Fujitsu Enterprise Postgres for Kubernetes

Technical presentation

FUJITSU

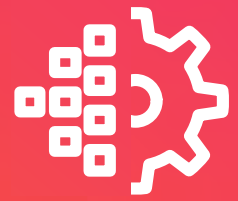
FUJITSU

Cloud and container markets growth

Kubernetes are gaining attention as virtualization technologies that make it easier to deploy and scale applications to shorten development cycles and streamline operations.

Ideal for digital services infrastructure

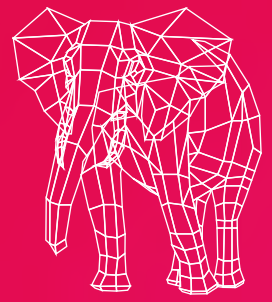
Responsiveness to rapid system changes and growth



Reduced operational maintenance load

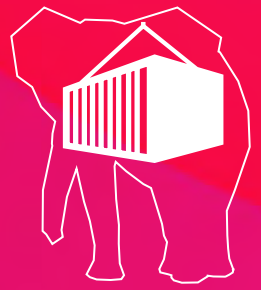
Fujitsu supports the transition to meet hybrid/multi-cloud requirements

Fujitsu Enterprise Postgres



- Enhanced security
- Enhanced performance
- Enhanced reliability

Fujitsu Enterprise Postgres for Kubernetes



- Reduced operational load
- Easy deployment
- No vendor lock-in

Start small and grow	• Quickly deploy from development to operations • Easily scale as your business grows
Reduced operational load	• Reduce the operational load on Database Administrators by automating operations such as failover, recovery, and backup.
Avoid vendor lock-in	• Leverage open container technology to move to the cloud without being locked to cloud vendors.
Security & reliability for business continuity	• Protects data from threats such as theft and falsification and ensures stable operation.
Ensured quality and compatibility	• Certified as a Red Hat OpenShift operator.

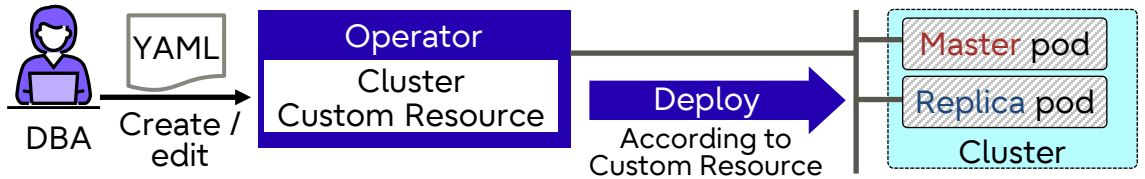
What is Fujitsu Enterprise Postgres Operator?

What is an Operator?

Operators automate the lifecycle of target software.

Fujitsu Enterprise Postgres Operator

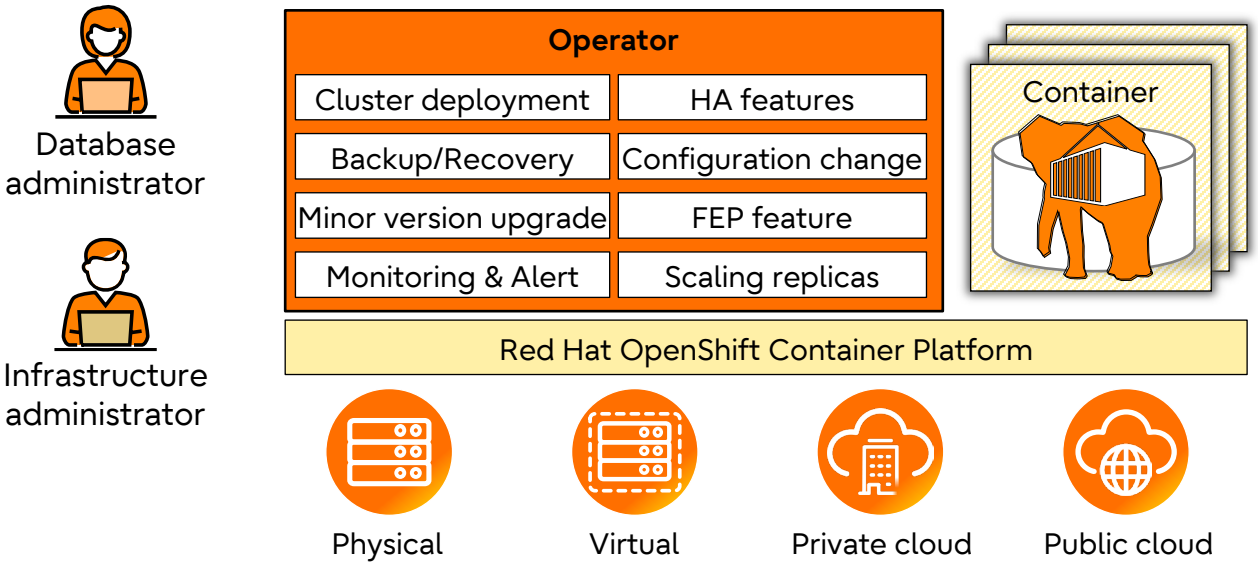
- Deploys and manages the following in a Red Hat OpenShift Container Platform environment:
  - FEPCluster, pgpool2cluster, backup, restore, exporter
- Manages operations consistently according to predefined Custom Resources (CR).
  - CR is a YAML file that defines the configuration of a system.



Operator features

Provides operator services to automate the creation and operation of databases on your container management infrastructure.

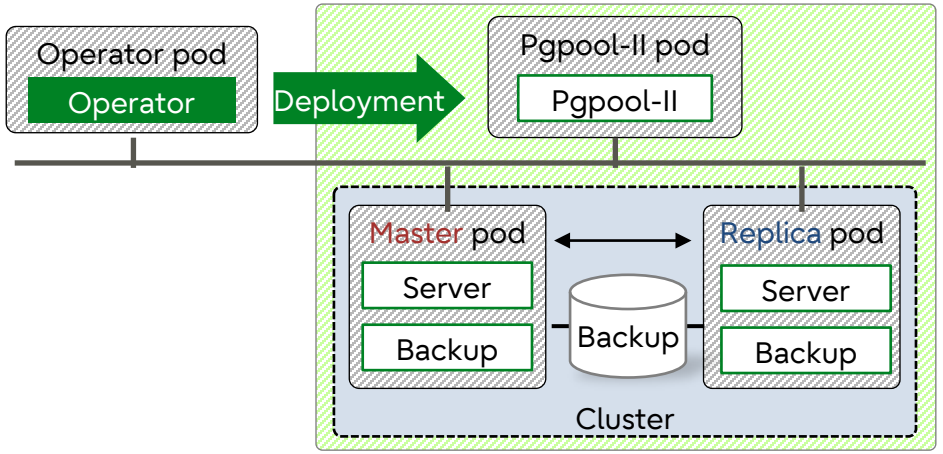
Operators make it easy to deploy anywhere in multi-cloud and hybrid cloud environments.



Easy deployment

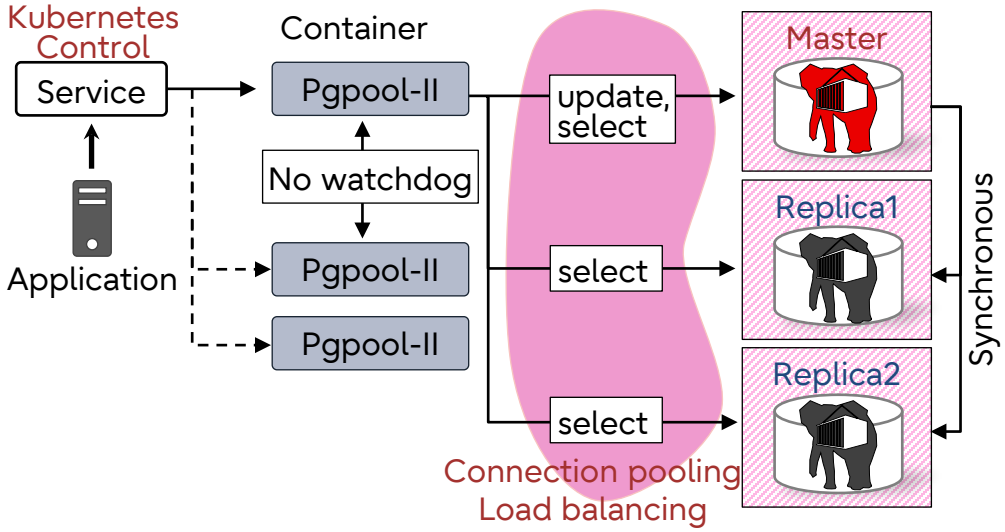
FUJITSU

- Easily build highly available Fujitsu Enterprise Postgres clusters
  - The Operator execute requests in minutes and provides the functionality needed to take full advantage of Fujitsu Enterprise Postgres.
  - The Operator deploys standalone and highly available Fujitsu Enterprise Postgres clusters in pre-defined configurations and start with a small workload. Configuration parameters can be tuned during and after deployment to ensure that the instances are suitable for the workload.



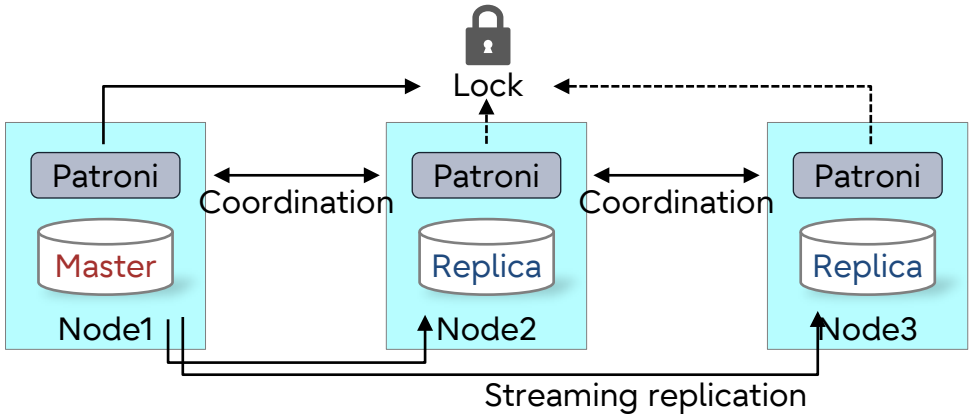
Load balancing

- Deploying Pgpool-II and connecting to the cluster from Operator
  - Users can deploy the Pgpool-II container and access the database via Pgpool-II to use load-balancing and connection pooling features.
  - Multiple Pgpool-II containers can be deployed for load sharing and high availability. Users can request a Kubernetes service to distribute their work across multiple Pgpool-II containers.



High availability

- High availability features
  - Automatic failover
  - Automatic recovery
  - Manual switchover
- Automatic failover overview
  - High availability and failover management are provided by Patroni.
  - If Patroni detects a failure in the cluster, such as the crash of a Postgres process or the outage of a container running the database, Patroni automatically conducts failover.



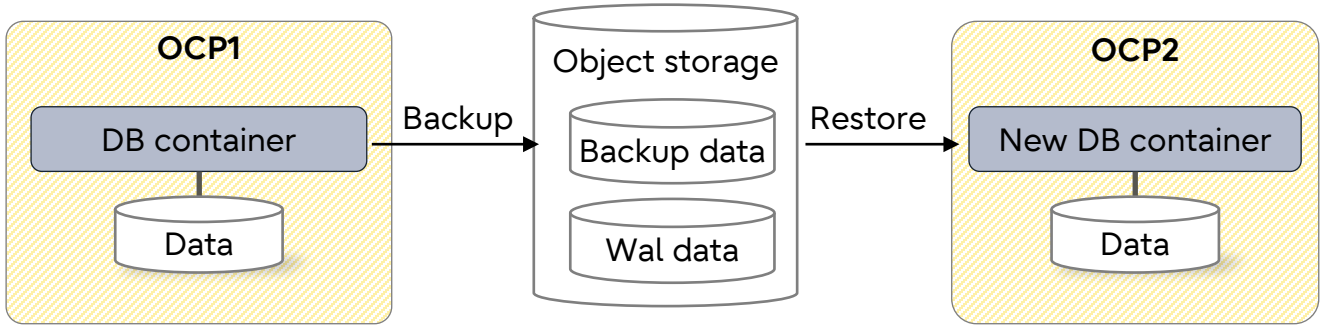
Backup and restore

FUJITSU

- Reliable, affordable backup with object storage
  - The backup container is deployed as a complement to each server pod.
  - The backup runs at user-specified scheduled times (similarly to crontab).
  - pgBackRest is used for backup and WAL archiving.
    - Automatic backup
      - Full backup / incremental backup
    - Backup location
      - NFS fixed volumes
      - AWS S3 compatible storage
      - Red Hat OpenShift Container Storage
    - Point-in-time recovery (PITR)
      - Restore the cluster with point-in-time recovery from manual or automatic backup.
    - Restore type
      - Restore backup data to an existing cluster.
      - Create a new cluster and restore backup data.

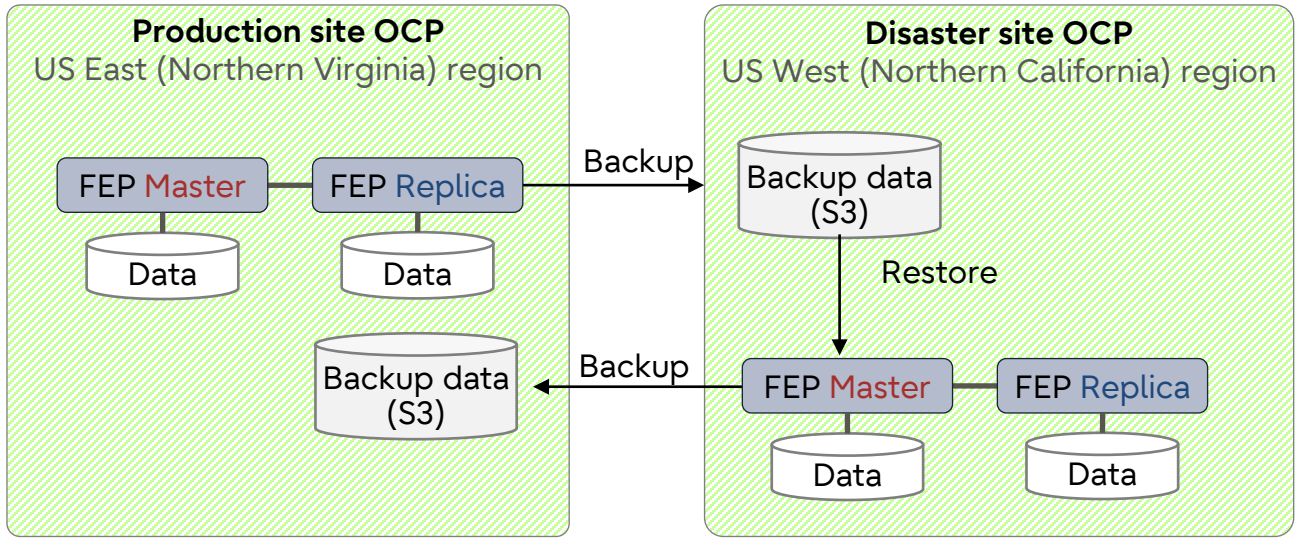
Disaster Recovery - Overview

- Store backups in object storage for data recovery to different OCPs
  - configurations: multi-region, multi-vendor
  - RPO : Ensures data recovery in less than five minutes in the event of a disaster
  - RTO : data volume dependent



Disaster Recovery - Use case

- Data can be restored between OCPs built between different regions.
  - Example: When OCPs in the US East region are damaged, the database can be restored to OCPs in the US West region and continue operation.
- Ability to restore data between OCPs from different vendors
  - Example: Migrate on-premises OCP DB containers to cloud OCP



Disaster Recovery - configuration point

FUJITSU

- Object Storage design
  - For disaster environments where storage to store backups must be designed to meet backup/restore requirements, the OCP environment and storage to store backups are in separate regions
- Restore time
  - Estimate business downtime in the event of a disaster by measuring restore times prior to operation, as it depends on the amount of data

Configuration changes

FUJITSU

- Easy configuration and resource changes
  - Changing parameters
    - Configuration files
      - postgresql.conf
      - pg_hba.conf
      - pgaudit.conf
    - Depending on the parameter, the change will take effect in either of the following:
      - Immediately after the change
      - After the server process is restarted
  - Modifying resources (CPU, memory)
    - The allocated resources for the following containers can be changed by the FEPCluster custom resource:
      - Server containers
      - Backup containers
      - Pgpool-II containers

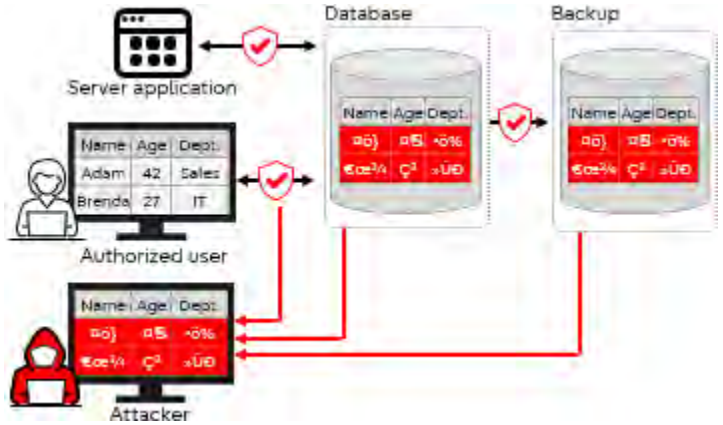
Security

Data security is one of the top concerns for enterprises –
Fujitsu Enterprise Postgres for Kubernetes has the right tools for that

FUJITSU

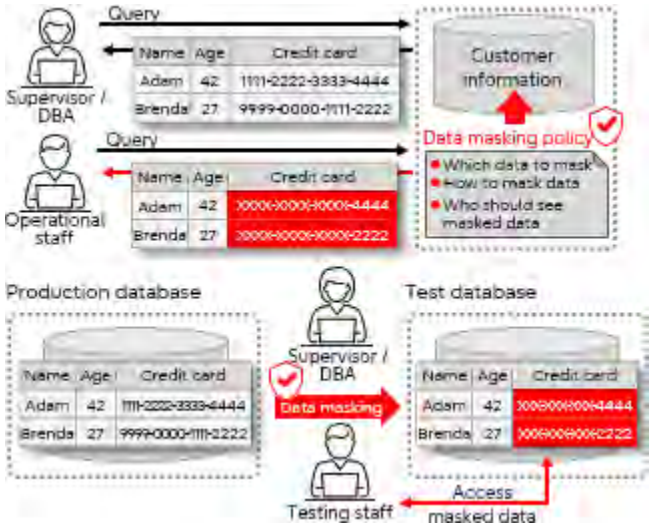
Transparent Data Encryption

All data in Fujitsu Enterprise Postgres can be encrypted using Advanced Encryption Standard, a PCI DSS-compliant 256-bit encryption technology that is standard for the credit card industry.



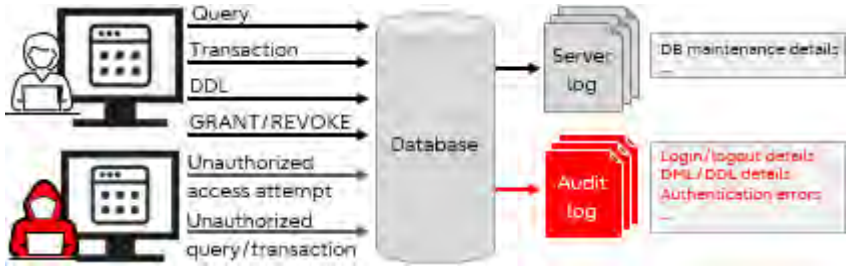
Data Masking

Data masking minimizes security risk by enabling user-based confidentiality, altering original data while maintaining its usability. Redaction is applied via powerful, user-friendly policies.



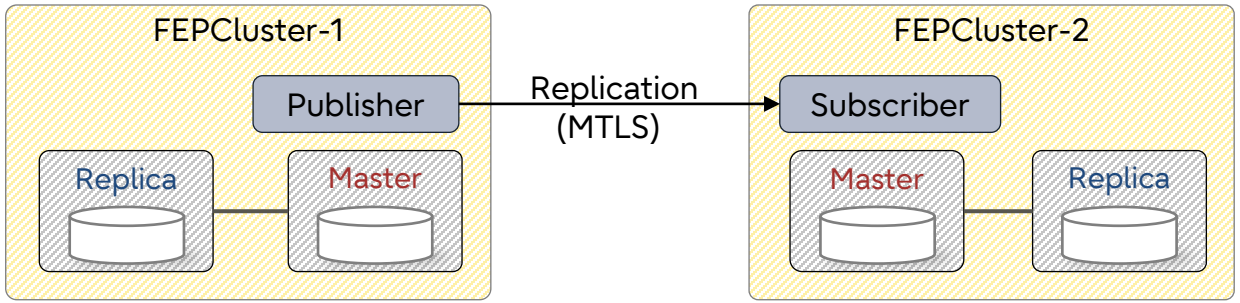
Dedicated Audit Log

Fujitsu Enterprise Postgres extends PostgreSQL's auditing by allowing the system to save audit records to a separate file – the Dedicated Audit Log



Logical replication

- The logical replication feature of PostgreSQL is supported.
- Logical replication provides fine-grained control over data replication and security.

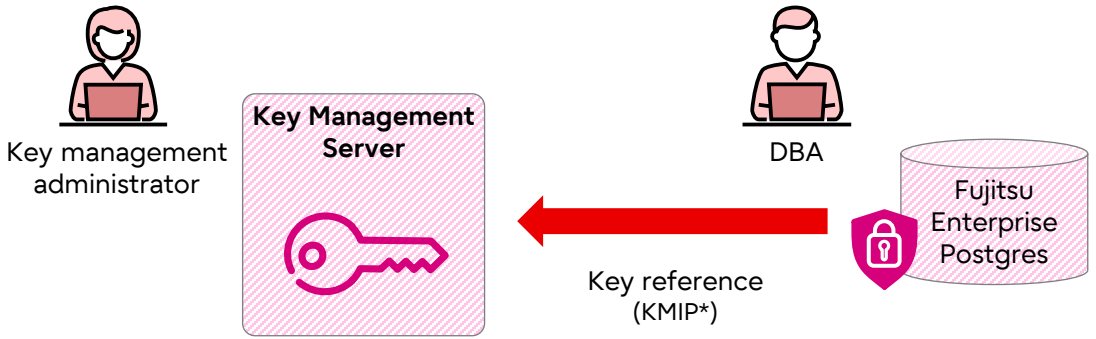


- Use case
  - Users can replicate data between different architectures, such as between public clouds and IBM LinuxONE™.

Key management for Transparent Data Encryption - Overview

FUJITSU

- Key Management for Transparent Data Encryption works with KMIP*, an open protocol
  - Improved data security
    - Improved flexibility and data security using a KMIP-certified Foreign Key Management System (KMS).
  - Reduced risk of data leakage
    - Reduced risk of data leakage by storing encryption keys outside the database.
  - Improved governance
    - The division of roles between database administrator and master encryption key administrator improves governance.

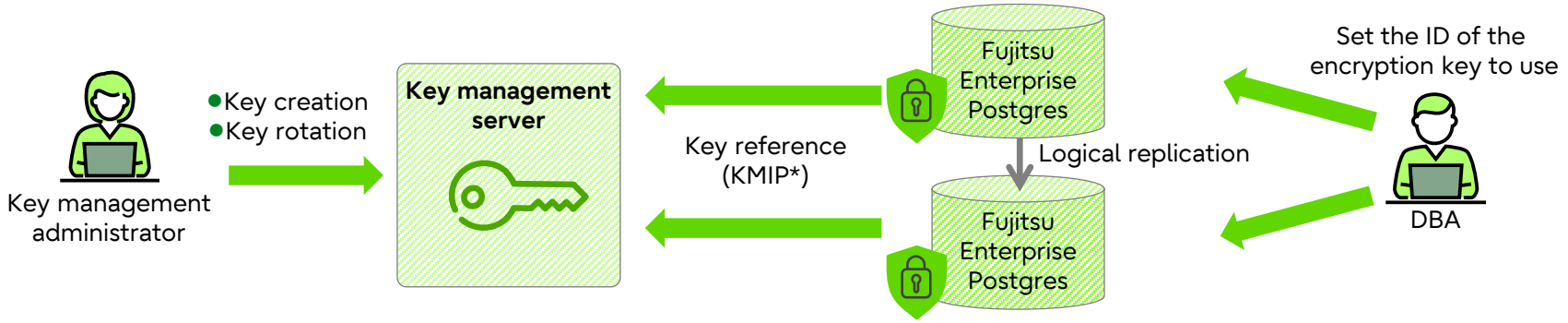


*KMIP: Key Management Interoperability Protocol

Key management for Transparent Data Encryption – Use case

FUJITSU

- Centralized key management and cost savings for systems that connect multiple locations with logical replication



*KMIP: Key Management Interoperability Protocol

Note

- Key management service availability and key lifecycle management
  - Database cannot be started if Key Management Service is not running properly
  - Since the encryption key itself is not backed up in Fujitsu Enterprise Postgres, data cannot be decrypted if the encryption key is lost
- As a countermeasure, request the following from the key administrator
  - Ensure key management services are up and running to meet database availability requirements
  - Proper key backup

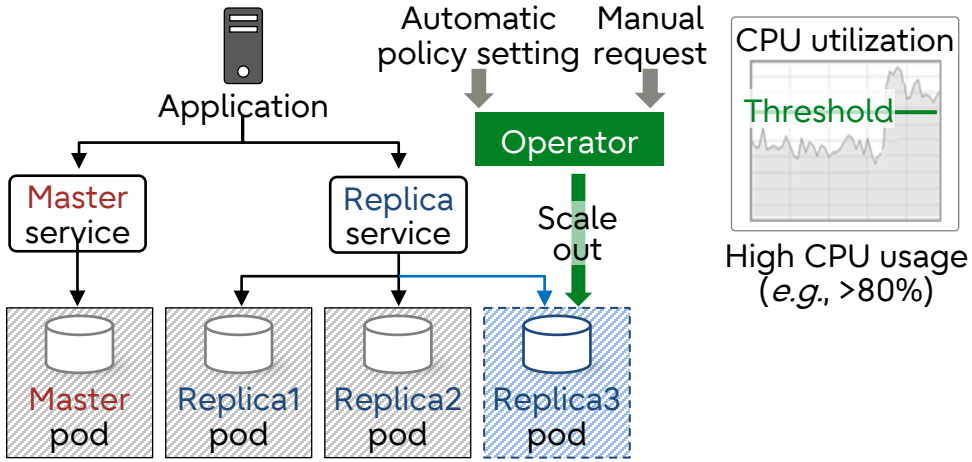
Key management for Transparent Data Encryption - Differences from original Transparent Data Encryption

	Original Transparent Data Encryption	Transparent Data Encryption with HPCS	Transparent Data Encryption with KMIP*
Key storage area	Local to the DB server	Key management server on the cloud service	Key management server
Operating environment	Anywhere	IBM Cloud Service	Anywhere (Open Protocol)
Centralized key management	Not available	Not available	Available
Master Encryption Key lifecycle management	Performed by the DBA	Performed by the DBA	DBA (FEP) is the key user. Key management is responsibility of the key administrator

*KMIP: Key Management Interoperability Protocol

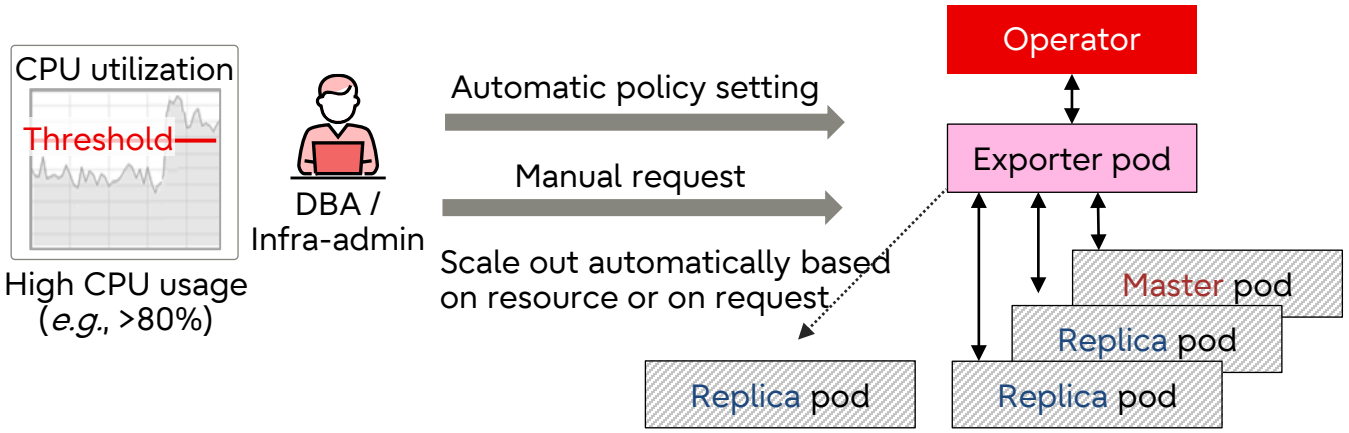
Scaling replicas - Overview

- Manual scale-out/scale-in
  - Scale out/in can be performed in the OpenShift web console in a few steps.
    - Database health can be also be checked using the web console
- Automatic scale-out
  - Dynamic database cluster expansion is conducted according to pre-defined scaling policies. Build and operational costs are reduced to respond to rapid transaction growth.



Scaling replicas – Use case

- Manual scale-in
  - Reduce pods manually during off-season with fewer transactions.
- Automatic scale-out
  - Maintain performance under temporary high load conditions.



Scaling replicas - Auto scale-out configuration

FUJITSU

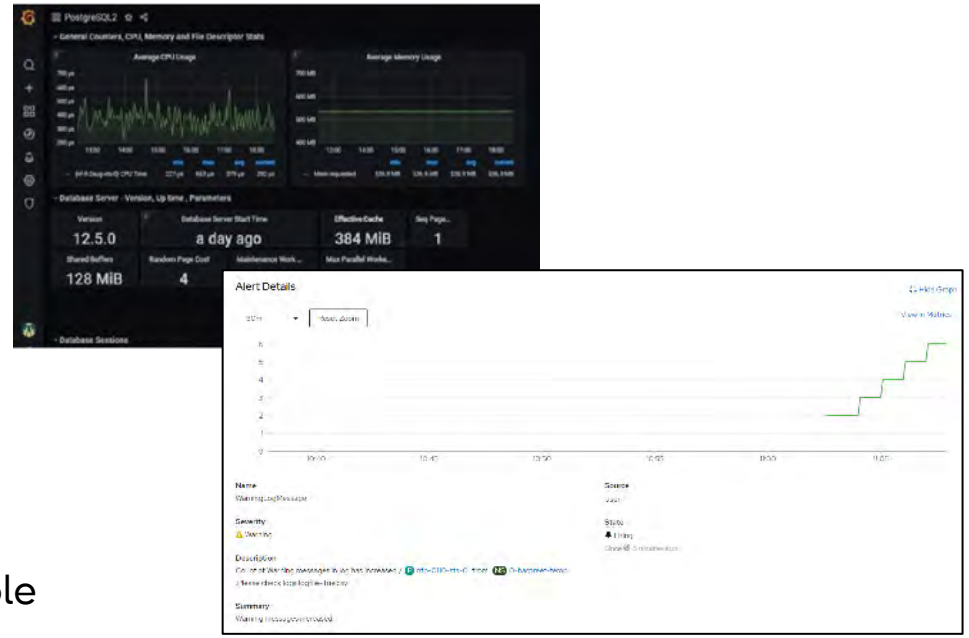
- Auto scale-out can be configured with the following threshold
  - Average CPU utilization of the cluster
  - Average number of connections to the cluster
- Maximum number of replicas that can be scaled out
  - Up to 15 replicas

Note

- When using the auto scale-out feature, consider the synchronous mode.
- When the workload on the system decreases, users should consider scaling in to reduce redundant resources. This is performed manually by editing FEPCluster CR. For details, refer to the Fujitsu Enterprise Postgres for Kubernetes User's Guide.

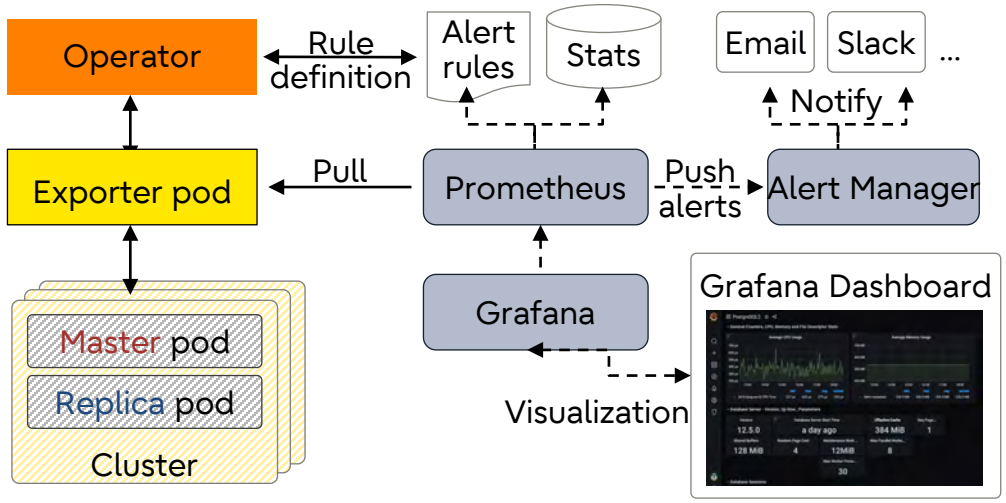
Monitoring - Overview

- The monitoring feature enables DBAs to manage the container environment to meet the system needs.
  - Running the environment 24 x 7
  - Maintaining desired database performance
- Key monitoring items include:
  - Physical resources
  - Workload
  - Connections
  - Database resource / diagnostics info
  - Load balance status
  - Replication / backup status
  - Liveliness
  - database server logs
- Out-of-the-box real-time monitoring, also configurable:
  - Monitoring items can be viewed on the OpenShift web console
  - The monitoring report is integrated to Grafana
  - Alerting can be set based on the monitoring metrics
  - Trend analysis and historical data is available
- Access control
- Providing a database log analysis feature using pgBadger
- Monitor the information collected in the audit log and notify you when a log that meets the preset conditions is output.
- Alerts are provided by working with SIEM tools such as kibana alert



Monitoring and alert – Use cases

- Monitoring is provided by Prometheus and Grafana, the de-facto standard monitoring tools for Kubernetes.
- Prometheus, the monitoring software, collects CPU, memory, and disk usage resource information, as well as health status of cluster Pods. DBAs and Infrastructure Administrators can view monitoring data captured via Grafana features in an advanced graphical display.
- DBAs obtain a holistic view of their environment, from physical resources to workload
  - Alert-based proactive monitoring
  - Issue investigation driven by events (alerts)
  - Regular performance review



Monitoring – Reporting

FUJITSU

- Contents of the report screen
  - Select the Grafana URL from the Networking > Routes section of the OCP platform and report it on the Grafana dashboard screen.
- How data is managed
  - Reports are archived every 2 hours and deleted after a specified period (default: 15 days)
- How to manage access
  - Grafana allows you to setup access restrictions on a per-group basis. Read-only access and access allowing modifications to report configuration can be granted

Default reporting metrics

	Type	Monitored metrics
1	Server resources	• Average CPU usage, average memory usage
2	Database server	• Version, database server start time • Effective cache, shared buffers, seq page cost, random page cost • Current update data, current insert data, current fetch data • Maintenance work mem, max worker processes • Max parallel workers
3	Database states	• Conflicts/deadlocks, lock tables • Buffers (bgwriter), temp file (bytes), database size • Number of times of table ANALYZE by AUTOVACUUM • Checkpoint stats, cache hit rate • Number of tables without VACUUM in last 24 hours
4	Database transactions	• Number of transactions lasting longer than 5 minutes • INSERT table data (SELECT), fetched database data, return data, DELETE data, UPDATE data, table UPDATED data • Transactions, dead rows, low performing queries • Sequential scans, index scans
5	Sizing	• Schema sizes, data and WAL volume size
6	Connections	• Active connection to max connection ratio • Max connections, connections in idle state for longer than 1 week
7	Database replication lag	• Replication lag on Replica (in seconds)

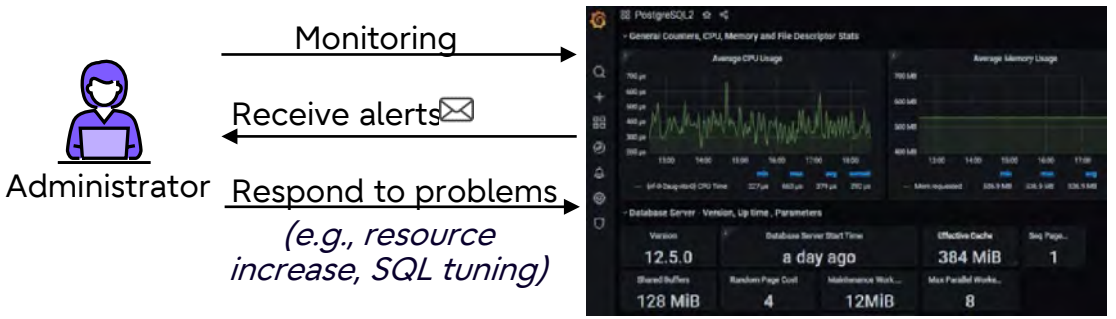
Monitoring - Configuration

- Templates for Grafana dashboards are provided.
- Report view settings
  - Interval
  - Namespace
  - Exporter
  - Database server
  - Database
  - Lock table
  - FEP Pod name

- If you want to change reporting metrics, change the metrics collected on Prometheus.
  If you want to change the graph display, edit it on Grafana.

Deep Insight

- Overview
  - Early detection of signs of failure is important. Prometheus sends alerts with the Alert Manager (via email, Slack, or other tools) to DBAs and Infrastructure Administrators if the collected data (metrics) is unhealthy.
  - Identify issues early and prevent problems before they occur.
- Sample alert rules
  - Server container/pod CPU usage is exceeding 80% of the resource limits
  - Server container/pod memory usage is exceeding 80% of the resource limits
  - PVC (volume) has less than 10% disk available
  - Server apparently went down or is not accessible



Grafana dashboard

Server log monitoring – Use case

FUJITSU

- Alert notification is performed according to the level of log messages to immediately detect errors in the database.
  - In the past, errors in resources and database parameters could be detected, but by monitoring log messages, detailed errors in the database can be reported.
- It is possible to detect queries that affect database performance by using statistical reports obtained by analyzing log files.
  - Detection of slow queries and queries that execute frequently
  - Detecting queries generating many temporary files
  - Detecting Locked Queries

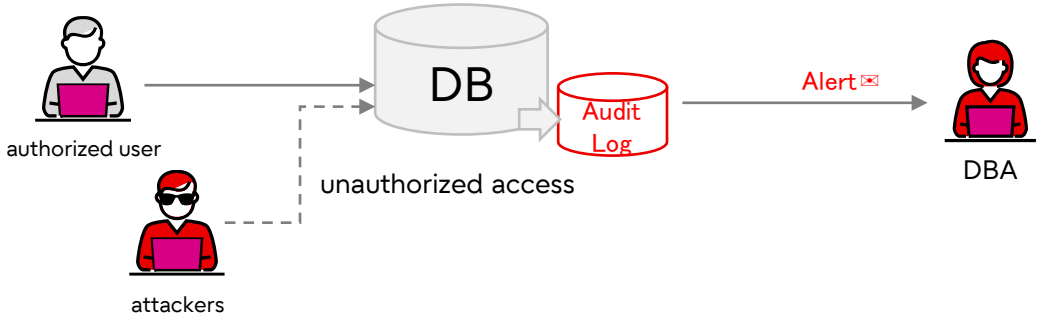
Server log monitoring - Configuration point

FUJITSU

- Log monitoring
  - Set alert rules based on the severity of errors according to the customer's business operations.
    - Example: Immediate detection of FATAL/PANIC/ERROR errors.
      WARNING is notified when more than a certain number is detected per unit time
- Log analysis
  - Preconfiguring Postgres parameters to output information to the database log is recommended
    - log_min_duration_statement: default "-1"(disabled) → "0" or value designed by user
    - log_checkpoints: default "off" → "on“
    - log_connections: default "off" → "on“
    - log_disconnections: default "off" → "on"

Audit log monitoring - Use case

- Quickly respond to external security attacks
- prevent insider attacks such as security breaches
- Instant detection of unauthorized access, data tampering, and other unauthorized operations



Deep Insight - Alerts

FUJITSU

- Prometheus will send alerts for abnormal metrics according to the alert rules.
- Alert rules
  - Default rules

Warning rules	Alert level	Duration	Description
Server Container high CPU utilization	Warning	5 minutes	Server container/pod CPU utilization exceeds 80% of resource limit.
Server Container high RAM usage	Warning	30 minutes	Server container/pod memory usage exceeds 80% of resource limit.
PVC low disk space	Warning	5 minutes	PVC (volume) has less than 10% usable disks
Server Container not found	Warning	60 seconds	Server container/pod has been inaccessible for 60 second.
Postgresql down	Error	-	Server may be down or inaccessible.
Postgresql too many connections	Warning	-	Server container/pod connection usage exceeds 90% of available capacity.

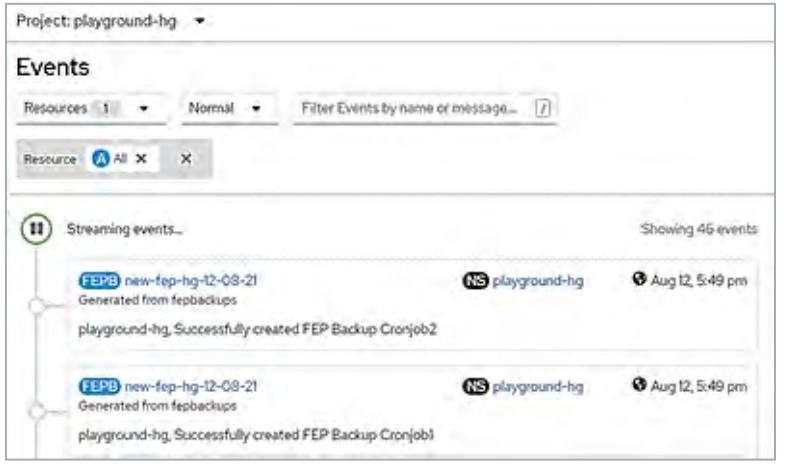
  - Alert rules are configurable
    - Alert levels, intervals, and thresholds can be set for any monitoring item
- Alert notification method
  - Set up notifications in Prometheus.
  - Alerts are sent via Alert Manager.
  - Integration available with email, SMS, Slack, and other systems.

Deep Insight - Event notifications

- When you create Fujitsu Enterprise Postgres resources, including clusters, you can see how custom resources are being created by integrating them with OpenShift standard event functionality.
- Use case
  - Visualize status of custom resource creation process
  - Quickly identify abnormal processing and facilitate debugging
- How to check events

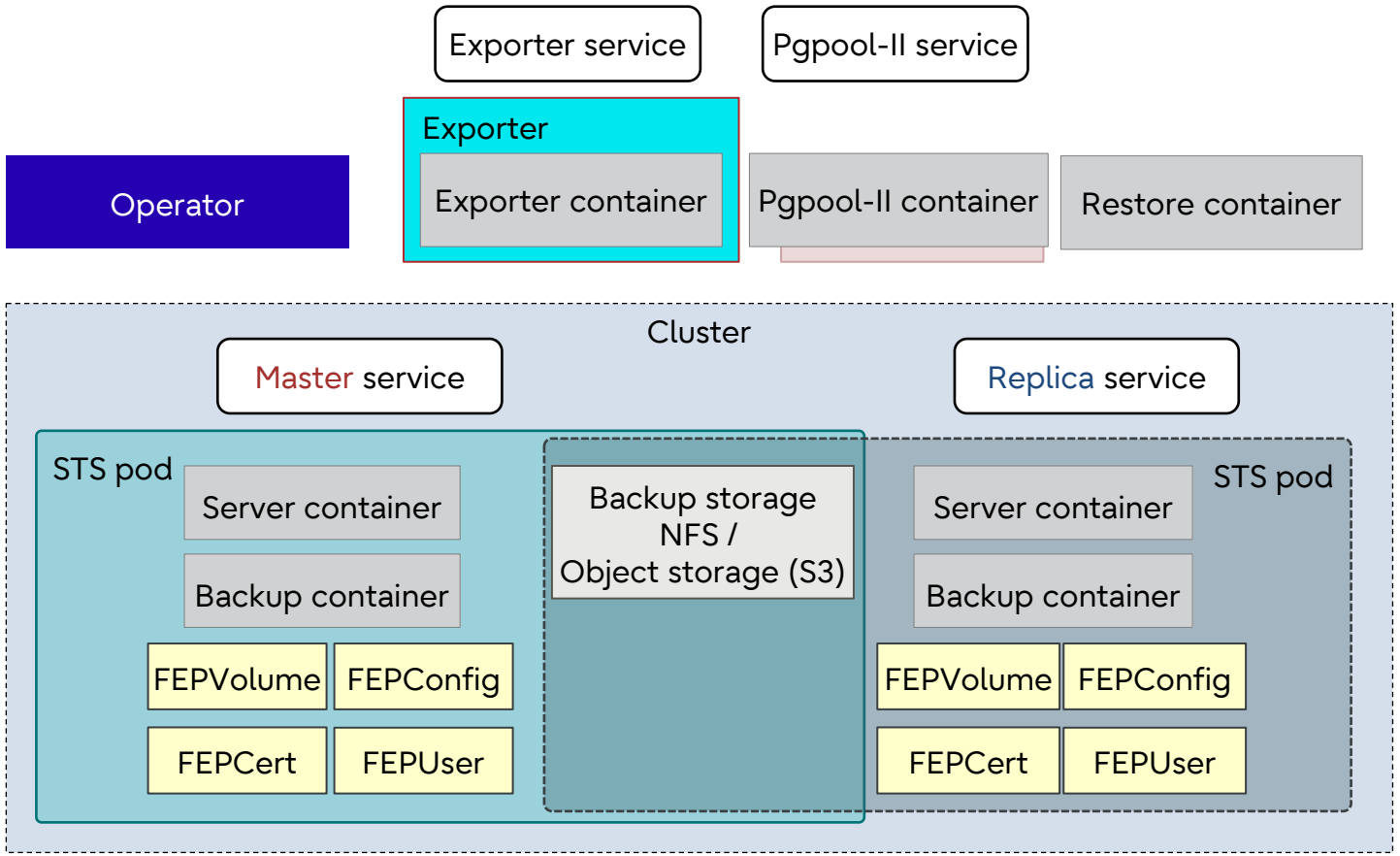
CLI (oc get events)

GUI




FUJITSU
Appendix

Operator system configuration



Operator	Container that accepts user requests and automates database construction and operations.
Server container	Container for the Fujitsu Enterprise Postgres server.
Backup container	Container created on the same pod as the server container to perform scheduled backup.
Restore container	Container temporarily created to perform restore.
Pgpool-II container	Container that uses Pgpool-II to provide load balancing and connection pooling. *(optional)*
Exporter container	Container that exposes http/https endpoint for monitoring stats scraping.
Backup storage	Storage for backup data. Not required if you do need to obtain backups.
Cluster	Parent CR for cluster definition and configuration.
FEPVolume	Child CR for volumes.
FEPConfig	Child CR for database configuration.
FEPCert	Child CR for system certificates.
FEPUser	Child CR for database users.
FEPExporter	CR for monitoring configuration.
Master service	Service to connect to the master server.
Replica service	Service to connect to the replica server.
Pgpool-II service	Service for connecting to Pgpool-II.
Exporter service	Service to scrape metrics from all cluster nodes.

Features and benefits

FUJITSU

Deployment	
Data centre	Within one data center
Model	• One master, two replica • One master
Replication Type	• Asynchronous • Synchronous • Logical replication
Scaling	• CPU • Memory • Number of connections

Load Balancing
Using Pgpool-II

High Availability	
Failover Type	Automatic
switchover type	Manual
Auto recovery	Automatic

Backup and Restore	
Frequency	Configurable
Generation	Configurable
Backup type	Full, incremental
Restore Type	Latest, PITR
Restore to	New cluster, existing cluster

Upgrade	
Supported	Minor version, Major version
Upgrade Type	Rolling update

Configuration Change
Dynamic configuration change

Monitoring & Deep Insights	
Monitoring	• Operator metrics • Operands (i.e., FEPCluster) metrics
Alert	Alert by metrics information
Event Notification	CR creation events

Scalability	
Scale-out read replica	Manual, Automatic
Scale-in read replica	Manual

Fujitsu Enteprise Postgres features

- Transparent Data Encryption
- Data Masking
- Dedicated Audit Log
- Vertical Clustered Index
- Global Meta Cache

Scope of Fujitsu Enterprise Postgres support

FUJITSU

- In addition to OSS PostgreSQL features, Fujitsu Enterprise Postgres clusters support the features below

Category	Feature
Operation	Global Meta Cache
Data security	Transparent Data Encryption
	Data Masking
	Dedicated Audit Log
High performance	Vertical Clustered Index
	High-speed data load
Application interface	Java integration
	ODBC integration
	.NET framework integration
	Embedded SQL integration (C language)
	Embedded SQL integration (COBOL)

Supported platforms

FUJITSU

Supported platforms:

The Operator is tested on the following platforms

Service	Platform
Self-Managed Kubernetes Service	• Red Hat OpenShift Container Platform 4.11 / 4.12 / 4.13 • Rancher Kubernetes Engine (on Linux hosts) * • SUSE Rancher 2.6
Fully Managed Kubernetes Service	• Red Hat OpenShift Service on AWS (ROSA) • Red Hat OpenShift on IBM Cloud • Azure Red Hat OpenShift (ARO) • Azure Kubernetes Service (AKS) * • Amazon Elastic Kubernetes Service (EKS) * • IBM Cloud Kubernetes Service * • Alibaba Cloud Container Service for Kubernetes (ACK) * • Google Kubernetes Engine (GKE) *

*: Kubernetes 1.23, 1.24, 1.25, 1.26 support

Supported storage :

Category	Storage
Type/interface	• Container Storage Interface • NFS • Red Hat OpenShift Container Storage
Cloud service	• Azure Blob Storage • Amazon S3

CPU : x86, s390x, ppc64le

Components embedded:

Component	Version	Description
Red Hat UBI minimal	8	Base OS image for the container
Fujitsu Enterprise Postgres Server	14.9 (x86, s390x) 14.7 (ppc64le)	Database server capabilities

Pre-requisite for Kubernetes

FUJITSU

Collaboration tools: Integration with the monitoring and alerting tools below is supported:

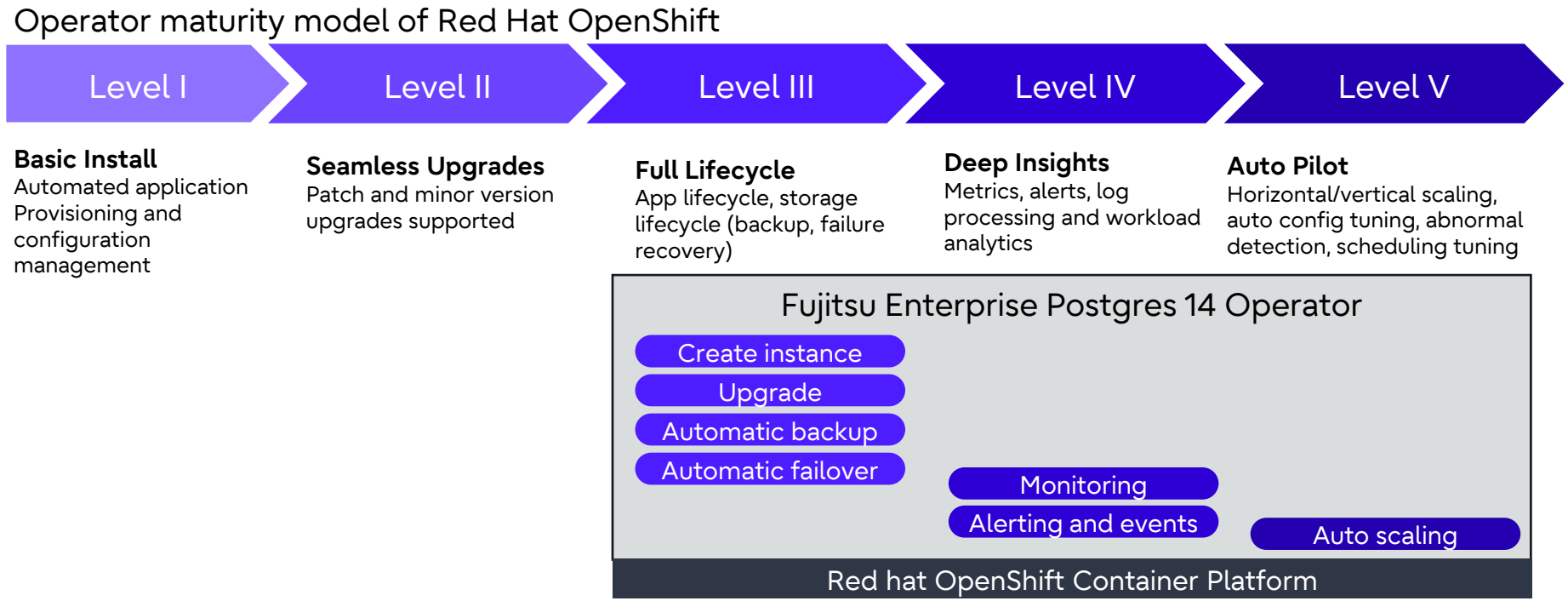
Tool		Version	How to obtain
Prometheus	OpenShift	Installed version of OpenShift	Pre-installed with OpenShift
AlertManager	Kubernetes	v0.52.1 and later	prometheus-operator
	Rancher	Latest version that can be installed with Rancher	rancher-monitoring application
Grafana	OpenShift	3.10.3 and later	Provided by OperatorHub
	Kubernetes	3.10.3 and later	grafana-operator
	Rancher	Latest version that can be installed with Rancher	rancher-monitoring application

Integration with Fujitsu Enterprise Postgres Operator management on Kubernetes tools below is supported.

Tool	Version	How to obtain
Helm	3.7.2 and later	Helm Web Site
Rancher	v2.6.2 and later	Rancher Web Site

Assured quality and compatibility

- Red Hat OpenShift Operator Certification
  - Level V certified (as of October 2021)
  - Asia's first commercial middleware provider
  - World's first multi-architecture operator certification



https://sdk.operatorframework.io/docs/overview/operator-capabilities/

Fujitsu Enterprise Postgres for Kubernetes

For more, visit our website at

fast.fujitsu.com/fujitsu-enterprise-postgres-for-kubernetes

FUJITSU

Published: 14-06-23 WW EN