable |
|-----------------------------|---|-----------|------------|------------------------|----------------|--------------|
| synchronous_standby_names | " | - | - | - | pending-reboot | f |
| syslog_ident | 'postgres' | - | - | - | pending-reboot | f |
| tablespace_path_prefix | '/userdata/tblspc' | - | - | - | pending-reboot | f |
| enable_indexonlyscan | on | - | - | {on,off} | pending-reboot | t |
| temp_file_limit | -1 | - | - | - | pending-reboot | t |
| timezone_abbreviations | 'Default' | - | - | - | pending-reboot | f |
| ssl_key_file | '/opt/dbaas/ssl_keys/server.key' | - | - | - | pending-reboot | f |
| max_standby_archive_delay | 30000 | -1 | 2147483647 | - | immediate | t |
| track_activity_query_size | 1024 | 100 | 102400 | - | pending-reboot | t |
| unix_socket_directory | " | - | - | - | pending-reboot | f |
| unix_socket_group | " | - | - | - | pending-reboot | f |
| hba_file | '/userdata/data/pg_hba.conf' | - | - | - | pending-reboot | f |
| ident_file | '/userdata/data/pg_ident.conf' | - | - | - | pending-reboot | f |
| autovacuum_freeze_max_age | 200000000 | 100000000 | 200000000 | - | pending-reboot | t |
| autovacuum_max_workers | 3 | 1 | 8388607 | - | pending-reboot | t |
| unix_socket_permissions | 0777 | - | - | - | pending-reboot | f |
| wal_buffers | -1 | - | - | - | pending-reboot | t |
| wal_level | hot_standby | - | - | - | pending-reboot | f |
| external_pid_file | " | - | - | - | pending-reboot | f |
| track_counts | on | - | - | {on,off} | immediate | t |
| archive_command | "/opt/dbaas/client_config/local_backup_script.sh %p %f" | - | - | - | pending-reboot | f |
| keystore_location | '/userdata/data' | - | - | - | pending-reboot | f |
| ssl_cert_file | '/opt/dbaas/ssl_keys/server.crt' | - | - | - | pending-reboot | f |
| archive_timeout | 300 | - | - | - | pending-reboot | f |
| autovacuum_vacuum_threshold | 50 | 0 | 2147483647 | - | immediate | t |
| backslash_quote | safe_encoding | - | - | {safe_encoding,on,off} | immediate | t |
| bgwriter_delay | 200 | 10 | 10000 | - | immediate | t |
| bgwriter_lru_maxpages | 100 | 0 | 1,000 | - | immediate | t |
| bgwriter_lru_multiplier | 2.0 | 0 | 10 | - | immediate | t |
| bytea_output | 'hex' | - | - | {escape,hex} | immediate | t |
| check_function_bodies | on | - | - | {on,off} | immediate | t |

| name | default value | min | max | allowedvalues | applymethod | ismodifiable |
|--------------------------------|---------------|-----|------------------------|---|-------------|--------------|
| vacuum_freeze_min_age | 50000000 | 0 | 100000000 | - | immediate | t |
| checkpoint_completion_target | 0.5 | 0 | 1 | - | immediate | t |
| checkpoint_segments | 20 | - | - | - | immediate | t |
| checkpoint_timeout | 3600 | 30 | 3600 | - | immediate | t |
| checkpoint_warning | 30 | - | - | - | immediate | t |
| client_encoding | sql_ascii | - | - | - | immediate | t |
| client_min_messages | notice | - | - | {debug5,debug4,debug3,debug2,debug1,log,notice,warning,error} | immediate | t |
| commit_delay | 0 | 0 | 100000 | - | immediate | t |
| commit_siblings | 5 | 0 | 1,000 | - | immediate | t |
| constraint_exclusion | partition | - | - | {partition,on,off} | immediate | t |
| cpu_index_tuple_cost | 0.005 | 0 | 1.797690000000001e+308 | - | immediate | t |
| cpu_operator_cost | 0.0025 | 0 | 1.797690000000001e+308 | - | immediate | t |
| cpu_tuple_cost | 0.01 | 0 | 1.797690000000001e+308 | - | immediate | t |
| cursor_tuple_fraction | 0.1 | 0 | 1 | - | immediate | t |
| db_user_namespace | off | - | - | {on,off} | immediate | f |
| deadlock_timeout | 1,000 | 1 | 2147483647 | - | immediate | t |
| debug_pretty_print | on | - | - | {on,off} | immediate | t |
| debug_print_parse | off | - | - | {on,off} | immediate | t |
| debug_print_plan | off | - | - | {on,off} | immediate | t |
| debug_print_rewritten | off | - | - | {on,off} | immediate | t |
| default_statistics_target | 100 | 1 | 10000 | - | immediate | t |
| default_tablespace | " | - | - | - | immediate | f |
| default_transaction_deferrable | off | - | - | {on,off} | immediate | t |
| full_page_writes | on | - | - | {on,off} | immediate | t |
| default_transaction_read_only | off | - | - | {on,off} | immediate | t |

| name | default value | min | max | allowedvalues | applymethod | ismodifiable |
|--------------------------|---------------|-----|----------------|---|-------------|--------------|
| y | | | | | | |
| default_with_oids | off | - | - | {on,off} | immediate | t |
| effective_cache_size | 16384 | 1 | 2147483 647 | - | immediate | t |
| effective_io_concurrency | 1 | 0 | 1,000 | - | immediate | t |
| enable_bitmapscan | on | - | - | {on,off} | immediate | t |
| enable_hashagg | on | - | - | {on,off} | immediate | t |
| enable_hashjoin | on | - | - | {on,off} | immediate | t |
| enable_indexscan | on | - | - | {on,off} | immediate | t |
| enable_material | on | - | - | {on,off} | immediate | t |
| enable_mergejoin | on | - | - | {on,off} | immediate | t |
| enable_nestloop | on | - | - | {on,off} | immediate | t |
| enable_seqscan | on | - | - | {on,off} | immediate | t |
| enable_sort | on | - | - | {on,off} | immediate | t |
| enable_tidscan | on | - | - | {on,off} | immediate | t |
| escape_string_warning | on | - | - | {on,off} | immediate | t |
| exit_on_error | off | - | - | {on,off} | immediate | f |
| extra_float_digits | 0 | -15 | -3 | - | immediate | t |
| from_collapse_limit | 8 | 1 | 2147483 647 | - | immediate | t |
| fsync | on | - | - | {on,off} | immediate | t |
| geqo_effort | 5 | 1 | 10 | - | immediate | t |
| geqo_generations | 0 | 0 | 2147483 647 | - | immediate | t |
| geqo_pool_size | 0 | 0 | 2147483 647 | - | immediate | t |
| geqo_seed | 0.0 | 0 | 1 | - | immediate | t |
| geqo_selection_bias | 2.0 | 1.5 | 2 | - | immediate | t |
| geqo_threshold | 12 | 2 | 2147483 647 | - | immediate | t |
| intervalstyle | 'postgres' | - | - | {postgres,postgres_verbose,sql_standard,iso_8601} | immediate | t |
| join_collapse_limit | 8 | 1 | 2147483 647 | - | immediate | t |
| lc_messages | 'en_US.UTF-8' | - | - | - | immediate | t |
| lc_monetary | 'en_US.UTF-8' | - | - | - | immediate | t |
| lc_numeric | 'en_US.UTF-8' | - | - | - | immediate | t |
| lc_time | 'en_US.UTF-8' | - | - | - | immediate | t |

| name | default value | min | max | allowedvalues | applymethod | ismodifiable |
|-----------------------------|----------------------------------|-----|------------|--|-------------|--------------|
| geqo | on | - | - | {on,off} | immediate | t |
| lo_compat_privileges | off | - | - | {on,off} | immediate | f |
| log_autovacuum_min_duration | -1 | -1 | 2147483647 | - | immediate | t |
| log_connections | off | - | - | {on,off} | immediate | t |
| log_destination | 'stderr' | - | - | - | immediate | f |
| log_directory | 'pg_log' | - | - | - | immediate | f |
| log_disconnections | off | - | - | {on,off} | immediate | t |
| log_duration | off | - | - | {on,off} | immediate | t |
| log_error_verbosity | default | - | - | {terse,default,verbose} | immediate | t |
| log_executor_stats | off | - | - | {on,off} | immediate | t |
| log_file_mode | 0600 | - | - | - | immediate | f |
| log_filename | 'postgresql.log-%Y-%m-%d_%H%M%S' | - | - | {postgresql.log.%Y-%m-%d,postgresql.log.%Y-%m-%d-%H} | immediate | t |
| log_hostname | off | - | - | {on,off} | immediate | t |
| log_line_prefix | " | - | - | - | immediate | t |
| log_lock_waits | off | - | - | {on,off} | immediate | t |
| log_min_duration_statement | -1 | -1 | 2147483647 | - | immediate | t |
| log_min_error_statement | error | - | - | {debug5,debug4,debug3,debug2,debug1,info,notice,warning,error,log,fatal,panic} | immediate | t |
| log_min_messages | warning | - | - | {debug5,debug4,debug3,debug2,debug1,info,notice,warning,error,log,fatal,panic} | immediate | t |
| log_planner_stats | off | - | - | {on,off} | immediate | t |
| log_rotation_age | 1440 | 1 | 1440 | - | immediate | t |
| log_rotation_size | 10240 | - | - | - | immediate | t |
| log_statement | 'none' | - | - | {none,ddl,mod,all} | immediate | t |
| log_statement_stats | off | - | - | {on,off} | immediate | t |
| log_temp_files | -1 | -1 | 2147483647 | - | immediate | t |
| log_truncate_on_rotation | on | - | - | {on,off} | immediate | f |

| name | default value | min | max | allowedvalues | applymethod | ismodifiable |
|---------------------------------|---------------|-----|------------------------|-----------------------------|-------------|--------------|
| maintenance_work_mem | 16384 | - | - | - | immediate | t |
| log_parser_stats | off | - | - | {on,off} | immediate | t |
| max_stack_depth | 2048 | - | - | - | immediate | t |
| max_standby_streaming_delay | 30000 | -1 | 2147483647 | - | immediate | t |
| password_encryption | on | - | - | {on,off} | immediate | f |
| quote_all_identifiers | off | - | - | {on,off} | immediate | t |
| random_page_cost | 4.0 | 0 | 1.797690000000001e+308 | - | immediate | t |
| restart_after_crash | on | - | - | {on,off} | immediate | f |
| seq_page_cost | 1.0 | - | - | - | immediate | t |
| session_replication_role | 'origin' | - | - | {origin,replica,local} | immediate | t |
| sql_inheritance | on | - | - | {on,off} | immediate | t |
| ssl_renegotiation_limit | 0 | - | - | - | immediate | f |
| standard_conforming_strings | on | - | - | {on,off} | immediate | t |
| statement_timeout | 0 | 0 | 2147483647 | - | immediate | t |
| stats_temp_directory | 'pg_stat_tmp' | - | - | - | immediate | f |
| synchronize_seqscans | on | - | - | {on,off} | immediate | t |
| synchronous_commit | on | - | - | {local,on,off,remote_write} | immediate | t |
| syslog_facility | 'LOCAL0' | - | - | - | immediate | f |
| tcp_keepalives_count | 0 | - | - | - | immediate | t |
| tcp_keepalives_idle | 0 | - | - | - | immediate | t |
| tcp_keepalives_interval | 0 | - | - | - | immediate | t |
| temp_buffers | 1024 | - | - | - | immediate | t |
| temp_tablespace | " | - | - | - | immediate | f |
| track_activities | on | - | - | {on,off} | immediate | t |
| timezone | UTC | - | - | - | immediate | t |
| track_functions | none | - | - | {none,pl,all} | immediate | t |
| track_io_timing | off | - | - | {on,off} | immediate | t |
| transform_null_equals | off | - | - | {on,off} | immediate | t |
| array_nulls | on | - | - | {on,off} | immediate | t |
| authentication_timeout | 60 | 1 | 600 | - | immediate | t |
| autovacuum | on | - | - | {on,off} | immediate | t |
| autovacuum_analyze_scale_factor | 0.1 | 0 | 100 | - | immediate | t |

| name | default value | min | max | allowedvalues | applymethod | ismodifiable |
|------------------------------------|-------------------|-----|----------------|--------------------|-------------|--------------|
| autovacuum_analyze_thresh old | 50 | 0 | 2147483 647 | - | immediate | t |
| autovacuum_naptime | 60 | 1 | 2147483 | - | immediate | t |
| autovacuum_vacuum_cost_d elay | 20 | -1 | 100 | - | immediate | t |
| autovacuum_vacuum_cost_li mit | -1 | -1 | 10000 | - | immediate | t |
| autovacuum_vacuum_scale_f actor | 0.2 | 0 | 100 | - | immediate | t |
| update_process_title | on | - | - | {on,off} | immediate | t |
| vacuum_cost_delay | 0 | 0 | 100 | - | immediate | t |
| vacuum_cost_limit | 200 | 1 | 10000 | - | immediate | t |
| vacuum_cost_page_dirty | 20 | 0 | 10000 | - | immediate | t |
| vacuum_cost_page_hit | 1 | 0 | 10000 | - | immediate | t |
| vacuum_cost_page_miss | 10 | 0 | 10000 | - | immediate | t |
| vacuum_defer_cleanup_age | 0 | - | - | - | immediate | t |
| vacuum_freeze_table_age | 150000000 | 0 | 2000000 000 | - | immediate | t |
| wal_keep_segments | 64 | - | - | - | immediate | f |
| wal_receiver_status_interval | 10 | 0 | 2147483 | - | immediate | f |
| wal_sync_method | fdatasync | - | - | - | immediate | f |
| wal_writer_delay | 200 | 1 | 10000 | - | immediate | t |
| work_mem | 1024 | - | - | - | immediate | t |
| xmlbinary | 'base64' | - | - | {base64,hex} | immediate | t |
| xmloption | 'content' | - | - | {content,document} | immediate | t |
| datestyle | ""iso,mdy"" | - | - | - | immediate | t |
| search_path | ""\$user,public"" | - | - | - | immediate | t |
| log_timezone | UTC | - | - | - | immediate | f |

A.2 Estimating Memory Usage

Estimate the amount of memory used by a DB instance using the following formula.

$$\text{Memory Used by DB Instance} = \text{Base Memory Usage} + \text{DB Engine Memory Usage}$$

(1) Base Memory Usage

Base memory usages are as follows:

For a single configuration: 300 MB

For a redundant configuration: 500 MB

(2) DB Engine Memory Usage

Estimate the amount of memory used by a DB engine using the following formula.

$$\text{DB Engine Memory Usage} = \text{Amount of Shared Memory} + \text{Amount of Local Memory}$$

For the amount of shared memory, refer to the following:

<For Enterprise Postgres 9.6>

“18.4.1 Shared Memory and Semaphores” in “18.4 Managing Kernel Resources” in “III. Server Administration” in the “PostgreSQL 9.6.2 Documentation”

<For Symfoware Server 12.1>

“Table 17-2. PostgreSQL Shared Memory Usage” in “17.4.1 Shared Memory and Semaphores” in “18.4 Managing Kernel Resources” in “III. Server Administration” in the “PostgreSQL 9.2.4 Documentation”

For the amount of local memory, refer to the following:

<For Enterprise Postgres 9.6>

“Local memory amount” in “H.1 Estimating Memory Requirements in FUJITSU Enterprise Postgres” in “Appendix H Estimating Memory Requirements” in the “FUJITSU Enterprise Postgres 9.6 Installation and Setup Guide for Server”.

<For Symfoware Server 12.1>

“Local memory amount” in “J.1 Estimating Memory Usage in Symfoware Server” in “Appendix J Estimating Memory Requirements” in the “Symfoware Server Installation and Setup Guide for Server”.

Appendix B: Supported Client

The IaaS Database Service supports the versions of DB client shown below.

To get DB client, download from the following.

<https://doc.cloud.global.fujitsu.com/jp/iaas/index.html>

| DB instance engine | Supported DB client |
|-------------------------|--------------------------------|
| Enterprise Postgres 9.6 | Enterprise Postgres Client 9.6 |
| Symfoware Server V12.1 | Symfoware Server Client V12.1 |

For details about the OS supported by each DB client, refer to "FUJITSU Enterprise Postgres 9.6 Installation and Setup Guide for Client", or the "Symfoware Server Installation and Setup Guide for Client".

To create a virtual server (DB client) with the IaaS and install a DB client, you can use the OS listed below.

* This information is current as of November 2017.

| OS | Enterprise Postgres Client 9.6 | Symfoware Server Client V12.1 | Remarks |
|---|--------------------------------|-------------------------------|--------------------------------------|
| Windows Server 2008 Standard Edition R2 64-bit Japanese Version | ✓ | ✓ | Only Eastern Japan and Western Japan |
| Windows Server 2008 Enterprise Edition R2 64-bit Japanese Version | ✓ | ✓ | |
| Windows Server 2012 Standard Edition 64-bit Japanese Version | ✓ | ✓ | |
| Windows Server 2012 R2 Standard Edition 64-bit Japanese Version | ✓ | ✓ | |
| CentOS 6.8 64bit (English) | ✓ | ✓ | |
| CentOS 7.2 64bit (English) | ✓ | ✓ | |
| CentOS 7.3 64bit (English) | ✓ | ✓ | |
| Red Hat Enterprise Linux 6.8 64bit (English) | ✓ | ✓ | |
| Red Hat Enterprise Linux 7.2 64bit (English) | ✓ | ✓ | |
| Red Hat Enterprise Linux 7.3 64bit (English) | ✓ | ✓ | |

Appendix C: Flavor ID List

You can check a list of flavors provided by the IaaS Database Service by using the flavor list reference API.

When creating a DB instance, specify any of the flavor IDs obtained with the flavor list reference API and change it if necessary.

| Command Example |
|--|
| <pre>ENDPOINT=[Endpoint of the database service] LIMIT=[The number of flavors to be displayed (Specify the desired value from 20 to 100.)] curl -X GET -i \${ENDPOINT}/v1.0/\${TENANTID}/flavors?limit=\${LIMIT} -H "X-Auth-Token: \${TOKEN}"</pre> |
| Example of Execution Results |
| <pre>\$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ LIMIT=100 \$ curl -X GET -i \${ENDPOINT}/v1.0/\${TENANTID}/flavors?limit=\${LIMIT} -H "X-Auth-Token: \${TOKEN}" HTTP/1.1 200 OK (Omitted) {"flavors":[{"id":"XXXXXXXX", (Omitted) }, {"id":"1101", "links": [{"href":"https://XXXXXXXX/flavors/1101", "rel":"SELF"}, {"href":"https://XXXXXXXX/flavors/1101", "rel":"BOOKMARK"}]}, {"name":"S-1"}, {"id":"1102", "links": [{"href":"https://XXXXXXXX/flavors/1102", "rel":"SELF"}, {"href":"https://XXXXXXXX/flavors/1102", "rel":"BOOKMARK"}]}, {"name":"S-2"}, (Omitted)]}]</pre> <p>*For "name" of "flavors", the virtual server type† is output. Select an appropriate one and use the corresponding id.</p> |

† To check the virtual server types (flavors) provided by the IaaS, refer to "IaaS Service Description" in the Service Descriptions category at <http://www.fujitsu.com/global/solutions/cloud/k5/terms/>.

D.1 Transparent Data Encryption Function

This section describes encrypting the data to be stored in the database.

Using the transparent data encryption function in Enterprise Postgres or Symfoware Server allows you to encrypt the data in the OS file.

Therefore, this function protects your valuable information even if the file is stolen through unauthorized access.

The transparent data encryption function in Enterprise Postgres or Symfoware Server is different from the encryption function in PostgreSQL in the following ways:

- Correcting the application (SQL statement) is not required.
- The size of the data is not changed after encryption.
- Changing the column data types is not required.

The data stored in the database is encrypted when written to a file and decrypted when read.

This operation is automatically performed by the instance, thus enabling users and applications to perform key management and encryption/decryption processes automatically.

[Reference]

The initial settings for the master encryption key and the automatic open for the keystore are automatically set during the creation of the DB.

With the IaaS Database Service, the settings for the master encryption key and automatic open for the keystore are not required.

For details about transparent data encryption, refer to the Enterprise Postgres manuals or the Symfoware Server manuals.

D.2 Encryption Range

This section describes the range of encryption.

| Encrypted Data | Description |
|--|--|
| All user data in the specified tablespace | <ul style="list-style-type: none">- Encryption is set on a tablespace basis.- All the tables, indexes, temporary tables, and temporary indexes created in the encryption tablespace are encrypted. |
| Backup data | <ul style="list-style-type: none">- Backup data created with the database service function remains encrypted. |
| WAL and temporary file | <ul style="list-style-type: none">- WAL that is created when updating the encrypted table and index is also encrypted.- When a large-scale merge or sort is performed, the encrypted data is kept encrypted when it is written to a temporary file. |
| Data synchronization of redundant DB instances | <ul style="list-style-type: none">- The data (table and index stored in the encrypted tablespace) and WAL that are encrypted with the primary server are kept encrypted when it is transferred and then stored in the standby server. |

The following files are not encrypted:

- Output file of the `pg_dump` and `pg_dumpall` commands
- COPY command output*
- Payload of notification events communicated with the LISTEN/NOTIFY command

* Database Import and Export

Output results of the COPY TO command are not encrypted.

When the import destination of the COPY FROM command is in an encryption tablespace, the imported data is automatically encrypted and stored.

D.3 Encrypting a Tablespace

The operations described in this section are performed by the database administrator.

This section describes the procedure to encrypt a tablespace.

When creating a tablespace to be encrypted, specify the encryption algorithm in the parameter during execution.

For the encryption algorithm, you can use AES with a key length of 128 or 256 bits. * 256-bit AES is recommended.

Valid values are AES128, AES256, or none (default value). Specifying "none" disables encryption.

You cannot encrypt the `pg_default` and `pg_global` tablespaces.

| Command Example |
|---|
| <pre>CLIENT_DIR=[<i>Directory path where the DB engine client is installed</i>] export PATH=\${CLIENT_DIR}/bin:\${PATH} export LD_LIBRARY_PATH=\${CLIENT_DIR}/lib:\${LD_LIBRARY_PATH} FQDN=[<i>Address in the FQDN confirmed in 2.3 (1)</i>] PORT=[<i>Port confirmed in 2.3 (1)</i>] MASTERUSERNAME=[<i>Database administrator name confirmed in 2.3 (1)</i>] ALGORITHM=[<i>Encryption algorithm (Specify AES128 or AES256.)</i>] SECURE_TABLESPACE=[<i>Name of an encryption tablespace to be created (a desired value.)</i>] SECURE_DATA=[<i>Identification name (a desired value) to store tablespace (Create a character string of up to 82 bytes consisting of alphanumeric characters and symbols, and add a slash (/) at the beginning of the string.)</i>] # SQL to specify the encryption algorithm for the tablespace to be created later echo "SET tablespace_encryption_algorithm = '\${ALGORITHM};'" > /tmp/temp.sql echo "CREATE TABLESPACE \${SECURE_TABLESPACE} LOCATION '\${SECURE_DATA};'" >> /tmp/temp.sql # SQL to disable encryption for the tablespace to be created later</pre> |

```

echo "SET tablespace_encryption_algorithm = 'none';" >> /tmp/temp.sql

# Checking the encrypted tablespace

SQL_STR="SELECT spcname, spcencalgo FROM pg_tablespace ts, pgx_tablespaces tsx WHERE ts.oid = tsx.spctablespace;"

psql -p ${PORT} -U ${MASTERUSERNAME} -h ${FQDN} -d postgres -f /tmp/temp.sql

psql -p ${PORT} -U ${MASTERUSERNAME} -h ${FQDN} -d postgres -c "${SQL_STR}"

rm /tmp/temp.sql

```

Example of Execution Results

```

$ CLIENT_DIR="/XXXXXXXX/XXXXXXXX"
$ export PATH=${CLIENT_DIR}/bin:${PATH}
$ export LD_LIBRARY_PATH=${CLIENT_DIR}/lib:${LD_LIBRARY_PATH}
$ FQDN="XXXXXXXX.XXX.XXX"
$ PORT="26500"
$ MASTERUSERNAME="masterusername"
$ ALGORITHM="AES256"
$ SECURE_TABLESPACE="secure_tablespace"
$ SECURE_DATA="/SecureData"

$ echo "SET tablespace_encryption_algorithm = '${ALGORITHM}';" > /tmp/temp.sql
$ echo "CREATE TABLESPACE ${SECURE_TABLESPACE} LOCATION '${SECURE_DATA}';" >> /tmp/temp.sql
$ echo "SET tablespace_encryption_algorithm = 'none';" >> /tmp/temp.sql
$ SQL_STR="SELECT spcname, spcencalgo FROM pg_tablespace ts, pgx_tablespaces tsx WHERE ts.oid = tsx.spctablespace;"

$ psql -p ${PORT} -U ${MASTERUSERNAME} -h ${FQDN} -d postgres -f /tmp/temp.sql
SET
CREATE TABLESPACE
SET
$ psql -p ${PORT} -U ${MASTERUSERNAME} -h ${FQDN} -d postgres -c "${SQL_STR}"

```

| spcname | spcencalgo |
|-------------------|------------|
| pg_default | none |
| pg_global | none |
| secure_tablespace | AES256 |

*Confirm that [spcencalgo] of the created tablespace is AES256.

```

(3 rows)
$ rm /tmp/temp.sql

```

Tables and indexes created in the above-mentioned encryption tablespace are automatically encrypted.

D.4 Changing Master Encryption Key

The operations described in this section are performed by the database administrator.

This section describes the procedure to change the master encryption key.

Using the same encryption key for a long time may give attackers the opportunity to decrypt it. We recommend that you change the master encryption key periodically.

Set the master encryption key by executing the SQL function as follows.

| Command Example |
|---|
| <pre>CLIENT_DIR=[<i>Directory path where the DB engine client is installed</i>] export PATH=\${CLIENT_DIR}/bin:\${PATH} export LD_LIBRARY_PATH=\${CLIENT_DIR}/lib:\${LD_LIBRARY_PATH} FQDN=[<i>Address in the FQDN confirmed in 2.3 (1)</i>] PORT=[<i>Port confirmed in 2.3 (1)</i>] MASTERUSERNAME=[<i>Database administrator name confirmed in 2.3 (1)</i>] PASSPHRASE=[<i>Master encryption key passphrase *</i>] SQL_STR="SELECT pgx_set_master_key('\${PASSPHRASE}');" *2 psql -p \${PORT} -U \${MASTERUSERNAME} -h \${FQDN} -d postgres -c "\${SQL_STR}"</pre> |
| Example of Execution Results |
| <pre>\$ CLIENT_DIR="/XXXXXXXX/XXXXXXXX" \$ export PATH=\${CLIENT_DIR}/bin:\${PATH} \$ export LD_LIBRARY_PATH=\${CLIENT_DIR}/lib:\${LD_LIBRARY_PATH} \$ FQDN="XXXXXXXX.XXX.XXX" \$ PORT="26500" \$ MASTERUSERNAME="masterusername" \$ PASSPHRASE="XXXXXXXX" \$ SQL_STR="SELECT pgx_set_master_key('\${PASSPHRASE}');" *2 \$ psql -p \${PORT} -U \${MASTERUSERNAME} -h \${FQDN} -d postgres -c "\${SQL_STR}" pgx_set_master_key ----- (1 row) *Confirm that a result similar to the one shown on the left is returned.</pre> |

*1 The administrator password that is specified during the creation of the DB instance is set as the default passphrase for the master encryption key. The default passphrase is a character string of 8 to 200 bytes.

[Note]

- If the administrator password that is specified during the DB instance creation is shorter than 8 bytes, then the initial passphrase is set as the administrator password with zeros added to the right side of the field of the passphrase until it reaches a length of 8 bytes.

Example:

| Administrator Password | Initial Passphrase |
|------------------------|--------------------|
| pass | pass0000 |

- If the administrator password is longer than 200 bytes, the initial passphrase is set as the administrator password truncated to a length of 200 bytes.

*2 The `pgx_set_master_key` function cannot be executed in a transaction block.

D.5 Changing the Keystore Passphrase

The operations described in this section are performed by the database administrator.

This section describes the procedure to change the keystore passphrase.

Change the passphrase by executing the SQL function as follows.

| Command Example |
|---|
| <pre>CLIENT_DIR=[<i>Directory path where the DB engine client is installed</i>] export PATH=\${CLIENT_DIR}/bin:\${PATH} export LD_LIBRARY_PATH=\${CLIENT_DIR}/lib:\${LD_LIBRARY_PATH} FQDN=[<i>Address in the FQDN confirmed in 2.3 (1)</i>] PORT=[<i>Port confirmed in 2.3 (1)</i>] MASTERUSERNAME=[<i>Database administrator name confirmed in 2.3 (1)</i>] OLD_PASSPHRASE=[<i>Master encryption key passphrase *1</i>] NEW_PASSPHRASE=[<i>New master encryption key passphrase (A character string of 8 to 200 bytes consisting of alphanumeric characters and symbols) (A desired value)</i>] SQL_STR="SELECT pgx_set_keystore_passphrase('\${OLD_PASSPHRASE}', '\${NEW_PASSPHRASE}');" *2 psql -p \${PORT} -U \${MASTERUSERNAME} -h \${FQDN} -d postgres -c "\${SQL_STR}"</pre> |
| Example of Execution Results |
| <pre>\$ CLIENT_DIR="/XXXXXXXX/XXXXXXXX" \$ export PATH=\${CLIENT_DIR}/bin:\${PATH} \$ export LD_LIBRARY_PATH=\${CLIENT_DIR}/lib:\${LD_LIBRARY_PATH} \$ FQDN="XXXXXXXX.XXX.XXX" \$ PORT="26500" \$ MASTERUSERNAME="masterusername" \$ OLD_PASSPHRASE="XXXXXXXX" \$ NEW_PASSPHRASE="YYYYYYYY"</pre> |

```

$ SQL_STR="SELECT pgx_set_keystore_passphrase('${OLD_PASSPHRASE}', '${NEW_PASSPHRASE}');"
$ psql -p ${PORT} -U ${MASTERUSERNAME} -h ${FQDN} -d postgres -c "${SQL_STR}"
pgx_set_keystore_passphrase
-----
(1 row)                *Confirm that a result similar to the one shown on the left is returned.

```

*1 The administrator password specified during the creation of the DB instance is set as the default passphrase for the master encryption key. The default passphrase is a character string of 8 to 200 bytes.

[Note]

- If the administrator password that is specified during the DB instance creation is shorter than 8 bytes, then the initial passphrase is set as the administrator password with zeros added to the right side of the field of the passphrase until it reaches a length of 8 bytes.

Example:

| Administrator Password | Initial Passphrase |
|------------------------|--------------------|
| pass | pass0000 |

- If the administrator password is longer than 200 bytes, the initial passphrase is set as the administrator password truncated to a length of 200 bytes.

*2 The `pgx_set_master_key` function cannot be executed in a transaction block.

[Reference]

If you forget the passphrase specified here, you cannot change the master encryption key and keystore passphrase. There is no way to retrieve a forgotten passphrase or decrypt the data. Be careful not to forget your passphrase.

D.6 Backing Up and Restoring/Recovering Database

When a DB instance is backed up or restored, the master encryption key and automatic open key used for the transparent data encryption function are also backed up or restored automatically.

No setting procedures are required.

D.7 Introducing Pre-Configured Application

Transparent data encryption facilitates encryption of all application data without changing the application.

Set the encryption tablespace as the default tablespace for a user with application owner privileges for the pre-configured application.

If the default tablespace of the user who executes the SQL script of the application is set to the encryption tablespace, the object created with the SQL script is located in the encryption tablespace.

An example of a command to set the encryption tablespace created in section D.3 is shown below.

| Command Example |
|--|
| <pre>CLIENT_DIR=[<i>Directory path where the DB engine client is installed</i>] export PATH=\${CLIENT_DIR}/bin:\${PATH} export LD_LIBRARY_PATH=\${CLIENT_DIR}/lib:\${LD_LIBRARY_PATH} FQDN=[<i>Address in the FQDN confirmed in 2.3 (1)</i>] PORT=[<i>Port confirmed in 2.3 (1)</i>] MASTERUSERNAME=[<i>Database administrator name confirmed in 2.3 (1)</i>] APP_USER=[<i>User with application owner privileges for a pre-configured application</i>] SECURE_TABLESPACE=[<i>Name of the encryption tablespace created in D.3</i>] SQL_STR1="ALTER USER \${APP_USER} SET default_tablespace = '\${SECURE_TABLESPACE}';" SQL_STR2="ALTER USER \${APP_USER} SET temp_tablespaces = '\${SECURE_TABLESPACE}';" psql -p \${PORT} -U \${MASTERUSERNAME} -h \${FQDN} -d postgres -c "\${SQL_STR1}" psql -p \${PORT} -U \${MASTERUSERNAME} -h \${FQDN} -d postgres -c "\${SQL_STR2}"</pre> |
| Example of Execution Results |
| <pre>\$ CLIENT_DIR="/XXXXXXXX/XXXXXXXX" \$ export PATH=\${CLIENT_DIR}/bin:\${PATH} \$ export LD_LIBRARY_PATH=\${CLIENT_DIR}/lib:\${LD_LIBRARY_PATH} \$ FQDN="XXXXXXXX.XXX.XXX" \$ PORT="26500" \$ MASTERUSERNAME="masterusername" \$ APP_USER="appuser" \$ SECURE_TABLESPACE="secure_tablespace" \$ SQL_STR1="ALTER USER \${APP_USER} SET default_tablespace = '\${SECURE_TABLESPACE}';" \$ SQL_STR2="ALTER USER \${APP_USER} SET temp_tablespaces = '\${SECURE_TABLESPACE}';" \$ psql -p \${PORT} -U \${MASTERUSERNAME} -h \${FQDN} -d postgres -c "\${SQL_STR1}" ALTER ROLE \$ psql -p \${PORT} -U \${MASTERUSERNAME} -h \${FQDN} -d postgres -c "\${SQL_STR2}" ALTER ROLE</pre> |

D.8 Other Points to Note

- Points to note on security

Because DB logs are not encrypted, if a constant is set to an SQL statement, its value may be output to the DB logs.

When SQL for "master encryption key change" or "keystore passphrase change" is executed, an SQL statement including the passphrase may be output to the DB logs.

In order to prevent these problems, consider setting a parameter such as "log_min_error_statement". We recommend setting the level to the default value of "ERROR" or higher. In this case, SQL in a normal or warning status is not output to the DB logs and the constant in the SQL is not output either.

- Actions to Take When an Error Occurs

If an error message is returned after a command described in Appendix D is executed, and if any of the error messages shown below are displayed, check the executed SQL.

| Message No. | Message Details |
|-------------|---|
| 15204 | passphrase is too short or too long |
| 15207 | passphrase of the keystore has been changed |
| 15214 | could not open keystore "@!@": passphrase is wrong, or the auto-open keystore was created by another computer or user |

For other messages, refer to "Messages" in the Enterprise Postgres manuals or the Symfoware Server manuals.

If a message with an error type (ERROR, FATAL, PANIC) is displayed and it is determined that the error cannot be solved with SQL, consider recovering the DB instance following the steps described in Chapter 5.

If the problem still remains after the above-mentioned actions are taken, contact the service provider for assistance.

Appendix E: Roles That Can Execute Each API

The table below shows the roles that can execute each API of the IaaS Database Service.

| API | Administrator | System Owner | Operator | Observer |
|--|---------------|--------------|----------|----------|
| Creating a DB Instance | ✓ | ✓ | - | - |
| Deleting a DB Instance | ✓ | ✓ | ✓ | - |
| Changing a DB Instance | ✓ | ✓ | ✓ | - |
| Recovering a DB Instance from a DB Snapshot | ✓ | ✓ | ✓ | - |
| Recovering a DB Instance with Point-in-Time Recovery | ✓ | ✓ | ✓ | - |
| Starting a DB Instance | ✓ | ✓ | ✓ | - |
| Stopping a DB Instance | ✓ | ✓ | ✓ | - |
| Restarting a DB Instance | ✓ | ✓ | ✓ | - |
| Creating a Reference Replica DB Instance | ✓ | ✓ | - | - |
| Referring to the DB Instance List | ✓ | ✓ | ✓ | ✓ |
| Checking the DB Instance Information | ✓ | ✓ | ✓ | ✓ |
| Cancelling a DB Instance Operation | ✓ | ✓ | ✓ | - |
| Creating a DB Snapshot | ✓ | ✓ | ✓ | - |
| Deleting a DB Snapshot | ✓ | ✓ | ✓ | - |
| Duplicating a DB Snapshot | ✓ | ✓ | - | - |
| Referring to a DB Snapshot List | ✓ | ✓ | ✓ | ✓ |
| Checking the DB Snapshot Information | ✓ | ✓ | ✓ | ✓ |
| Referring to a DB Log File List | ✓ | ✓ | ✓ | ✓ |
| Referring to a DB Log File | ✓ | ✓ | ✓ | ✓ |
| Creating a DB Subnet Group | ✓ | ✓ | - | - |
| Deleting a DB Subnet Group | ✓ | ✓ | - | - |
| Modifying a DB Subnet Group | ✓ | ✓ | - | - |
| Referring to the DB Subnet Group List | ✓ | ✓ | ✓ | ✓ |
| Checking the DB Subnet Group Information | ✓ | ✓ | ✓ | ✓ |
| Creating a DB Parameter Group | ✓ | ✓ | - | - |
| Deleting a DB Parameter Group | ✓ | ✓ | - | - |
| Referring to the DB Parameter Group List | ✓ | ✓ | ✓ | ✓ |
| Checking the DB Parameter Group Information | ✓ | ✓ | ✓ | ✓ |

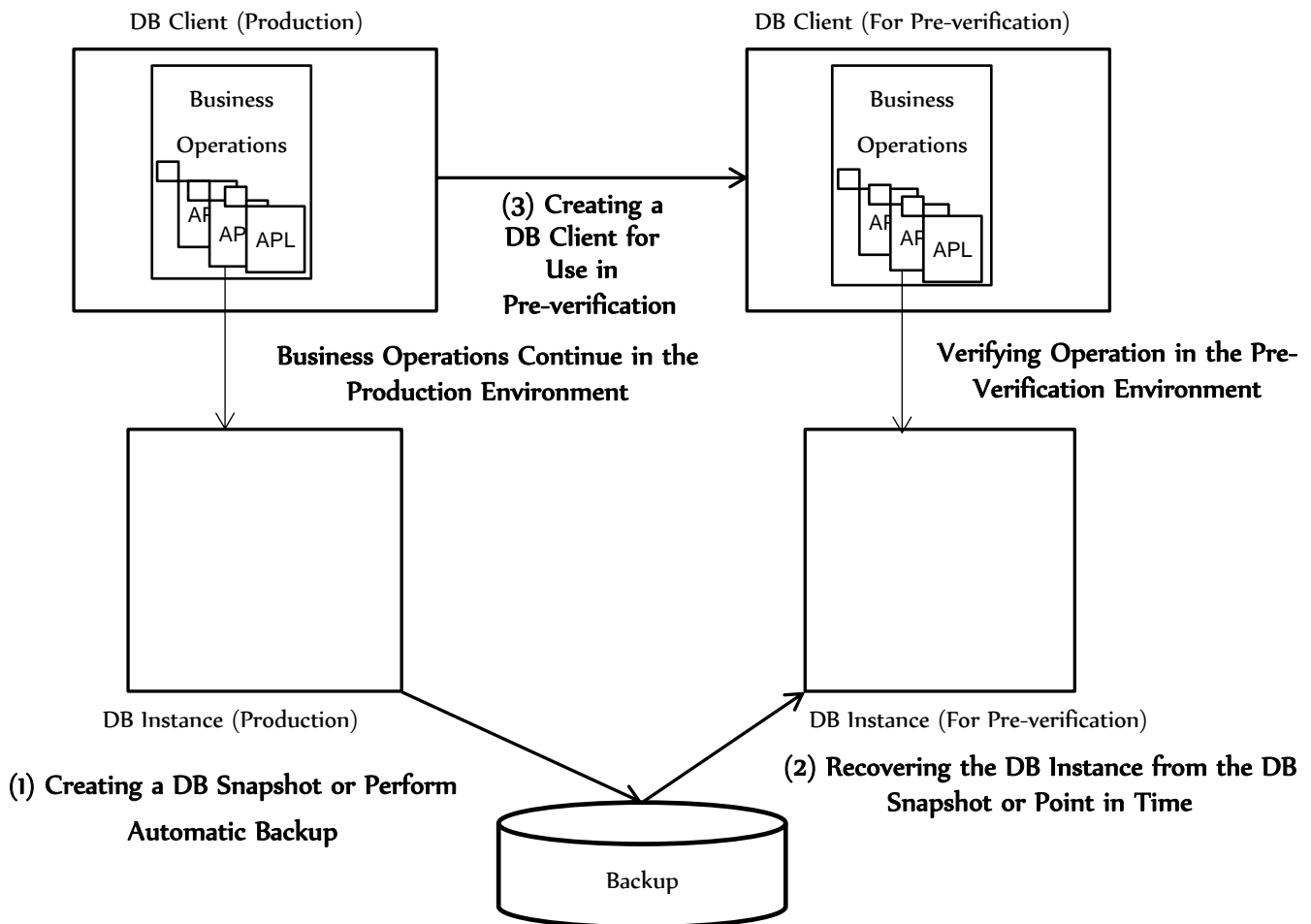
| | | | | |
|--|---|---|---|---|
| Changing a DB Parameter Value | ✓ | ✓ | - | - |
| Creating an Event Notification Registration | ✓ | ✓ | - | - |
| Deleting an Event Notification Registration | ✓ | ✓ | - | - |
| Changing an Event Notification Registration Attribute | ✓ | ✓ | - | - |
| Adding/Deleting a Target for Event Monitoring | ✓ | ✓ | - | - |
| Referring to the Event Notification Registration List | ✓ | ✓ | ✓ | ✓ |
| Checking the Event Notification Registration Information | ✓ | ✓ | ✓ | ✓ |
| Referring to the Event Notification Category List | ✓ | ✓ | ✓ | ✓ |
| Referring to the Event Notification List | ✓ | ✓ | ✓ | ✓ |
| Checking the DB Engine Information | ✓ | ✓ | ✓ | ✓ |
| Referring to the Flavor List | ✓ | ✓ | ✓ | ✓ |
| Checking the Flavor Information | ✓ | ✓ | ✓ | ✓ |

Perform the following procedure when upgrading the DB engine version of an existing DB instance, such as when migrating from a Symfoware Server V12.1 DB instance to an Enterprise Postgres 9.6 DB instance.

[Note]

.....
 - As a Symfoware Server V12.1 DB instance is equivalent to Postgres 9.2, and an Enterprise Postgres 9.6 DB instance is equivalent to PostgreSQL 9.6, it may be necessary to repair existing applications after migrating. For this reason, when upgrading the DB engine version, perform pre-verification using a copy of the DB instance and confirm that the applications work correctly, and then perform migration of the original DB instance in the production environment. The copy of the DB instance for use in pre-verification can be created using the procedure for recovering from a snapshot or the procedure for recovering from a point in time, both of which are described in “5.1 (1) Recovering a DB Instance”.

[Creation of the Pre-verification Environment]



- When performing upgrade of the DB engine version, it is necessary to stop all applications. Calculate the time it will take to perform migration following F. (3)-(10), and estimate the amount of time that business operations will need to be stopped.

- When performing this procedure, in addition to the migration source DB instance, a new DB instance is created as the migration destination. You will be charged for this instance separately from the migration source DB instance.

.....

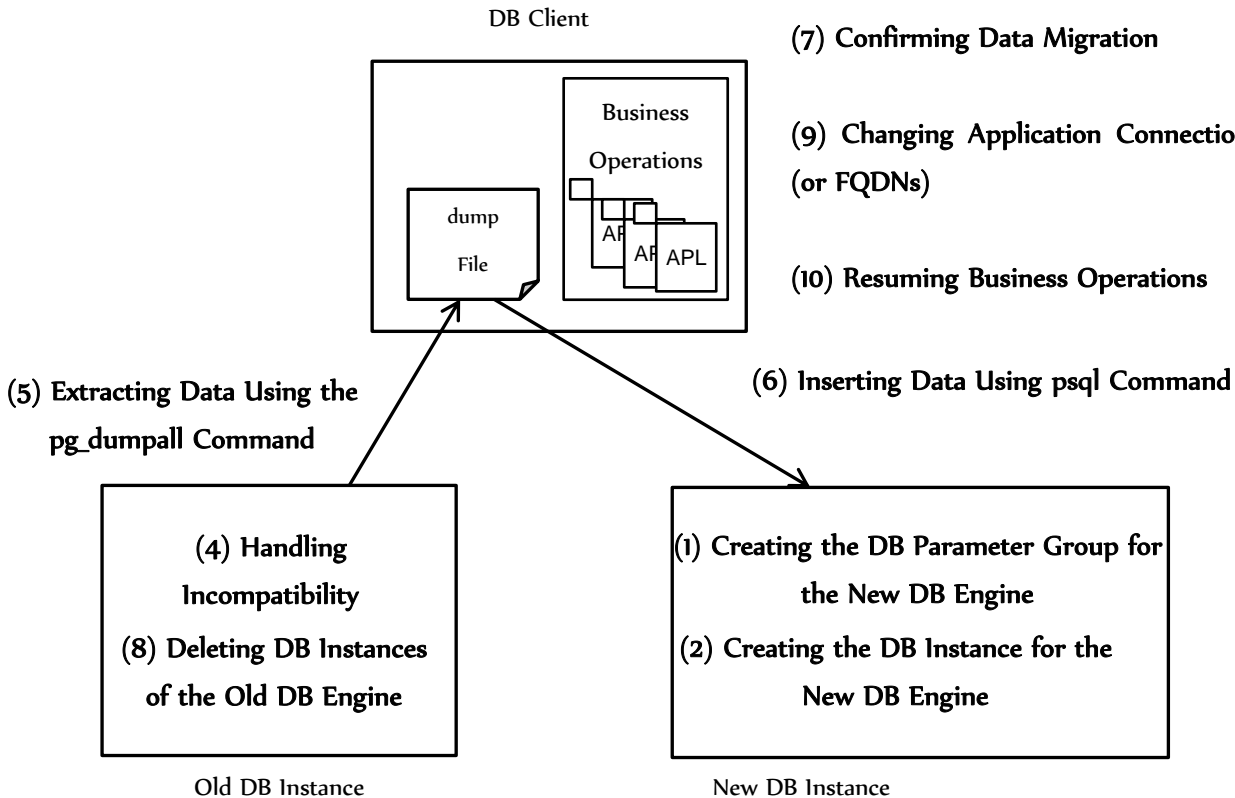
[Overview of DB Engine Version Upgrade]

(3) Stopping Business Operations

(7) Confirming Data Migration

(9) Changing Application Connection Destinations (or FQDNs)

(10) Resuming Business Operations



(1) Creating the DB Parameter Group for the New DB Engine

Refer to "4.5 (1) Creating a DB Parameter Group" and "(2) Changing a Parameter Value", and then create the DB parameter group and change the parameter values.

For a list of parameters that can be changed using each DB engine, refer to "Appendix A.1: List of DB Parameters That Can Be Changed".

(2) Creating the DB Instance for the New DB Engine

Refer to "2.2 Creating a DB instance" and create the DB instance for the new DB engine.

In this procedure, specify the DB parameter group created in (1) above.

(3) Stopping Business Operations

Stop any business operations involving the old DB instance.

(4) Handling Incompatibility

<When Upgrading from Symfoware Server to Enterprise Postgres>

In Enterprise Postgres, as role names beginning with "pg_" will be recognized as reserved words, if any such role names exist in the old DB, the version upgrade will fail. Therefore, be sure to delete or change any role names that begin with "pg_" before upgrading.

(5) Extracting Data Using the pg_dumpall Command

Perform this operation using the DB client (Enterprise Postgres Client or Symfoware Server Client) corresponding to where the old DB engine is installed.

Obtain the database data and schema from the old DB instance by executing the command shown below.

[Note]

.....

- Confirm the size of the DB in advance by connecting to the target DB and executing the following DB command.

```
SELECT pg_size_pretty(sum(pg_database_size(datname))) AS dbsize FROM pg_database;
```

As the pg_dumpall command outputs the DB data as SQL commands, the file that is created when executing this commands will be larger than the size of the DB confirmed using the above command (for example, the integer "2147483647" will be 4 bytes when output as DB data, and 10 bytes when output as an SQL command character string). For this reason, be sure to execute the command in a location with plenty of disk space available.

.....

| Command Example |
|--|
| CLIENT_DIR= <i>[Directory path where the DB instance is installed]</i> export PATH=\${CLIENT_DIR}/bin:\${PATH} export LD_LIBRARY_PATH=\${CLIENT_DIR}/lib:\${LD_LIBRARY_PATH} FQDN= <i>[Connection destination of the old DB instance (For details on checking FQDNs, refer to section 2.3 (1))]</i> PORT= <i>[Port number of the old DB instance (For details on checking port numbers, refer to section 2.3 (1))]</i> MASTERUSERNAME= <i>[Database administrator name for the old DB instance]</i> pg_dumpall -h \${FQDN} -p \${PORT} -U \${MASTERUSERNAME} > db.dump |

(6) Inserting Data Using psql Command

Execute the following commands on the DB client where the client of the new DB engine is installed.

Insert the schema and data extracted in (5) in the new DB instance created in (2).

| Command Example |
|---|
| CLIENT_DIR= <i>[Directory path where the DB instance is installed]</i> export PATH=\${CLIENT_DIR}/bin:\${PATH} export LD_LIBRARY_PATH=\${CLIENT_DIR}/lib:\${LD_LIBRARY_PATH} FQDN= <i>[Connection destination of the new DB instance (For details on checking FQDN, refer to section 2.3 (1))]</i> PORT= <i>[Port number of the new DB instance (For details on checking port numbers, refer to section 2.3 (1))]</i> |

```
MASTERUSERNAME=[Database administrator name for the new DB instance]  
psql -h ${FQDN} -p ${PORT} -U ${MASTERUSERNAME} -d postgres -f db.dump
```

[Note]

-
- When inputting DB data, a message is output indicating that registration of the following extensions has failed. Ignore this message.
 - pgx_cpu
 - pgx_disk
 - pgx_io
 - pgx_log
 - pgx_memory
 - pgx_network
 - pgx_network_err
 - pgx_paging
 - pgx_process
 - pg_stat_statements
-

(7) Confirming Data Migration

Connect to the new DB instance and confirm that migration of the data has completed successfully.
For the procedure to connect to a DB instance, refer to "2.3 Connecting to a DB Instance".

(8) Deleting DB Instances of the Old DB Engine

Refer to "6.1 Deleting a DB Instance", and delete the DB instance of the old DB engine.

(9) Changing Application Connection Destinations (or FQDNs)

Use either of the following methods to configure the system so that applications connect to the new DB instance.

<Change the Connection Destination of the Applications>

Change the connection destination configured for applications to the FQDN of the new DB instance.

<Change the FQDN>

Change the FQDN of the DB instance of the new DB engine to that of the DB instance of the old DB engine.

In addition, if the DB instances were created using Heat, also change the ID of the new DB instance to that of the old DB instance.

| Command Example |
|---|
| INSTANCEID=[<i>DB instance ID of the new DB engine</i>] |
| PRIVATE_ADDRESS=[<i>FQDN of the DB instance of the old DB engine</i>] |
| OLD_INSTANCEID=[<i>DB instance ID of the old DB engine</i>] |
| APPLY_IMMEDIATELY=[<i>Immediate application of DB instance enabled/disabled (Select true or false)</i>] |

```

ENDPOINT=[Endpoint of the database service]

curl -X PUT -i ${ENDPOINT}/v1.0/${TENANTID}/instances/${INSTANCEID} -H "X-Auth-Token: ${TOKEN}" -H "Content-Type: application/json" -d
"{\"instance\": {\"privateAddress\": \"${PRIVATE_ADDRESS}\", \"id\": \"${OLD_INSTANCEID}\", \"applyImmediately\": \"${APPLY_IMMEDIATELY}\"}"

Example of Execution Results

$ INSTANCEID=enterprisepostgres-instance-id
$ PRIVATE_ADDRESS="XXXXXXXX.XXX.XXX"
$ OLD_INSTANCEID=symfoware-instance-id
$ APPLY_IMMEDIATELY=true
$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com
$ curl -X PUT -i ${ENDPOINT}/v1.0/${TENANTID}/instances/${INSTANCEID} -H "X-Auth-Token: ${TOKEN}" -H "Content-Type: application/json" -
d  "{\"instance\":  {\"privateAddress\":  \"${PRIVATE_ADDRESS}\",  {\"id\":  \"${OLD_INSTANCEID}\",  \"applyImmediately\":
\"${APPLY_IMMEDIATELY}\"}"
HTTP/1.1 202 Accepted
(Omitted)

```

(10) Resuming Business Operations

Resume business operations.

[Reference]

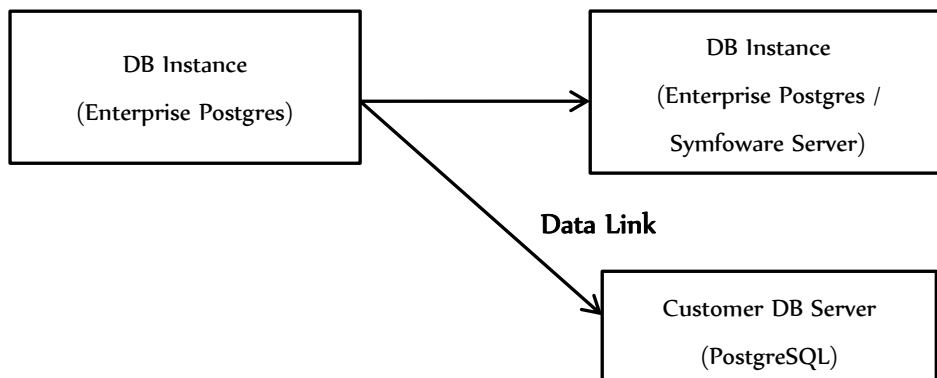
-
- For details on the `pg_dumpall` command, refer to the following sections in the documentation of the migration source DB engine (the "PostgreSQL 9.6.2 Documentation" or the "PostgreSQL 9.2.4 Documentation"):
 - "`pg_dumpall`" in "II. PostgreSQL Client Applications" in "VI. Reference"
 - For details on the `psql` command, refer to the following sections in the documentation of the migration source DB engine (the "PostgreSQL 9.6.2 Documentation" or the "PostgreSQL 9.2.4 Documentation"):
 - "`psql`" in "II. PostgreSQL Client Applications" in "VI. Reference"
-

G.1 postgres_fdw

Using postgres_fdw, you can link a DB instance to another DB instance or an external PostgreSQL DB.

postgres_fdw can be used when using an Enterprise Postgres DB instance.

[postgres_fdw Configuration Example]



The following DBs can be connected to using postgres_fdw:

- DB instances (Enterprise Postgres or Symfoware Server)
- DBs other than DB instances (Enterprise Postgres, Symfoware Server (Open Interface), PostgreSQL 8.3 and later)

G.2 How to Use

In order to use postgres_fdw, the following conditions must be met:

- There must be a path for communication between the connection source and the connection destination DB (*1).
- It must be possible for the connection source to resolve the FQDN of the connection destination DB (*2). (When an FQDN is specified for the connection destination)
- The connection destination must have access authority for the connection destination DB. (It is not necessary to consider this when the connection destination is a DB instance)

(*1)

It is necessary to configure the following rules in the security group of the connection source DB instance.

| Purpose of Rule | Direction | Protocol | Permission Target | Port |
|-------------------------------------|-----------|----------|--|--|
| DB communication using postgres_fdw | egress | tcp | Security group of the connection destination DB instance, or connection destination server | Port number of the connection destination DB |
| Name resolution | egress | udp | DNS server | 53 |

When the connection destination DB is a DB instance or a VM in the IaaS, it is also necessary to configure the following rule in the security group.

| Purpose of Rule | Direction | Protocol | Permission Target | Port |
|-------------------------------------|-----------|----------|---|--|
| DB communication using postgres_fdw | ingress | tcp | Security group of the connection source DB instance | Port number of the connection destination DB |

(*2)

Configure a DNS server for all subnets in the DB subnet group that was specified during creation of the DB instance.

[Reference]

.....

➤ For details on postgres_fdw, refer to "F.33 postgres_fdw" in the "PostgreSQL 9.6.2 Documentation".

.....

When distributing the load between DB instances and performing reference operations using a read replica, create a read replica inside the region. Perform the following procedure.

(1) Create the DB Instance

Refer to 2.1 and 2.2, create the following, and then create the DB instance.

- A network subnet
- A virtual router
- A security group
- A DB subnet
- A DB parameter group
- A DB instance

In order to create a Read Replica, the DB instance must fulfill the requirements below.

- Use Enterprise Postgres 9.6 or later as the DB engine
- Be a redundant DB instance
- Have automatic backup enabled

However, when you already have a DB instance that fulfills these requirements that is on a tenant, it is not necessary to re-create that DB instance.

When creating a new Read Replica for a DB instance that does not fulfill the requirements above, perform the following.

- When using Symfaware Server V12.1 as the DB engine, refer to Appendix F, and perform upgrade to Enterprise Postgres.
- When using a DB instance with a single configuration, refer to 4.3 and migrate to a DB instance with a redundant configuration.
- When automatic backup has not been enabled, refer to 4.4 and enable automatic backup.

(2) Connect from the DB Client to the DB Instance

Refer to 2.3 (2), and connect from the DB client to the DB instance.

(3) Create a Read Replica

When performing reference operations on the read replica, consider whether the settings of the following should be different from those of the DB instance.

- The flavor
- The DB parameter group

After that, use the read replica creation API to create a read replica.

| Command Example |
|--|
| AZ=[AZ name (Select the desired name from the released availability zones)] |
| INSTANCEID=[ID of the DB instance of the read replica (Set any desired value)] |

| |
|--|
| <p>INSTANCENAME=[Read replica name (Set any desired value)]</p> <p>SUBNETGROUPID=[ID of the DB subnet group]</p> <p>SECGRP_ID=[ID of the security group]</p> <p>PARAMG_ID=[ID of the DB parameter group for the read replica]</p> <p>FLAVOR=[ID of the flavor (Select the desired name from the released flavors)]</p> <p>SOURCE_INSTANCEID=[ID of the original DB instance]</p> <p>ENDPOINT=[Endpoint of the database service in the disaster response region]</p> <pre>curl -X POST -i \${ENDPOINT}/v1.0/\${TENANTID}/instances -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d "{\"action\":\"readreplica\", \"instance\": {\"availabilityZone\": \"\${AZ}\", \"name\": \"\${INSTANCENAME}\", \"subnetGroupId\": \"\${SUBNETGROUPID}\", \"flavorRef\": \"\${FLAVOR}\", \"parameterGroupId\": \"\${PARAMG_ID}\", \"securityGroupIds\": [\"securityGroupId\": \"\${SECGRP_ID}\", \"id\": \"\${INSTANCEID}\", \"readReplicaSrcDBInstance\": {\"id\": \"\${SOURCE_TENANTID}\"}}}"</pre> |
| <p>Example of Execution Results</p> |
| <pre>\$ AZ=jp-east-1a \$ INSTANCEID=replica-instance-id \$ INSTANCENAME=replica-instance-name \$ SUBNETGROUPID=test-subnetgroup-id \$ FLAVOR=i101 \$ SECGRP_ID=XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX \$ PARAMG_ID=replica-paramg-id \$ SOURCE_INSTANCEID=test-instance-id \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X POST -i \${ENDPOINT}/v1.0/\${TENANTID}/instances -H "X-Auth-Token: \${TOKEN}" -H "Content-Type: application/json" -d "{\"action\":\"readreplica\", \"instance\": {\"availabilityZone\": \"\${AZ}\", \"name\": \"\${INSTANCENAME}\", \"subnetGroupId\": \"\${SUBNETGROUPID}\", \"flavorRef\": \"\${FLAVOR}\", \"parameterGroupId\": \"\${PARAMG_ID}\", \"securityGroupIds\": [\"securityGroupId\": \"\${SECGRP_ID}\", \"id\": \"\${INSTANCEID}\", \"readReplicaSrcDBInstance\": {\"id\": \"\${SOURCE_TENANTID}\"}}}" HTTP/1.1 202 Accepted (Omitted)</pre> |

Executing the read replica creation API copies the data from the original DB instance to the disaster response region and creates the read replica. Therefore the amount of time necessary to create the read replica depends on the amount of data involved.

To check whether the creation of the read replica is complete, execute the DB instance information reference API for the read replica. When creation is complete, the status will become Active and the replicaStatus will become Replicating. When the data in the DB instance is updated after creation of the read replica is complete, change logs are progressively sent to the read replica and data is reflected asynchronously.

| Command Example |
|--|
| <p>INSTANCEID=[ID of the DB instance of the read replica]</p> <p>ENDPOINT=[Endpoint of the database service]</p> |

```
curl -X GET -i ${ENDPOINT}/v1.0/${TENANTID}/instances/${INSTANCEID} -H "X-Auth-Token: ${TOKEN}"
```

| Example of Execution Results |
|---|
| <pre>\$ INSTANCEID=replica-instance-id \$ ENDPOINT=https://database.jp-east-1.cloud.global.fujitsu.com \$ curl -X GET -i \${ENDPOINT}/v1.0/\${TENANTID}/instances/\${INSTANCEID} -H "X-Auth-Token: \${TOKEN}" HTTP/1.1 200 OK (Omitted) {"instance":{"volume":{"size":10,"type":"M1"},"id":"test-instance-id","name":"test-instance- name","description":null,"multiAZ":false,"multi":true,"port":26500,"preferredBackupWindow":"17:50- 18:20","preferredMaintenanceWindow":"mon:01:46- mon:02:16","preferredRecoveryTime":{"applyImmediately":true,"time":null},"securityGroupIds":[{"securityGroupId":"XXXXXXXX-XXXX-XXXX-XXXX- XXXXXXXXXXXX"}],"parameterGroupId":"DefaultGroup- 12.1","backupRetentionPeriod":0,"autoMinorVersionUpgrade":true,"engineVersion":"9.6","engineMinorVersion":0,"autoMaintenance":true,"availabilit yZone":"XXXXXXXX","subnetGroupId":"test-subnetgroup- id","publiclyAccessible":false,"engine":"enterprisepostgres","masterUserName":"masterusername","characterSet":"UTF8","collate":"C","created":"2016- 07- 12T02:09:01Z","flavor":{"id":"1101","links":[{"href":"http://XXXXXXXX/v1.0/XXXXXXXX/flavors/1101","rel":"SELF"},{"href":"http://XXXXXXXX/XXXXXXX/flavo rs/1101","rel":"BOOKMARK"}]},"links":[{"href":"http://XXXXXXXX/v1.0/XXXXXXXX/instances/test-instance- id","rel":"SELF"},{"href":"http://XXXXXXXX/XXXXXXXX/instances/test-instance- id","rel":"BOOKMARK"}],"status":"Active","updated":null,"privateAddress":"XXXXXXXX.XXX.XXX","privateIp":"XX.XX.XX.XX","publicAddress":null,"publi clp":null,"subPrivateIp":null,"subPublicIp":null,"pendingModifiedValues":null,"readReplicaDBInstanceIdentifiers":[],"readReplicaSrcDBInstanceIdentifi er":null,"secondaryAvailabilityZone":null,"readReplicaDBInstances":[],"readReplicaSrcDBInstance":{"region":"jp-east-1","tenantId":"YYYYYYYY-YYYY- YYYY-YYYY-YYYYYYYYYYYY","id":"test-instance-id"},"replicaStatus":"Replicating"}}</pre> |

(4) Connect from the DB Client to the Read Replica

Refer to 2.3 (2), and connect client to the read replica from the DB. The FQDN to use at this time is the FQDN of the read replica.

Appendix 1: List of Restrictions

For details on the restrictions of the IaaS Database Service, refer to the "IaaS Restrictions / Advisory Notes" at <http://jp.fujitsu.com/solutions/cloud/k5/document/>.

FUJITSU Cloud Service for OSS IaaS
Database Service User Guide Ver. 1.7
Issued: Nov, 2018
All Rights Reserved, Copyright FUJITSU LIMITED 2018

- The content of this manual is subject to change without prior notice.
- The duplication or reproduction of this manual without permission is prohibited.