

FUJITSU Software Symfoware Server V12.1.0





Windows

B1WS-1137-02Z0(00) 2014年12月

まえがき

本書の目的

Symfoware Serverは、PostgreSQLの機能を拡張し、Windowsプラットフォームで動作するデータベースシステムです。 本書は、Symfoware Serverの運用ガイドです。

本書の読者

本書は、Symfoware Serverを導入、運用される方を対象としています。

なお、本書は、以下についての一般的な知識があることを前提に書かれています。

- PostgreSQL
- SQL
- · Windows

本書の構成

本書の構成と内容は以下のとおりです。

第1章 Symfoware Serverの運用

Symfoware Serverの運用について説明しています。

第2章 インスタンスの起動とデータベースの作成

Symfoware Serverのインスタンスの起動とデータベースの作成について説明しています。

第3章 データベースのバックアップ

データベースのバックアップについて説明しています。

第4章 Secure Sockets Layerによる安全な通信の構成

クライアントとサーバ間の通信データの暗号化について説明しています。

第5章 透過的データ暗号化による格納データの保護

データベースに格納するデータの暗号化について説明しています。

第6章 定期的な運用操作

Symfoware Serverの定期的なデータベースの運用について説明しています。

第7章 PL/extJavaの導入と運用

PL/extJavaの導入と運用について説明しています。

第8章 異常時の対処

ディスク障害やデータ破壊が発生した場合のリカバリについて説明しています。

付録A パラメータ

Symfoware Serverのパラメータについて説明しています。

付録B システム管理関数

Symfoware Serverのシステム管理関数について説明しています。

付録C システムビュー

Symfoware Serverのシステムビューについて説明しています。

付録D WebAdminのWebサーバ機能の起動と停止

WebAdmin(Webサーバ機能)の起動と停止について説明しています。

付録E 障害調査情報の採取

初期調査のための情報の採取方法について説明しています。

付録F PL/extJavaの注意事項

PL/extJavaを使用する場合の注意事項について説明しています。

付録G PL/extJavaのログ情報

PL/extJavaが出力するログ情報について説明しています。

輸出管理規制について

本ドキュメントを輸出または第三者へ提供する場合は、お客様が居住する国および米国輸出管理関連法規等の規制をご確認のうえ、 必要な手続きをおとりください。

出版年月および版数

平成26年 12月 第2版		
平成25年 11月 初版		

著作権

Copyright 2013-2014 FUJITSU LIMITED

<u>目 次</u>

第1章 Symfoware Serverの運用	1
1.1 運用方法の種類	1
1.2 WebAdminの起動方法	
1.2.1 WebAdmin画面の流れ	2
1.2.2 WebAdminへのログイン	4
1.3 pgAdminの起動方法	6
1.3.1 pgAdminの起動	6
1.3.2 接続するインスタンスの追加	7
1.3.3 インスタンスへの接続と切断	9
1.4 コマンドによる運用操作	
1.5 Symfoware Serverの動作環境	11
1.5.1 動作環境	
1.5.2 ファイル構成	
1.6 運用に利用するアプリケーションの互換に関する注意事項	
第2章 インスタンスの起動とデータベースの作成	15
2.1 インスタンスの起動と停止	
2.1.1 WebAdminを使用する場合	
2.1.2 コマンドを利用する場合	
2.2 データベースの作成	
2.2.1 pgAdminを使用する場合	19
2.2.2 クライアントコマンドを使用する場合	
第3章 データベースのバックアップ	
3.1 定期的なバックアップ	
3.2 バックアップ方法	
3.2.1 WebAdminを使用する場合	
3.2.2 サーバコマンドを使用する場合	26
	20
第4章 Secure Sockets Layerによる安全な通信の構成	
第4章 Secure Sockets Layerによる安全な通信の構成	
 第4章 Secure Sockets Layerによる安全な通信の構成 4.1 通信データの暗号化のセットアップ 4.1.1 証明書発行の手続き 	
 第4章 Secure Sockets Layerによる安全な通信の構成 4.1 通信データの暗号化のセットアップ 4.1.1 証明書発行の手続き	
 第4章 Secure Sockets Layerによる安全な通信の構成	
 第4章 Secure Sockets Layerによる安全な通信の構成 4.1 通信データの暗号化のセットアップ 4.1.1 証明書発行の手続き	30 30 31 31 31 31 31 31 31
 第4章 Secure Sockets Layerによる安全な通信の構成. 4.1 通信データの暗号化のセットアップ. 4.1.1 証明書発行の手続き	30 30 31 31 31 31 31 31 31 31 31
 第4章 Secure Sockets Layerによる安全な通信の構成. 4.1 通信データの暗号化のセットアップ. 4.1.1 証明書発行の手続き. 4.1.2 サーバ証明書ファイル、サーバ秘密鍵ファイルの配置. 4.1.3 CA証明書ファイルのクライアントへの配布. 4.1.4 データベースサーバの動作環境の設定. 4.1.5 クライアントの動作環境の設定. 4.1.6 データベース多重化運用を行う場合. 	30 30 31 31 31 31 31 31 31 32
 第4章 Secure Sockets Layerによる安全な通信の構成. 4.1 通信データの暗号化のセットアップ 4.1.1 証明書発行の手続き. 4.1.2 サーバ証明書ファイル、サーバ秘密鍵ファイルの配置 4.1.3 CA証明書ファイルのクライアントへの配布. 4.1.4 データベースサーバの動作環境の設定 4.1.5 クライアントの動作環境の設定 4.1.6 データベース多重化運用を行う場合. 第5章 透過的データ暗号化による格納データの保護 	
 第4章 Secure Sockets Layerによる安全な通信の構成. 4.1 通信データの暗号化のセットアップ. 4.1.1 証明書発行の手続き. 4.1.2 サーバ証明書ファイル、サーバ秘密鍵ファイルの配置. 4.1.3 CA証明書ファイルのクライアントへの配布. 4.1.4 データベースサーバの動作環境の設定. 4.1.5 クライアントの動作環境の設定. 4.1.6 データベース多重化運用を行う場合. 第5章 透過的データ暗号化による格納データの保護. 5.1 暗号化によるデータの保護.	
 第4章 Secure Sockets Layerによる安全な通信の構成. 4.1 通信データの暗号化のセットアップ. 4.1.1 証明書発行の手続き. 4.1.2 サーバ証明書ファイル、サーバ秘密鍵ファイルの配置. 4.1.3 CA証明書ファイルのクライアントへの配布. 4.1.4 データベースサーバの動作環境の設定. 4.1.5 クライアントの動作環境の設定. 4.1.6 データベース多重化運用を行う場合. 第5章 透過的データ暗号化による格納データの保護. 5.1 暗号化によるデータの保護. 5.2 マスタ暗号化キーの設定.	
 第4章 Secure Sockets Layerによる安全な通信の構成. 4.1 通信データの暗号化のセットアップ. 4.1.1 証明書発行の手続き. 4.1.2 サーバ証明書ファイル、サーバ秘密鍵ファイルの配置. 4.1.3 CA証明書ファイルのクライアントへの配布. 4.1.4 データベースサーバの動作環境の設定. 4.1.5 クライアントの動作環境の設定. 4.1.6 データベース多重化運用を行う場合. 第5章 透過的データ暗号化による格納データの保護. 5.1 暗号化によるデータの保護. 5.2 マスタ暗号化キーの設定. 5.3 キーストアのオープン.	30 30 31 31 31 31 31 31 31 32 33 33 34 34 35
 第4章 Secure Sockets Layerによる安全な通信の構成. 4.1 通信データの暗号化のセットアップ. 4.1.1 証明書発行の手続き 4.1.2 サーバ証明書ファイル、サーバ秘密鍵ファイルの配置 4.1.3 CA証明書ファイルのクライアントへの配布. 4.1.4 データベースサーバの動作環境の設定 4.1.5 クライアントの動作環境の設定 4.1.6 データベース多重化運用を行う場合. 第5章 透過的データ暗号化による格納データの保護 5.1 暗号化によるデータの保護 5.2 マスタ暗号化キーの設定 5.3 キーストアのオープン 5.4 テーブル空間の暗号化	
 第4章 Secure Sockets Layerによる安全な通信の構成. 4.1 通信データの暗号化のセットアップ. 4.1.1 証明書発行の手続き. 4.1.2 サーバ証明書ファイル、サーバ秘密鍵ファイルの配置. 4.1.3 CA証明書ファイルのクライアントへの配布. 4.1.4 データベースサーバの動作環境の設定. 4.1.5 クライアントの動作環境の設定. 4.1.6 データベース多重化運用を行う場合. 第5章 透過的データ暗号化による格納データの保護. 5.1 暗号化によるデータの保護. 5.2 マスタ暗号化キーの設定. 5.3 キーストアのオープン. 5.4 テーブル空間の暗号化. 5.5 暗号化されているテーブル空間の確認.	30 30 31 31 31 31 31 31 32 33 33 33 34 34 35 36 37
 第4章 Secure Sockets Layerによる安全な通信の構成. 4.1 通信データの暗号化のセットアップ. 4.1.1 証明書発行の手続き. 4.1.2 サーバ証明書ファイル、サーバ秘密鍵ファイルの配置. 4.1.3 CA証明書ファイルのクライアントへの配布. 4.1.4 データベースサーバの動作環境の設定. 4.1.5 クライアントの動作環境の設定. 4.1.6 データベース多重化運用を行う場合. 第5章 透過的データ暗号化による格納データの保護. 5.1 暗号化によるデータの保護. 5.2 マスタ暗号化キーの設定. 5.3 キーストアのオープン. 5.4 テーブル空間の暗号化. 5.5 暗号化されているテーブル空間の確認. 5.6 キーストアの管理.	30 30 31 31 31 31 31 31 32 33 33 33 33 33 34 34 35 36 37 37
 第4章 Secure Sockets Layerによる安全な通信の構成	30 30 31 31 31 31 31 31 32 33 33 33 33 33 33 34 34 35 36 37 37 37 37
 第4章 Secure Sockets Layerによる安全な通信の構成	30 30 31 31 31 31 31 31 31 32 33 33 33 34 34 35 36 37 37 37 37 37 38
 第4章 Secure Sockets Layerによる安全な通信の構成. 4.1 通信データの暗号化のセットアップ. 4.1.1 証明書発行の手続き. 4.1.2 サーバ証明書ファイル、サーバ秘密鍵ファイルの配置. 4.1.3 CA証明書ファイル、サーバ秘密鍵ファイルの配置. 4.1.4 データベースサーバの動作環境の設定. 4.1.6 データベースサーバの動作環境の設定. 4.1.6 データベース多重化運用を行う場合. 第5章 透過的データ暗号化による格納データの保護. 5.1 暗号化によるデータの保護. 5.2 マスタ暗号化キーの設定. 5.3 キーストアの常理 5.6.3 キーストアの自動オープンの有効化.	30 30 31 31 31 31 31 31 32 33 33 33 34 34 35 36 37 37 37 37 37 38 38
 第4章 Secure Sockets Layerによる安全な通信の構成 4.1 通信データの暗号化のセットアップ 4.1.1 証明書発行の手続き 4.1.2 サーバ証明書ファイル、サーバ秘密鍵ファイルの配置 4.1.3 CA証明書ファイルのクライアントへの配布 4.1.4 データベースサーバの動作環境の設定 4.1.5 クライアントの動作環境の設定 4.1.6 データベース多重化運用を行う場合 第5章 透過的データ暗号化による格納データの保護 5.1 暗号化によるデータの保護 5.2 マスタ暗号化キーの設定 5.5 暗号化されているテーブル空間の確認 5.6 キーストアの常理 5.6.1 マスダ暗号化キーの変更 5.6.2 キーストアのパスフレーズの変更 5.6.3 キーストアのパスフレーズの変更 5.6.4 キーストアのバックアップとリカバリ 	30 30 31 31 31 31 31 32 33 33 33 34 34 35 36 37 37 37 37 37 37 37 38 38 39
第4章 Secure Sockets Layerによる安全な通信の構成. 4.1 通信データの暗号化のセットアップ. 4.1.1 証明書発行の手続き. 4.1.2 サーバ証明書ファイル、サーバ秘密鍵ファイルの配置. 4.1.3 CA証明書ファイルのクライアントへの配布. 4.1.4 データベースサーバの動作環境の設定. 4.1.5 クライアントの動作環境の設定. 4.1.6 データベース多重化運用を行う場合. 第5章 透過的データ暗号化による格納データの保護. 5.1 暗号化によるデータの保護. 5.2 マスタ暗号化キーの設定. 5.3 キーストアのオープン. 5.4 テーブル空間の暗号化. 5.5 暗号化されているテーブル空間の確認. 5.6 キーストアの管理. 5.6.1 マスタ暗号化キーの変更. 5.6.2 キーストアのパスフレーズの変更. 5.6.3 キーストアの自動オープンの有効化. 5.6.4 キーストアのバックアップとリストアノリカバリ.	30 30 31 31 31 31 31 31 32 33 33 33 33 34 34 35 35 36 37 37 37 37 37 37 37 38 38 39 41
第4章 Secure Sockets Layerによる安全な通信の構成. 4.1 通信データの暗号化のセットアップ. 4.1.1 証明書発行の手続き. 4.1.2 サーバ証明書ファイル、サーバ秘密鍵ファイルの配置. 4.1.3 CA証明書ファイルのクライアントへの配布. 4.1.4 データベースサーバの動作環境の設定. 4.1.5 クライアントの動作環境の設定. 4.1.6 データベース多重化運用を行う場合. 第5章 透過的データ暗号化による格納データの保護. 5.1 暗号化によるデータの保護. 5.2 マスタ暗号化キーの設定. 5.3 キーストアのオープン. 5.4 テーブル空間の暗号化. 5.5 暗号化されているテーブル空間の確認. 5.6 キーストアの常理. 5.6.1 マスタ暗号化キーの変更. 5.6.2 キーストアのパスフレーズの変更. 5.6.3 キーストアの自動オープンの有効化. 5.6 キーストアのバックアップとりカバリ. 5.7 データベースのバックアップとりカバリ. 5.8 データベースのバックアップとりカバリ. 5.8 データベースのインポートとエクスポート.	30 30 30 30 31 32 33 34 35 36 37 37 38 38 39 41 43 30 31 32 33 34 35 36 37 38
 第4章 Secure Sockets Layerによる安全な通信の構成. 4.1 通信データの暗号化のセットアップ. 4.1.1 証明書発行の手続き. 4.1.2 サーバ証明書ファイル、サーバ秘密鍵ファイルの配置. 4.1.3 CA証明書ファイル、サーバ秘密鍵ファイルの配置. 4.1.4 データベースサーバの動作環境の設定. 4.1.5 クライアントの動作環境の設定. 4.1.6 データベース多重化運用を行う場合. 第5章 透過的データ暗号化による格納データの保護. 5.1 暗号化によるデータの保護. 5.2 マスタ暗号化キーの設定. 5.3 キーストアのオープン. 5.6 キーストアの管理. 5.6.1 マスタ暗号化キーの変更. 5.6.2 キーストアのパスフレーズの変更. 5.6.3 キーストアの自動オープンの有効化. 5.6.4 キーストアのパックアップとリストア/リカバリ. 5.7 データベースのバックアップとリストア/リカバリ. 5.8 データベースのバックアップとリストア/リカバリ. 5.9 既存データの暗号化.	30 30 30 31 32 33 34 35 36 37 37 38 38 39 41 43 44
第4章 Secure Sockets Layerによる安全な通信の構成. 4.1 通信データの暗号化のセットアップ 4.1.1 証明書発行の手続き. 4.1.2 サーバ証明書ファイル、サーバ秘密鍵ファイルの配置 4.1.3 CA証明書ファイルのクライアントへの配布 4.1.4 データベースサーバの動作環境の設定. 4.1.5 クライアントの動作環境の設定. 4.1.6 データベース多重化運用を行う場合 第5章 透過的データ暗号化による格納データの保護. 5.1 暗号化によるデータの保護. 5.2 マスタ暗号化キーの設定. 5.3 キーストアのオープン. 5.4 テーブル空間の暗号化. 5.5 暗号化されているテーブル空間の確認. 5.6 キーストアの学ーズの変更. 5.6.1 マスタ暗号化キーの変更. 5.6.2 キーストアのパスフレーズの変更. 5.6.3 キーストアのパスフレーズの変更. 5.6.4 キーストアのパスフレーズの変更. 5.6.5 キーストアの自動オープンの有効化. 5.6 キーストアのパックアップとリカバリ. 5.7 データベースのバックアップとリカバリ. 5.8 データベースのインポートとエクスポート. 5.9 既存データの暗号化. 5.10 クラスタシステムにおける運用.	30 30 30 31 32 33 34 35 36 37 38 38 39 41 42 44
第4章 Secure Sockets Layerによる安全な通信の構成. 4.1 通信データの暗号化のセットアップ. 4.1.1 証明書発行の手続き. 4.1.2 サーバ証明書ファイル、サーバ秘密鍵ファイルの配置. 4.1.3 CA証明書ファイル、サーバ秘密鍵ファイルの配置. 4.1.4 データベースサーバの動作環境の設定. 4.1.5 クライアントの動作環境の設定. 4.1.6 データベース多重化運用を行う場合. 第5章 透過的データ暗号化による格納データの保護. 5.1 暗号化によるデータの保護. 5.2 マスタ暗号化キーの設定. 5.3 キーストアのオープン. 5.4 デーブル空間の暗号化. 5.5 暗号化されているテーブル空間の確認. 5.6 キーストアの管理. 5.6.1 マスタ暗号化キーの変更. 5.6.2 キーストアの管理. 5.6.3 キーストアの自動オープンの有効化. 5.6.4 キーストアのバックアップとリストア/リカバリ. 5.7 データベースのインポートとエクスポート. 5.9 既存データの暗号化. 5.10 クラスタシステムにおける運用. 5.10.1 データベース多重化運用を用いないHAクラスタ.	30 30 30 31 32 33 33 34 35 36 37 37 38 38 39 41 44 44 44 44

5.11 セキュリティに関する注意事項	
5.12 構築済みアプリケーションの導入のヒント	
第6音 京期的な運用過作	10
50年 に効りな建用休日	40
0.1 ビノ い設定 2 温 尻	
0.2 / イバンの反用重の監視と上で限場の確保	
0.2.1 / イハノの反用重の量況 6.2.2 ディスクの穴や領域の確保	40 /10
0.2.2 / 1ハ/の上で頃吸り確応	
0.5 ニートノンヨン の日勤 90時	
6.41ビュー(ng stat activity)を使用する方注	50
0.4.1 Cユ (pg_stat_activity)を使用する方法	51
0.4.2 pgAumine 反用する方法	52
0.5 イン / / / / / / / / / / / / / / / / / /	53
6.61 取得できる情報	53
6.62 取得するための設定	55
663 情報のリセット方法	55
第7章 PL/extJavaの導入と運用	
7.1 PL/extJavaの概要	
7.1.1 PL/extJavaの構成	
7.1.2 使用できるアプリケーションサーバ	
7.1.3 ユーザー定義	61
7.2 PL/extJavaの導入	61
7.2.1 ポート番号の準備	61
7.2.2ドメインの作成	
7.2.3 PL/extJavaの作成	
7.2.3.1 データベースクラスタの設定	
7.2.3.2 コンテナの作成	
7.2.4 Java関数の登録	
7.3 PL/extJavaの運用	
7.3.1 コンテナの起動・停止	
7.3.2 PL/extJavaの確認	
7.3.2.1 ドメイン情報の確認	
7.3.2.2 コンテナ情報の確認	
7.3.3 PL/extJavaの変更	
7.3.3.1 サーバインスタンス(JavaVM)の追加と削除	
7.3.3.2 データベース接続情報の変更	
7.3.3.3 ポート番号の変更	
7.3.4 Java関数の削除	
7.3.5 コンテナの削除	
7.3.6ドメインの削除	
7.3.7 バックアップ / リストア	
7.3.7.1 バックアップ方法	
7.3.7.2 リストア方法	
	75
第0早 共吊时の対処	
0.1 / イベク障害(ハードリエ) // つのクタルハリ 8.1.1 Woh Adminを使用する場合	
8.1.1 WebAuminで使用りる場合 8.1.2 サーバコマンバン体田ナズ担合	/0
0.1.2 り ハーマントを使用りる物ローーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーーー	
6.2 / ク観家からのリルバリー 8.2.1 WohAdminを使用する場合	
0.2.1 WCDAUIIIIIで区用する場合	۵۵ مح
o.2.2 pgx_lcvall-*メービ化用りつ物口 2 2 ユーザーの調晶作動にのIIカバII	۵۵ ۵۲
0.5 ー 9 9000000000000000000000000000000000	80 סא רס
0.5.1 webAullillで使用する物ローーーー 8.3.1 mebAullillで使用する物ローーーー	۸/ ۸/ ۸/ ۸/ ۸/ ۸/ ۸/ ۸/ ۸/ ۸/ ۸/ ۸/ ۸/ ۸
o.5.2 pgx_lcvall- * < lで (欠用 り 3 切口 & A アプリケーション 思 堂 の 対加	
0.7 / / / / マコイ 共田シバルだ	
uniter (pg_stat_activity)で区内する第日	

D 1 Web A dmin @ Web ++	· 12270-2-2010-17-12	
↓録D WebAdminのWebサー	バ機能の起動と停止	
C.5 pgx_stat_sql		
C.4 pgx_stat_walwriter		
C.3 pgx_stat_latch		
C.2 pgx_stat_lwlock		
C.1 pgx_tablespaces		
す録C システムビュー		
B.2 透過的データ暗号化制行	即関数	
B.1 WAL二重化制御関数		
け録Β システム管理関数		
「球A ハフメータ		
8.15.2 Java 関数の無応答		
8.15.1 Java関数の異常		
8.15 アプリケーションの運用	こ関する異常(PL/extJava運用時)	
8.14 PL/extJavaの運用操作の)異常	
8.13.2 電源未投入やマウン	/トによる異常	
8.13.1 外部ディスクとの間	のネットワーク異常	
8.13 ディスク障害以外の入出	力異常	
8.12 分散トランザクションの昇	常時の対処	
8.11.3 PL/extJavaの停止失	敗による異常	
8.11.2.3 サーバプロセス	の強制停止	
8.11.2.2 Immediateモー	ドによる停止	
8.11.2.1 Fastモードによ	5停止	
8.11.2 サーバコマンドを使	用する場合	
8.11.1 WebAdminを使用す	る場合	
8.11 インスタンス停止失敗時	の対処	
8.10.4.2 サーバコマンド	を使用する場合	
8.10.4.1 WebAdminを使	用する場合	
8.10.4 その他の異常		
8.10.3 PL/extJavaの起動生	敗による異常	
8.10.2 電源未投入やマウン	/トによる異常	
8.10.1 設定ファイルの誤り		
8.10 インスタンス起動失敗時	の対処	
8.9 各格納先ディスクの異常		
8.8.1.2 サーバコマンドを	使用する場合	
8.8.1.1 WebAdminを使	月する場合	
8.8.1 容量の大きいディスク	パにディスク交換する方法	
8.8トランザクションログ格納	もの容量不足時の対処	
8.7.2.2 サーバコマンドを	使用する場合	
8.7.2.1 WebAdminを使	月する場合	
8.7.2 容量の大きいディスク	パにディスク交換する方法	
8.7.1.2 サーバコマンドを	使用する場合	
8.7.1.1 WebAdminを使	月する場合	
8.7.1 バックアップデータを	一時退避する方法	
8.7 バックアップデータ格納分	5の容量不足時の対処	
8.6.2.2 サーバコマンドを	使用する場合	
8.6.2.1 WebAdminを使	月する場合	
8.6.2 容量の大きいディスク	パニディスク交換する方法	
8.6.1 テーブル空間を使用	する方法	
86データ格納先の容量不長	時の対処	

付録E 障害調査情報の採取	
付録F PL/extJavaの注意事項	
F.1 スレッドダンプツールの使い方	
F.1.1 thdumpコマンドの使用方法	
F.1.2 thdumpSVCコマンドの使用方法	
F.1.3 オプション	
F.2 jstackを利用する場合の注意	
付録G PL/extJavaのログ情報	
G.1ドメイン	
G.1.1 サーバログ	
G.1.2 JavaVMログ	
G.2 コンテナ	
G.2.1 サーバログ	
G.2.2 JavaVMログ	
G.2.3 HTTPアクセスログ	
G.2.4 HTTPトレースログ	
G.3 Webサーバ	
G.3.1 エラーログ	
G.3.2トレースログ	
G.3.3 アクセスログ	
G.3.4 内部ログ	
索引	

第1章 Symfoware Serverの運用

Symfoware Serverの運用について説明します。

1.1 運用方法の種類

Symfoware Serverの運用管理には、以下の2つの方法があります。

- ・ GUIツールを使用した運用管理
- ・ コマンドを使用した運用管理

💦 参照

データベース多重化機能を利用してスイッチオーバ、またはフェールオーバ運用を行う場合は、"クラスタ運用ガイド"の"データベー ス多重化運用"を参照してください。

.

GUIツールを使用した運用管理

GUIツールによる運用管理の方法には、WebAdminとpgAdminがあります。

WebAdminによる管理

従来、データベースを運用する際に必要不可欠であった、煩雑な環境設定やバックアップ/リカバリの複雑な運用設計を行う必要がありません。データベースの専門知識がなくても、簡単に、確実に、データベースの状態監視、バックアップ、リストアを運用することができます。

pgAdminによる管理

アプリケーション開発やデータベースの保守を行うにあたり、pgAdminを使用して、以下のようなデータベースオブジェクトに対する簡単な操作を行うことができます。

- インデックスの再構築や統計情報の更新など
- データベースオブジェクトの作成、削除および更新

また、Symfoware ServerのpgAdminからは、PostgreSQLのSQLコマンドに対するSymfoware Serverによる拡張機能を利用することができます。

💦 参照

Symfoware ServerによるpgAdminの拡張機能については、"pgAdminヘルプ"を参照してください。

コマンドを使用した高度な運用管理

コマンドを使用してデータベースの詳細な設定や運用操作、運用管理ができます。コマンドを使用して運用管理を進めた場合、 WebAdminによる運用管理に移行できません。

.



• WebAdminとサーバコマンドを以下のように組み合わせて運用することはできません。

- initdbコマンドで作成したインスタンスのWebAdminによる運用

- WebAdminで作成したインスタンスのコマンドによる運用

- WebAdminでバックアップしたデータベースのコマンドによるリカバリ

ただし、WebAdminで作成したインスタンスにおいてpgx_dmpallコマンドによりバックアップを取得することが可能です。また、pgx_dmpallコマンドで取得したバックアップを使用してWebAdminでリカバリすることが可能です。

・ pgAdminにおいても、バックアップとリストアを行うことができますが、WebAdminおよびpgx_dumpallで取得したバックアップデータと、pgAdminで取得したバックアップデータとは、相互に互換性がありません。

・ pgAdminのその他の注意事項は、"pgAdminヘルプ"を参照してください。

各フェーズで利用する機能

GUIによる運用とコマンドによる運用のそれぞれについて、各フェーズで利用する機能を以下に示します。

	運用	GUIによる運用	コマンドによる運用
セットアップ	インスタンスの作成	WebAdmin	initdbコマンド
	設定ファイルの変更	WebAdmin	設定ファイルを直接編集
インスタンスの起動		WebAdmin	OSのnetコマンドまたはscコマン ド
データベースの作成		pgAdmin	DDL文で指定し、psqlやアプリ ケーションで定義
データベースのバック	アップ	WebAdmin	pgx_dmpallコマンド
		pgx_dmpallコマンド	
モニタリング	データベースの異常	WebAdmin (注1)	イベントログに出力されるメッセー ジ (注1)
	ディスク容量	WebAdmin (注1)(注2)	OSのfsutilコマンド(空き容量の確 認)、およびdirコマンド(使用容量 の確認)
	コネクション状況	pgAdmin	psqlコマンド(注3)
データベースのリカバ		WebAdmin	pgx_rcvallコマンド

注1) 運用管理ミドルウェア(Systemwalker Centric Managerなど)による監視が可能です。

注2) ディスクの使用量が全体の80%に到達すると、警告を表示します。

注3)標準統計情報ビューのpg_stat_activityを検索して状況を監視します。

1.2 WebAdminの起動方法

WebAdminの起動方法とログイン方法について説明します。

1.2.1 WebAdmin画面の流れ

WebAdminのGUI画面の流れについて以下の図に示します。



[監視]メニュー

以下に示すインスタンスの操作、および状態表示などを行うことができます。

- ・ 新規作成:データベースクラスタの生成とインスタンスの作成を行います。
- ・ 削除:データベースクラスタおよびインスタンスを削除します。
- ・ 起動/停止:インスタンスの起動、およびインスタンスの停止を行います。
- ・ バックアップ:データベースクラスタのバックアップを行います。
- ・ データ復旧:データベースクラスタをバックアップ時点へリカバリします。
- ・メッセージ:WebAdminで実施した操作、および検出した異常などのメッセージを表示します。



[監視]メニューの各機能については、以下を参照してください。

- ・ 新規作成および削除: "導入ガイド(サーバ編)"の"インスタンスの作成"
- ・ 起動および停止: "2.1.1 WebAdminを使用する場合"
- バックアップ: "3.2.1 WebAdminを使用する場合"
- データ復旧: "8.3.1 WebAdminを使用する場合"

[設定]メニュー

以下に示すインスタンスの定義情報の設定を行うことができます。

- ・ 文字コード:文字コードとロケールを設定します。
- ・ クライアント認証:クライアントがインスタンスに接続するための認証情報を設定します。
- ・ 通信:アプリケーションとインスタンスの通信定義を設定します。
- ・ SQLオプション:SQL文を実行する際に関係する定義を設定します。
- ・ 使用メモリ:使用メモリを設定します。

💦 参照

[設定]メニューの詳細は、"導入ガイド(サーバ編)"の"設定内容の変更"を参照してください。

1.2.2 WebAdminへのログイン

WebAdminへのログイン方法を説明します。

WebAdminの利用環境

WebAdminを使用するには、以下のブラウザが必要です。

• Internet Explorer 8.0, 9.0, 10.0, 11.0

WebAdminの起動URL

ブラウザのURLにWebAdmin画面の起動URLを、以下の形式で指定します。

http://ホスト名またはIPアドレス:ポート番号/

- ・ホスト名またはIPアドレス:Symfoware Serverをインストールしたサーバのホスト名、またはIPアドレス
- ・ポート番号:WebAdminのポート番号。デフォルトのポート番号は26515です。



```
サーバのIPアドレスが"192.0.2.0"、ポート番号が"26515"の場合
```

http://192.0.2.0:26515/

以下の起動画面が表示されます。





.

・ WebAdminを利用するには、あらかじめWebAdminのWebサーバ機能を起動しておく必要があります。

・ WebAdminのWebサーバ機能の起動方法の詳細については、"付録D WebAdminのWebサーバ機能の起動と停止"を参照してく ださい。

WebAdminへのログイン

起動画面の"Symfoware WebAdmin"をクリックすると、WebAdminが起動し、[ログイン]画面が表示されます。[ログイン]画面から、 WebAdminにログインできます。

UJITSU Softwar Symfoware Serv		FL	JITS
ユーザーID: パスワード:			
		(ログイン	

ログイン時には、以下を指定します。

- ・「ユーザーID」:インスタンス管理者のユーザーID(OSのユーザーアカウント)
- ・「パスワード」:ユーザーIDのパスワード

関 ポイント

インスタンス管理者のユーザーIDはOSのユーザーアカウントを使用します。詳細は"導入ガイド(サーバ編)"の"インスタンス管理者ユー ザーの作成"を参照してください。

1.3 pgAdminの起動方法

pgAdminの起動方法と、データベースの管理に必要な接続先のインスタンスの追加、および接続と切断方法について説明します。 なお、pgAdminは、Windowsクライアントで利用できます。

1.3.1 pgAdminの起動

使用するpgAdminを含む製品が、「Symfoware Server Client(Openインタフェース) (AAbit) Vxx.y.z」であった場合の起動方法を示します。AAは32か64、xx.y.zはバージョン番号に合わせて読み替えてください。

Windows(R) 8、またはWindows Server(R) 2012の場合

[スタート]画面の[pgAdmin III(AAbit)(Vxx.y.z)]を起動します。

Windows(R) 8.1、またはWindows Server(R) 2012 R2の場合

[アプリ]ビューの[pgAdmin III(AAbit)(Vxx.y.z)]を起動します。

上記以外の場合

[スタート]メニューの[すべてのプログラム]の[Symfoware Server Client (Openインタフェース) (AAbit) Vxx.y.z]から[pgAdmin III(AAbit) (Vxx.y.z)]を起動します。

pgAdminを起動すると、以下の画面が表示されます。

👎 pgAdmin III	
ファイル(E) 編集(E) プラグイン(P) ビュー(V) ツ-	-ル(I) ヘルプ(H)
* C u ' \ 7 8	
オブジェクトブラウザ × ■ サーバウループ	プロパティ 統計情報 属性情報 依存関係 マロパティ プロパティ しいかなるプロパティも現在の選択で利用できません。 く ************************************
	۲ III F
サーバで詳細の回復完了	0.00 秒



- pgAdminを利用するには、あらかじめ接続するインスタンスを起動しておく必要があります。
- ・ インスタンスの起動方法の詳細については、"2.1 インスタンスの起動と停止"を参照してください。
- ・ pgAdminの[Symfoware Help]からマニュアルを参照する場合、Adobe(R) Reader(R)が必要です。

1.3.2 接続するインスタンスの追加

接続先のインスタンスの追加方法を説明します。

1. pgAdminの[ファイル]メニューの[サーバの追加]をクリックします。

2. [新しいサーバ登録]画面で各項目を指定します。

[新しいサーバ	登録 📃 🔀
プロパティ SSL	先進的
名前	db01
ホスト	sv1
Port	26500
サービス	
DBメンテナンス	postgres 👻
ユーザ名	symfo
パスワード	•••••
パスワード保存	
色づけ	
グループ	サーバ ・
ヘルプ	<u> OK</u> キャンセル(<u>C</u>)
	ł.

(プロパティタブ)

- 「名前」:管理するインスタンスの名前
- 「ホスト」: Symfoware Serverをインストールしたホスト名、またはIPアドレス
- 「Port」: インスタンスのポート番号
- 「ユーザ名」:インスタンス管理者のユーザーID
- 「パスワード」:「ユーザ名」に指定したユーザーIDのパスワード

pgAdminでインスタンスを追加すると、追加が完了した直後に自動的にインスタンスへ接続します。



「パスワード保存」をチェックした場合、以下の場所にSymfoware Serverに接続するためのパスワードを格納したファイルを作成します。パスワードファイルは適切なアクセス権限を設定し、不正なアクセスから保護してください。

1.3.3 インスタンスへの接続と切断

pgAdmin とインスタンスの接続と切断について説明します。



WebAdminの[クライアント認証]画面の詳細については、"導入ガイド(サーバ編)"の"設定内容の変更"を参照してください。

インスタンスへの接続

pgAdminを起動した場合、インスタンスとは切断状態です。

インスタンスと接続する場合は、[オブジェクトブラウザ]からインスタンスを選択し、右クリックで[接続]を選択します。



インスタンスの追加時にパスワード保存しなかった場合、以下のパスワード入力画面が表示されます。

🏄 サーバに接続	
ユーザ symfo のパスワードを入力してください	
5-710001 (5917)	
I	
□ パスワード保存	
ヘルプ <u>OK</u> キャンセル(<u>C</u>)	

インスタンスとの切断

インスタンスから切断する場合、pgAdmin画面の[オブジェクトブラウザ]でサーバを選択し、右クリックして[Disconnect server]を選択します。

1.4 コマンドによる運用操作

以下のコマンドを使用して、データベースを運用管理できます。

・ サーバコマンド

データベースクラスタの作成、データベースを制御するコマンドなどが含まれます。これらのコマンドは、データベースが稼働して いるサーバ上で実行できます。

コマンドを使用するには、環境変数の設定が必要です。



- サーバコマンドの詳細は、"PostgreSQL文書"の"リファレンス"の"PostgreSQLサーバアプリケーション"、または"リファレンス" を参照してください。

- 環境変数の設定内容については、"導入ガイド(サーバ編)"の"initdbコマンドを使用する場合"において、インスタンスの作成 手順の"環境変数の設定"を参照してください。

・ クライアントコマンド

psqlコマンド、データベースクラスタをスクリプトファイルへ抽出するコマンドなどが含まれます。これらのコマンドは、データベースに 接続できるクライアント上、またはデータベースが稼動しているサーバ上で実行できます。

コマンドを使用するには、環境変数の設定が必要です。



- クライアントコマンドの詳細は、"PostgreSQL文書"の"リファレンス"の"PostgreSQLクライアントアプリケーション"または、"リファレンス"を参照してください。
- 環境変数の設定内容については、"導入ガイド(クライアント編)"の"環境変数の設定"を参照してください。

1.5 Symfoware Serverの動作環境

Symfoware Serverの動作環境、およびファイル構成について説明します。

1.5.1 動作環境

Symfoware Serverの動作環境の構成を、以下の図に示します。また、OS資源、およびSymfoware Server資源の役割を、以下の各表 に示します。



注) I/0負荷分散したい場合は、トランザクションログをデータ格納先とは別のディスクに配置することもできます。

表1.1 OS資源

種類	役割
共用メモリ	データベースプロセスが外部との情報交換に使用します。

種類	役割
セマフォ	

<u>表1.2 Symfoware Serverクライアント資源</u>

種類	役割
接続サービスファイル	Symfoware Serverに接続するためのホスト名、ユーザーID、パスワードなどの情報を指定するファイルです。
パスワードファイル	Symfoware Serverに接続するためのパスワードを安全に管理するためのファイルです。
CA証明書ファイル	通信データを暗号化する場合、サーバの正当性を検証するために使用するCA(認証局)証明書です。

表1.3 Symfoware Serverサーバ資源

種類	役割
データベースクラスタ	データベース格納ディスク上にあるデータベース格納領域です。インス タンスで管理されるデータベースの集合体です。
システムカタログ	利用者が作成したデータベースの定義情報、運用情報を含め、システムが動作するうえで必要な情報を保持します。
デフォルトテーブル空間	デフォルトで格納されるテーブルファイル、インデックスファイルを保持します。
トランザクションログ	クラッシュリカバリ、ロールバックに備えたログ情報を保持します。 WAL(Write-Ahead Log)と同義です。
作業ファイル	アプリケーションの実行、コマンドの実行時に使用する作業ファイルで す。
postgresql.conf	Symfoware Serverの動作環境を規定する各種情報を保持します。
pg_hba.conf	Symfoware Serverがクライアントをクライアントホスト単位で認証するため に使用するファイルです。
サーバ証明書ファイル	通信データを暗号化し、サーバ認証を行う場合に、サーバ証明書情報 を保持します。
サーバ秘密鍵ファイル	通信データを暗号化し、サーバ認証を行う場合に、サーバ秘密鍵情報 を保持します。
テーブル空間	データベースクラスタとは別領域に、テーブルファイル、インデックスファ イルを保持します。
バックアップ	ディスク障害などの異常時に、データベースを復旧するために必要な データが格納されています。
データベースバックアップ	データベースのバックアップデータを保持します。
アーカイブログ	リカバリに備えたログ情報を保持します。
コアファイル	Symfoware Serverのプロセスで異常が発生した場合に出力する Symfoware Serverプロセスのコアファイルです。
キー管理サーバまたはキー管理スト レージ	マスタ暗号化キーファイルを配置するサーバまたはストレージです。
マスタ暗号化キーファイル	格納データを暗号化する場合に、マスタ暗号化キーを保持します。マス タ暗号化キーファイルは、キー管理サーバまたはキー管理ストレージ上 で管理します。

1.5.2 ファイル構成

Symfoware Serverは、データベースを制御および格納するため、以下のファイルから構成されています。1つのインスタンスにおける、 これらのファイルの数や配置位置などの関係を、以下に示します。

ファイル種別	必須	個数	配置位置の指定方法
プログラムファイル	0	複数	 デフォルトでは以下に配置されます。 64ビット製品 %Program Files%¥Fujitsu¥symfoserver64 32ビット製品(OSが64ビットの場合) %Program Files(x86)%¥Fujitsu¥symfoserver32 32ビット製品(OSが32ビットの場合) %Program Files%¥Fujitsu¥symfoserver32
データベースクラス タ	0	1	WebAdminまたはサーバコマンドで指定します。
テーブル空間	0	複数	pgAdminまたはDDL文で指定します。
バックアップ	0	複数	WebAdminまたはサーバコマンドで指定します。
コアファイル	0	複数	サーバコマンドまたはpostgresql.confで指定します。
サーバ証明書ファ イル(注)	_	1	postgresql.confで指定します。
サーバ秘密鍵ファ イル(注)	_	1	postgresql.confで指定します。
マスタ暗号化キー ファイル(注)	_	1	postgresql.confでキーストアとして作成先のディレクトリを指定します。
接続サービスファイ ル(注)	_	1	環境変数で指定します。
パスワードファイル (注)	_	1	環境変数で指定します。
CA証明書ファイル (注)	_	1	環境変数で指定します。

+			
表1.4 1つのインスタン	スにおけるファイル	図と配直位直0)指定万法

○:必須、-:選択

注) それぞれの該当機能を使用する場合に手動で設定します。



ウィルス対策ソフトを使用している場合、Symfoware Serverを構成するすべてのファイルがウィルススキャンの対象外となるようにディレクトリに対するスキャンの除外設定を行ってください。また、Symfoware Serverを構成するファイルに対してウィルススキャンを行う場合は、Symfoware Serverを停止し、Symfoware Serverを利用した業務が動作していない状態で実行してください。

1.6 運用に利用するアプリケーションの互換に関する注意事項

Symfoware Serverをバージョンアップする際、機能改善や機能拡張に伴ってアプリケーションに影響するような変更が発生する場合があります。

したがって、アプリケーションを開発する際には、今後新しいバージョンのSymfoware Serverにアップグレードする場合にも互換性を維持できるように注意して作成してください。



第2章 インスタンスの起動とデータベースの作成

本章では、インスタンスの起動からデータベースの作成まで、基本的な操作に関して説明します。

2.1 インスタンスの起動と停止

インスタンスの起動と停止について説明します。

- 2.1.1 WebAdminを使用する場合
- 2.1.2 コマンドを利用する場合

関 ポイント

データベースサーバのOSの起動・停止に連動してインスタンスの自動起動・停止を行う設定を変更する場合は、"導入ガイド(サーバ 編)"の"インスタンスの自動起動・停止の設定"を参照してください。



インスタンスの停止にimmediateモードで停止した場合やインスタンスが異常終了した場合は、収集している統計情報が初期化されま す。統計情報の初期化に備えて、SELECT文を使用して、定期的に統計情報を採取するなどの実施を検討してください。統計情報の 詳細については"PostgreSQL文書"の"サーバの管理"の"統計情報コレクタ"を参照してください。

2.1.1 WebAdminを使用する場合

WebAdminを使用して、インスタンスの起動、停止、および稼働状態の確認を行うことができます。

インスタンスの起動

インスタンスの起動は、WebAdminの[監視]メニュー画面で行います。

インスタンスが停止中の場合は画面に[起動]ボタンが表示されます。

停止中のインスタンスを起動する場合は、[起動]ボタンをクリックしてください。

インスタンスの停止

インスタンスの停止は、WebAdminの[監視]メニュー画面で行います。

インスタンスが起動中の場合は画面に[停止]ボタンが表示されます。 起動中のインスタンスを停止する場合は、[停止]ボタンをクリックしてください。

停止モード

インスタンスを停止するモードを選択します。各モードの動作を以下に示します。

停止モード	接続されているクライアント	実行中のコマンドによるバックアップ
Smartモード(注)	接続されているすべてのクライアントの切断を 待ちます。	実行中のコマンドによるバックアップの終了を 待ちます。

停止モード	接続されているクライアント	実行中のコマンドによるバックアップ		
Fastモード	すべての実行中のトランザクションをロール バックし、クライアントとの接続を強制的に切 断します。	実行中のコマンドによるバックアップを終了さ せます。		
Immediateモード	ー すべてのサーバプロセスを即座に中断します。次回起動時にクラッシュリカバリ処理が実行な れます。			

注) Smartモードで停止処理中になった後、即座に停止したい場合は、以下の手順で停止させてください。

- 1. WebAdminのWebサーバ機能を再起動します。
- 2. WebAdminに再ログインします。
- 3. [監視]メニュー画面の[停止]ボタンをクリックし、Immediateモードを選択してインスタンスを停止します。

インスタンスの稼働状態の確認

インスタンスの稼働状態は、[監視]メニュー画面で確認できます。

インスタンスの起動中は稼動状態に「起動」と表示されます。インスタンスの停止中は、稼動状態に「停止」と表示されます。異常を検出 した場合は、メッセージリストにエラーメッセージが表示されます。

インスタンスが停止した際には、停止となった原因を取り除き、WebAdminから、インスタンスを起動してください。

図2.1 インスタンス起動中の状態

Symfoware			ユーザー: symfo <u>ログアウト</u> R	มโทรม
監視 設定			😌 🎹 📓 💶	ZIL.
インスタンス一覧	インスタンス名: inst1			
新規作成		•		_
inst1		1592		
	インスタンス	データ格納先	バックアップデータ格納先	
	稼働状態: 起動	状態: 正常	状態: 正常	
	インスタンス名: inst1 ボート番号: 26500 エンコード: UTF8	バス: D:VeymfoVinst1Vdata	パス: EVsymfoVinst1Vbackup	
	停止	デーダ期日 くバックア	ップ日時> 2013-10-17 11 36	
	▼ メッセージ			4
				w.

図2.2 インスタンス停止中の状態

Symfoware 监视 設定			ユーザー: symfo ロ <u>クアウト</u> FUji ② 更能 マニュア
インスタンス一覧 新規作成 inst1 副陰	インスタンス名: inst1		tor vot
	インスタンス 稼働状態: 停止 インスタンス名: inst1 ボート番号: 26500 エンコード: UTF8	データ格納先 状態: 正常 パス: D.¥symfo¥inst1¥data	<mark>バックアップデータ格納先</mark> 状態: 正常 パス: E¥symfo¥inst1¥backup
	起動 ▼ メッセージ () 2013-10-17 1459-18 インスタンス () 2013-10-17 1459-24 インスタンス	データ復旧 0パック7 を停止します。 を停止しました。	7ップ日時> 2013-10-17 1458



・サーバとの通信状態が異常となった場合、WebAdminが無応答になることがあります。その場合、ブラウザを閉じて、再度ログインしてください。解決できない場合、サーバのイベントログを確認し、通信に異常が発生していないか確認してください。

 インスタンスの起動中に出力される以下のメッセージは、起動処理の正常な動作によって出力されるメッセージのため、ユーザー が意識する必要はありません。

 $\ensuremath{\mathsf{FATAL}}\xspace$ the database system is starting up

2.1.2 コマンドを利用する場合

Windowsサービスに関するコマンドを利用して、インスタンスの起動、停止、および稼働状態の確認を行うことができます。

Windowsサービスを利用するにあたっては、Windowsサービスへのインスタンスの登録を行ってください。



Windowsサービスへのインスタンスの登録については、"導入ガイド(サーバ編)"の"インスタンスの自動起動・停止の設定"において、 "Windowsサービスへのインスタンスの登録"を参照してください。



Windowsサービスにインスタンスを登録せず、pg_ctlコマンドの実行によりインスタンスを起動・停止することも可能ですが、以下の理由 によりWindowsサービスを利用したインスタンスの起動・停止を推奨します。 ・ pg_ctlコマンドでインスタンスを起動するとユーザープロセスとして起動されます。そのためコマンドを実行したコマンドプロンプトウィンドウを閉じると、postgresのプロセスが強制停止となります。

インスタンスの起動

net startコマンド、またはsc startコマンドにサービス名を指定して起動します。

またWindowsサービスの画面から以下の手順で起動することもできます。

- [サービス]ウィンドウを表示する
 [管理ツール]で、[サービス]をクリックします。
- 2. サービスを起動する

サービス名一覧の中から、起動したいインスタンス名を選択し、[サービスの開始]ボタンをクリックします。

インスタンスの停止

net stopコマンド、またはsc stopコマンドにサービス名を指定して停止します。 またWindowsサービスの画面から以下の手順で停止することもできます。

- [サービス]ウィンドウを表示する
 [管理ツール]で、[サービス]をクリックします。
- 2. サービスを停止する

サービス名一覧の中から、停止したいインスタンス名を選択し、[サービスの停止]ボタンをクリックします。アプリケーションおよび コマンドの実行中にサービスの停止を行うとSymfoware Serverはそれらを強制終了して正常停止します。

インスタンスの稼働状態の確認

インスタンスの起動の操作直後に稼働されているかどうかを確認する場合は、サービス上で以下の手順でサービスの状態を確認して ください。

- [サービス]ウィンドウを表示する
 [管理ツール]で、[サービス]をクリックします。
- 2. サービスの状態を確認する

サービス名一覧の中から、対象となるSymfoware Serverのサービス状態を確認します。

運用中にインスタンスの稼働状態を確認する場合は、pg_ctlコマンドを使用してください。

pg_ctlコマンドには、以下を指定します。

- モードは、statusを指定します。
- ・ -Dオプションは、データ格納先のディレクトリを指定します。-Dオプションを省略した場合、PGDATA環境変数の値が使用されます。



> pg_ctl status -D D:¥database¥inst1 pg_ctl: サーバが動作中です(PID: 1234)

【未起動状態の場合】

> pg_ctl status -D D:¥database¥inst1 pg_ctl: サーバが動作していません



net startコマンドや、sc queryコマンドにより、インスタンスの稼働状態を確認することもできます。

🐴 参照

pg_ctlコマンドの詳細は、"PostgreSQL文書"の"リファレンス"の"pg_ctl"を参照してください。

2.2 データベースの作成

データベースの作成について説明します。

- 2.2.1 pgAdminを使用する場合
- ・ 2.2.2 クライアントコマンドを使用する場合

2.2.1 pgAdminを使用する場合

pgAdminを使用したデータベースの定義について説明します。

1. pgAdmin画面の[オブジェクトブラウザ]のデータベースを選択し、右クリックで表示される[新しいデータベース]をクリックして[新しいデータベース]画面を表示します。

🁎 pgAdmin III		
ファイル(F) 編集(E) プラグイン(P) ビュ	ー(V) ツール(T) ヘルプ(H)	
🎽 🥙 📾 🧏 🗑 🔊	📰 🛃 🌽 🚉 - 🛡 💡	
オブジェクトブラウザ	× プロパティ 統計情報 属性情報 依存関係	
□ □ □ U U U U U U U U U U U U U U U U U	データベース オーナー コメン	<u>۲</u>
🖶 🗋 db01 (sv1:26500)	gepostgres symfo	
■ ● ● データベース (1) ● ● ● テーブル空間 (2 リフレッシ	⊐(f)	
一 2回 グループロール(5 et 7	
■ ※ ログインロール (1 新しいテー	9/-/	
レポート(R) +	
		<u> </u>
	SQLË1-	×
		E. F.
データベースで詳細の回復完了		0.00秒 🔐

- 2. [新しいデータベース]画面で以下の項目を指定してください。
 - ー [プロパティ]タブ

以下は、データベース db01 を作成するときの例です。

○新しい	データベース	×
プロパティ	定義 変数 権限 Security Labels SQL	_
名前	db01	
OID		
オーナー		-
אעצב		6.
		F
ヘルプ	<u></u>	<u>c)</u>
		зđ

- [名前]:管理するデータベースの名前

3. [OK]ボタンをクリックしてデータベースを作成します。

2.2.2 クライアントコマンドを使用する場合

クライアントコマンドを使用したデータベースの定義について説明します。

以下にサーバ上からの操作例を示します。

- 1. psqlコマンドを使用して、データベースpostgresに接続します。 "psql postgres"と実行します。
 - > psql postgres psql (9.2.8) Type "help" for help.

2. データベースを作成します。 データベースを作成するために、"CREATE DATABASE データベース名;"文を実行します。

```
postgres=# CREATE DATABASE db01;
CREATE DATABASE
```

3. データベースが作成されたことを確認します。

"¥l+"コマンドを実行し、2で作成したデータベース名が表示されることを確認します。

postgres=# \|+

postgres=#	\ +							
					データベース一覧			
名前	所有者	エンコーディング	照合順序	Ctype(変換演算子)	アクセス権	サイズ	テーブルスベース	認明
db01	cduser	UTF8	C	С		7653 kB	pg_default	
postgres	cduser	UTF8	I C	C		7653 kB	pg_default	default administrative connection database
template0	cduser	UTF8	I C	C	=c/cduser +	7545 kB	pg_default	unmodifiable empty database
	1				cduser=CTc/cduser			
template1	cduser	UTF8	l C	I C I	=c/cduser +	7653 kB	pg_default	default template for new databases
					cduser=CTc/cduser			
(4 行)								

4. データベースpostgresと切断します。

psqlコマンドを終了するために、"¥q"を実行します。

postgres=# ¥q

なお、createdbコマンドを使用してデータベースを作成することもできます。



第3章 データベースのバックアップ

本章では、データベースのバックアップに関して説明します。

バックアップの方法

以下のバックアップ方法でバックアップすることにより、ディスク装置の物理的破損やデータの論理的破損が発生する直前や、バックアップ時点にリカバリできます。

- WebAdminによるバックアップ

GUIにより直感的な画面操作でバックアップできます。

リカバリを行うときは、WebAdminでリカバリします。

- pgx_dmpallコマンドによるバックアップ

スクリプトからpgx_dmpallコマンドを実行して、自動的にバックアップできます。

自動的にバックアップするためには、OSの自動化ソフトウェアに登録する必要があります。各OSのドキュメントに従って実施してください。

リカバリを行うときは、pgx_rcvallコマンドでリカバリします。

バックアップの目安時間

WebAdminまたはpgx_dmpallコマンドを使用した場合の、バックアップ目安時間の導出式を示します。

バックアップ時間 = データ格納先の使用量 ÷ ディスク書込み性能 × 1.5

- データ格納先の使用量:データ格納先のディスク使用量
- ディスク書込み性能:運用を行うシステム環境における、1秒間あたりに書き込み可能な最大のデータ量(バイト/秒)の実測値
- 1.5:最も時間のかかるディスク書き込み以外の時間を見込んだ係数



・ 選択したバックアップ方法を継続して使用してください。

各バックアップ方法は、保存するデータ形式などに差異があります。そのため、以下の制約があります。

- バックアップデータを使用して、異なる方法でリカバリすることはできません。
- バックアップデータを異なる方法のバックアップデータに移行できません。
- データベースに格納するデータを暗号化する場合、キーストアのバックアップやバックアップにおける留意事項があります。詳細は、以下を参照してください。
 - 5.6.4 キーストアのバックアップとリカバリ
 - 5.7 データベースのバックアップとリストア/リカバリ
- テーブル空間を定義した場合、バックアップを行ってください。バックアップを行わなかった場合、リカバリ実行時にテーブル空間のディレクトリ作成が行われず、リカバリが失敗することがあります。リカバリに失敗した場合は、イベントログを参照してテーブル空間のディレクトリ作成後、リカバリを再実行してください。
- ・WebAdminによるバックアップを行う場合、WebAdminがデータベースに接続するために、以下のパスワードファイルをバックアップ 中に一時的に作成します。

- <ユーザープロファイルのフォルダ>¥<ローカル設定のフォルダ>¥Fujitsu¥symfo_バージョン¥インスタンス名¥pgpass.conf

そのため、postgresql.confのcore_directoryパラメータに作成されるコアファイルや、log_directoryパラメータに作成されるログファイルを退避する場合には、同じディレクトリ配下にあるパスワードファイルを同時に退避しないようにしてください。



以下の方式によるバックアップも可能です。これらの方法でバックアップをすることにより、バックアップを行った時点に復旧できます。

- SQLによるダンプを用いたバックアップ
 SQLを使用してデータをダンプします。データの移行にも使えるバックアップ方法です。
- ファイルシステムレベルのバックアップ
 インスタンスを停止し、OSコマンドなどを使用してデータベース資源をファイルとしてバックアップする方法です。
- 継続的アーカイブによるバックアップ
 PostgreSQLの標準的なバックアップ方法です。

これらのバックアップ方法の詳細は、"PostgreSQL 文書"の"サーバの管理"の"バックアップとリストア"を参照してください。

3.1 定期的なバックアップ

バックアップを定期的に行うことを推奨します。

WebAdminまたはpgx_dmpallコマンドを使用して定期的にバックアップすることにより、以下の効果があります。

 不要になったアーカイブログ(バックアップデータ格納先にコピーされたトランザクションログ)が削除されるため、ディスク使用率を 抑えることができます。また、障害発生時にリカバリ時間を最小限にできます。

バックアップ周期

定期的にバックアップを行う間隔のことを、バックアップ周期と呼びます。たとえば、毎朝バックアップを行う場合、バックアップ周期は1日となります。

バックアップ周期は運用する業務内容によって異なりますが、Symfoware Serverでは、1日単位で1回以上のバックアップ周期で運用することを推奨します。

3.2 バックアップ方法

データベースのバックアップ方法について説明します。

- 3.2.1 WebAdminを使用する場合
- ・ 3.2.2 サーバコマンドを使用する場合

3.2.1 WebAdminを使用する場合

WebAdminを使用して、バックアップの実行およびバックアップ状態の確認を行うことができます。



バックアップの実行

以下の手順でデータベースをバックアップしてください。

1. データベースのバックアップ選択

WebAdminの[監視]メニュー画面の「バックアップ」と記された[→]ボタンをクリックします。

Symfoware			ユーザー: symfo <u>ログアウト</u> FU <mark>jiTSU</mark>
監視設定			😧 更新 🍵 マニュアル
イノスダノス一覧	イノスダノス名: inst1		
新規作成			
inst!			
		J.F.	171717
	インスタンス	データ格納先	バックアップデータ格納先
	稼働状態: 起動	状態: 正常	状態: 正常
	インスタンス名: inst1	パス: D.VeymfoVinst1Vdata	パス: E:VaymfoVinst1Vbackup
	ボート番号: 26500		
	IVI-F: UTF8		
	停止	デーダ調日 くパック	7~7日時> 2013-10-17 11 36
	▼ メッセージ		A

2. データベースのバックアップ実行

[バックアップ]ダイアログが表示されます。バックアップを実行する場合は[実行]ボタンをクリックします。バックアップを実行すると、インスタンスが自動的に起動されます。

バックアップ状態

何らかの問題が発生してバックアップに失敗すると、[監視]メニュー画面の「データ格納先」または「バックアップデータ格納先」の「状態」に「異常」と表示されます。また、メッセージリストにエラーメッセージが表示されます。

この場合、バックアップデータは最適化されません。バックアップを実行した際には、必ずバックアップの実行結果を確認してください。 バックアップに失敗した場合は、エラーメッセージの右側に[対処]ボタンが表示され、クリックするとエラーの原因を解決するための方 法が表示されます。失敗の原因を取り除き、再度バックアップを実行してください。

Symfoware			ユーザー: symfo <u>ログアウト</u>	คปุการม
監視 設定			😌 🍱 🖥 S	7 <u>= a7/k</u>
イノスタンス一覧	インスダンス名: inst1			
新規作成				
inst1			177-17	
				バックアップ状態を確認
	インスタンス	データ格納先	バックアップデータ格納先	
	稼働状態: 起動	状態: 正常	状態: 正常	
	インスタンス名: inst1 ボート番号: 26500 エンコード: UTF8	パス: Di¥symfo¥inst1¥data	バス: EVsymfoVinst1Wbackup	
	停止	デーダ調日 くバック	アップ日時> 2013-10-17 11:36	
	▼ メッセージ			A
				w

3.2.2 サーバコマンドを使用する場合

pgx_dmpallコマンド、およびpgx_rcvallコマンドを使用して、バックアップの実行、およびバックアップ状態の確認を行います。

バックアップの準備

バックアップを実施する前に、バックアップの準備を実施する必要があります。 以下の手順で実施してください。

💦 参照

バックアップに必要となるディレクトリの配置や注意事項については、"導入ガイド(サーバ編)"の"資源配置用のディレクトリの準備"を 参照してください。

1. バックアップデータ格納ディスクの用意

データ格納ディスクとは別のディスク装置をバックアップのために用意し、OSの機能を使用してマウントします。

2. バックアップデータ格納先のディレクトリの作成

空のディレクトリを作成します。

Windows(R)のエクスプローラの[プロパティ]で、作成したディレクトリにインスタンス管理者のみがアクセスできるように許可を設定します。



[プロパティ]の詳細は、Windows(R)の[ヘルプとサポート]を参照してください。

3. バックアップに必要な設定

インスタンスを停止し、postgresql.confファイルに以下のパラメータを設定してください。postgresql.confファイルを編集後は、インスタンスを起動してください。

パラメータ名	設定値	説明	
backup_destination	バックアップデータ格納先のディレクトリ 名	バックアップデータを格納するディレクト リ名を指定します。	
		指定するディレクトリは、インスタンス管 理者のみがアクセスできるように権限を 設定する必要があります。	
		なお、バックアップデータ格納先のディ レクトリは、データ格納先のディレクトリ、 テーブル空間ディレクトリ、およびトラン ザクションログ格納先のディレクトリの外 に配置してください。	
wal_level	archiveまたはhot_standby(注)	トランザクションログの出力レベルを指定 します。	
		注) hot_standbyはストリーミングレプリ ケーション向けの設定です。	
archive_mode	on	アーカイブログモードを指定します。	
		on(行う)を指定してください。	
archive_command	'cmd /c ""インストールディレクトリ¥¥bin¥ ¥pgx_xlogcopy.cmd" "%p" "バックアッ	トランザクションログを保存するコマンド と格納先パス名を指定します。	
	プデータ格納先ディレクトリ¥ ¥archived_xlog¥¥%f""'	なお、パスを指定するときは以下に注意 してください。	
		・ パスの区切り文字は¥¥を指定する。	
		・ パスに空白を含む場合は""で囲む。	

パラメータの詳細については、"付録A パラメータ"、および"PostgreSQL文書"の"サーバの管理"の"ログ先行書き込み(WAL)" を参照してください。

バックアップの実行

pgx_dmpallコマンドを使用して、バックアップを実行します。また、pgx_dmpallコマンドをOSの自動化ソフトウェアに組み込んでバックアップを実行することもできます。

バックアップデータは、postgresql.confのbackup_destinationパラメータに指定したディレクトリに格納されます。

-Dオプションは、データ格納先のディレクトリを指定します。-Dオプションを省略した場合、PGDATA環境変数の値が使用されます。





バックアップを実行した際には、実行時に取得したデータと、前回取得したデータのバックアップデータが保管されます。 データベースに格納するデータを暗号化している場合、以下を参照してキーストアのバックアップを実施してください。

• 5.6.4 キーストアのバックアップとリカバリ

バックアップ状態

pgx_rcvallコマンドを使用して、バックアップの状況を確認します。

pgx_rcvallコマンドには、以下を指定します。

- ・ lオプションは、バックアップデータの情報を表示します。
- ・ -Dオプションは、データ格納先のディレクトリを指定します。-Dオプションを省略した場合、PGDATA環境変数の値が使用されます。

> pgx_rcvall -l -D D:¥database¥inst1					
Date	Status	Dir			
2013-07-01 13:30:40	COMPLETE	E:¥backup¥inst1¥2013-07-01_13-30-40			

何らかの問題が発生してバックアップに失敗していると、イベントログにメッセージが出力されます。

この場合、バックアップデータは最適化されません。バックアップを実行した際には、必ずバックアップの実行結果を確認してください。バックアップに失敗した場合は、失敗の原因を取り除き、再度バックアップを実行してください。



pgx_dmpallコマンド、およびpgx_rcvallコマンドの詳細は、"リファレンス"の"pgx_dmpall"、および"pgx_rcvall"を参照してください。

リストアポイントの設定

データベースを、ある決めた時点にリカバリしたい場合に備え、リカバリによってデータを戻したい時点に名前を付けることができます。 この時点をリストアポイントと呼び、psqlコマンドにより設定することができます。

アプリケーションの実行前などに設定しておくと、データの内容がいつの時点に戻るかがわかりやすくなり便利です。

リストアポイントは、バックアップ実行後の任意の時点に設定することができます。一方、バックアップの実行前にリストアポイントを設定 した場合は、その時点にリカバリすることはできません。リストアポイントはアーカイブログに記録され、アーカイブログはバックアップの 実行により破棄されるためです。



以下は、psqlコマンドでデータベースに接続し、SQL文を実行してリストアポイントを設定する場合の例です。

ただし、アプリケーションの互換性を維持することを考慮し、SQL文中の関数を直接使用しないようにしてください。詳細は、"アプリケーション開発ガイド"の"アプリケーションの互換に関する注意事項"を参照してください。

postgres=# SELECT pg_create_restore_point('batch_20130703_1'); LOG: restore point "batch_20130703_1" created at 0/20000E8 STATEMENT: select pg_create_restore_point('batch_20130703_1');
pg_create_restore_point

0/20000E8

(1 行)

.....

リストアポイントを利用してデータベースをリカバリする場合、"8.3.2 pgx_rcvallコマンドを使用する場合"を参照してください。



リストアポイントは、データベース内で一意となるように命名してください。以下の例に示すようにリストアポイントを設定する年月日や時刻を付加し、他のリストアポイントと混同しないようにしてください。

- YYMMDD_HHMMSS
 - YYMMDD:年月日を表します。
 - HHMMSS:時刻を表します。
- ・ 設定したリストアポイントを確認する方法はありません。任意のファイルなどに記録しておいてください。



pg_create_restore_pointの詳細は、"PostgreSQL 文書"の"関数と演算子"の"システム管理関数"を参照してください。
第4章 Secure Sockets Layerによる安全な通信の構成

クライアントとサーバ間の通信データに機密情報を含む場合、通信データを暗号化することで、ネットワーク上の盗聴による脅威から通 信データを保護することができます。

4.1 通信データの暗号化のセットアップ

クライアントとサーバ間の通信データを暗号化する場合、以下のセットアップを行ってください。

通信データの暗号化を行うと、通信内容を保護することに加え、中間者攻撃(例えばサーバのなりすましによりデータやパスワードを横 奪するなど)を防止することができます。

表4.1 セットアップ手順

セットアップ手順
1) 証明書発行の手続き
2) サーバ証明書ファイル、サーバ秘密鍵ファイルの配置
3) CA証明書ファイルのクライアントへの配布
4) データベースサーバの動作環境の設定
5) クライアントの動作環境の設定

通信データの暗号化の環境を以下に示します。



図4.1 通信データの暗号化の環境

4.1.1 証明書発行の手続き

サーバ認証を行う場合は、認証局(CA)に証明書を発行してもらう手続きが必要です。 Symfoware Serverでは、X.509の規格のPEM形式のファイルをサポートしています。 認証局からDER形式で発行された場合は、opensslコマンドなどのツールを使用してDER形式をPEM形式に変換してください。

以下に手順の概要を示します。詳細は、証明書ファイルの取得先である、公的または独自の認証局(CA)より公開されている手順を参照してください。

- a. サーバ秘密鍵ファイルの作成
- b. サーバ秘密鍵ファイルのパスフレーズ解除
- c. サーバ秘密鍵ファイルからCSR(サーバ証明書を取得するための署名要求)を作成
- d. 認証局(CA)ヘサーバ証明書を申請
- e. 認証局(CA)から、サーバ証明書ファイルおよびCA証明書ファイルを取得
- f. サーバ証明書ファイルおよびCA証明書ファイルを保管 注)紛失や破損した場合は、再発行が必要になります。

上記の手順により、以下のファイルが準備できます。

- サーバ秘密鍵ファイル
- サーバ証明書ファイル
- CA証明書ファイル

4.1.2 サーバ証明書ファイル、サーバ秘密鍵ファイルの配置

データベースサーバのローカルディスクにディレクトリを作成し、サーバ証明書ファイル、サーバ秘密鍵ファイルを配置します。 サーバ証明書ファイル、サーバ秘密鍵ファイルのアクセス権は、OSの機能を利用して、データベース管理者にのみ読込み権を設定し てください。

また、サーバ証明書ファイルおよびサーバ秘密鍵ファイルは破損に備え、バックアップをして、厳重に管理してください。

4.1.3 CA証明書ファイルのクライアントへの配布

クライアントのローカルディスクにディレクトリを作成し、配布されたCA証明書ファイルを配置します。 CA証明書ファイルを誤って削除などしないように、OSの機能を利用して、読込み権を設定してください。

4.1.4 データベースサーバの動作環境の設定

💦 参照

詳細については、"PostgreSQL 文書"の"サーバの管理"の"SSLによる安全なTCP/IP接続"を参照してください。

4.1.5 クライアントの動作環境の設定

🐴 参照

詳細については、アプリケーションの開発環境に応じて、"アプリケーション開発ガイド"の以下の項を参照してください。

- ・ "JDBCドライバ"の"セットアップ"の"通信データを暗号化する場合の設定"
- ・ "C言語用ライブラリ(libpq)"の"セットアップ"の"通信データを暗号化する場合の設定"
- ・ "C言語による埋め込みSQL"の"セットアップ"の"通信データを暗号化する場合の設定"

4.1.6 データベース多重化運用を行う場合

データベース多重化機能を使用して、かつSSLのサーバ証明書を利用した通信を行う場合には、"Common Name"パラメータの値が 同じサーバ証明書を使用する必要があります。データベース多重化運用を構成する各サーバにおいて、"Common Name"パラメータ の値が同じサーバ証明書を使用するためには、以下のいずれかを行ってください。

- ・ サーバ証明書を1つ作成し、複製して各サーバに配置する。
- ・ サーバ証明書を各サーバごとに作成し、"Common Name"を同じ名前にする。

💦 参照

クライアント側でのアプリケーションの指定方法については、"アプリケーション開発ガイド"の"アプリケーションの接続先切り替え機能 を利用する"を参照してください。

第5章 透過的データ暗号化による格納データの保護

データベースに格納するデータの暗号化について説明します。

5.1 暗号化によるデータの保護

PostgreSQLでは、認証とアクセス制御によって、データベース内のデータは認可されていないデータベースユーザーによるアクセスから保護されます。しかし、データベースサーバの認証とアクセス制御を迂回する攻撃者に対しては、OSファイルは保護されません。

Symfoware ServerではOSのファイル内のデータが暗号化されているため、たとえそのファイルやディスクが盗まれても、貴重な情報は保護されます。

データベースに格納するデータは、データファイルに書き出されるときに暗号化され、読み出されるときに復号されます。これはインス タンスによって自動的に行われるため、ユーザーやアプリケーションが意識することなく、キーの管理や暗号化/復号の処理を実行でき ます。これを透過的データ暗号化 (TDE: Transparent Data Encryption) と呼びます。

TDEには次の特長があります。

暗号化の仕組み

2層の暗号化キーとキーストア

各テーブル空間には、その中のすべてのデータを暗号化/復号するテーブル空間暗号化キーがあります。テーブル空間暗号化 キーは、マスタ暗号化キーで暗号化されて保存されます。

マスタ暗号化キーは、データベースクラスタに1つだけ存在します。利用者が指定するパスフレーズに基づいて暗号化され、キーストアに保存されます。Symfoware Serverは、ファイルベースのキーストアを提供します。パスフレーズを知らない攻撃者は、キーストアからマスタ暗号化キーを読み出すことはできません。

強力な暗号化アルゴリズムを利用

暗号化アルゴリズムとしてAES (Advanced Encryption Standard)を使用します。AESは2002年に米国連邦政府の標準として採用され、世界中で広く使われています。

ハードウェアに基づく暗号化/復号の高速化

Intel Xeonプロセッサの5600番台以降に搭載されたAES-NI (Advanced Encryption Standard New Instructions)を活用し、暗号化と 復号のオーバヘッドを極小化します。これにより、従来は性能とセキュリティとのトレードオフのために最少限の暗号化対象を選別 していた場面でも、アプリケーション全体のデータを暗号化できるようになります。

Intel社Webサイトの次のページで、AES-NIを備えたプロセッサの一覧を参照できます。

http://ark.intel.com/search/advanced/?s=t&AESTech=true

記憶領域のゼロ・オーバヘッド

テーブルやインデックス、WALに格納されるデータの大きさは、暗号化しても変わりません。そのため、追加の見積りやディスクは不要です。

暗号化の範囲

指定したテーブル空間内のすべてのユーザーデータ

暗号化を指定する単位はテーブル空間です。暗号化テーブル空間内に作成されるテーブルとインデックス、一時テーブルと一時 インデックスの全体が暗号化されます。利用者はどのテーブルや列を暗号化するかを考える必要はありません。

バックアップデータ

pgx_dmpallコマンド、およびpg_basebackupコマンドは、OSファイルをコピーすることによりバックアップデータを作成します。そのため、暗号化されたデータのバックアップは暗号化されたままです。バックアップ・メディアが盗まれても、情報は漏えいから保護されます。

WALと一時ファイル

暗号化されたテーブルとインデックスの更新で生成されるWALは、更新対象と同じセキュリティ強度で暗号化されます。大きな結 合やソートを実行するときには、暗号化データは一時ファイルにも暗号化された形で書き出されます。

ストリーミングレプリケーションのサポート

ストリーミングレプリケーションと透過的データ暗号化を組み合わせて使うことができます。プライマリサーバで暗号化されたデータとWALは、暗号化されたままスタンバイサーバに転送され、格納されます。



以下については暗号化されません。

- ・ pg_dumpおよびpg_dumpallコマンドの出力ファイル
- COPYコマンドの出力ファイル
- ・ LISTEN/NOTIFYコマンドでやりとりする通知イベントのペイロード

5.2 マスタ暗号化キーの設定

透過的データ暗号化を使用するには、キーストアを作成し、マスタ暗号化キーを設定する必要があります。

1. postgresql.confのkeystore_locationパラメータに、キーストアを格納するディレクトリを設定します。

データベースクラスタごとに異なる場所を指定してください。

keystore_location = 'C:¥¥key¥¥store¥¥location'

postgresql.confについては、"付録A パラメータ"を参照してください。

postgresql.confファイルを編集後は、インスタンスを起動、または再起動してください。

- WebAdminを使用する場合

"2.1.1 WebAdminを使用する場合"を参照してインスタンスを再起動します。

- コマンドを利用する場合

"2.1.2 コマンドを利用する場合"を参照してインスタンスを再起動します。

2. 以下のようなSQL関数を実行してマスタ暗号化キーを設定します。

データベースのスーパーユーザーで実行してください。

SELECT pgx_set_master_key('passphrase');

passphraseは、今後キーストアをオープンするために使われるパスフレーズです。マスタ暗号化キーはこのパスフレーズによって 保護されるため、短く単純で推測されやすい文字列を指定しないよう注意してください。 pgx_set_master_key関数の詳細は、"B.2 透過的データ暗号化制御関数"を参照してください。



ここで指定したパスフレーズを忘れてしまうと、暗号化されたデータにはアクセスできなくなります。忘れたパスフレーズを取り戻したり、 データを復号する方法はありません。決してパスフレーズを忘れないようにしてください。

pgx_set_master_key関数は、キーストア格納ディレクトリにkeystore.ksというファイルを作成します。そして、ランダムなビット列からなるマ スタ暗号化キーを生成し、それを指定されたパスフレーズで暗号化してkeystore.ksに格納します。キーストアは、オープンされた状態 になります。

5.3 キーストアのオープン

暗号化テーブル空間を作成したり暗号化データにアクセスするには、キーストアをオープンしておく必要があります。キーストアをオー プンすると、マスタ暗号化キーがデータベースサーバのメモリにロードされ、暗号化と復号に利用できるようになります。

インスタンスを起動するたびにキーストアをオープンしてください。キーストアをオープンするには、データベースのスーパーユーザーが次のようにSQL 関数を実行します。

SELECT pgx_open_keystore('passphrase');

passphraseはキーストアの作成時に指定したパスフレーズです。

pgx_open_keystore関数の詳細は、"B.2 透過的データ暗号化制御関数"を参照してください。

ただし、以下の場合には、リカバリのために暗号化されたWALを復号する必要があるため、インスタンスを起動する時にパスフレーズの入力が必要になります。この場合は、上述のpgx_open_keystore関数を実行することができません。

- ・ インスタンス起動時にクラッシュリカバリが行われる場合
- ・ 継続的アーカイブによるリカバリを行う場合

上記の場合には、以下のいずれかの方法を選択してください。

• 自動オープン・キーストアを使用する

セキュリティ強度より運用操作の簡便さを優先する場合に選択してください。自動オープン・キーストアを使用すると、キーストアファイルの内容を復号してキーストアファイルのコピーを生成します。このファイルの内容は難読化されていますが、セキュリティ強度が若干弱くなります。

WebAdminを使用して運用操作を行っている場合は、こちらを選択してください。

・ インスタンスの起動時にパスフレーズを入力する

セキュリティ強度を運用操作のしやすさより優先する場合に選択してください。

pg_ctlコマンドに--keystore-passphraseオプションを指定して起動します。パスフレーズの入力を促すプロンプトが表示されます。

```
> pg_ctl --keystore-passphrase start
パスフレーズを入力してください:
サーバは起動中です
>
```

上記操作の後、一度、pg_ctlコマンドでインスタンスを停止してください。

そして、Windowsサービスからインスタンスを起動してください。Windowsサービスからインスタンスを起動する方法については、 "2.1.2 コマンドを利用する場合"を参照してください。



自動オープン・キーストアを使用すると、パスフレーズを入力することなく、インスタンスの起動時に自動的にキーストアをオープンできます。詳細は"5.6.3 キーストアの自動オープンの有効化"を参照してください。

5.4 テーブル空間の暗号化

暗号化テーブル空間を作成するには、事前にキーストアがオープンされている必要があります。

暗号化するテーブル空間を作成するときに、実行時パラメータに暗号化アルゴリズムを設定します。たとえば、暗号化アルゴリズムとしてキー長が256ビットのAESを用い、secure_tablespaceという名前のテーブル空間を作成するには、次のようにします。

-- 以降に作成するテーブル空間の暗号化アルゴリズムを指定
 SET tablespace_encryption_algorithm = 'AES256';
 CREATE TABLESPACE secure_tablespace LOCATION 'C:¥My¥Data¥Dir';
 -- 以降に作成するテーブル空間は暗号化しないように指定
 SET tablespace_encryption_algorithm = 'none';

暗号化アルゴリズムとして、キー長が128または256ビットのAESを使用できます。256ビットのAESを推奨します。実行時パラメータの指定方法については"付録A パラメータ"を参照してください。

テーブル空間pg_defaultおよびpg_globalを暗号化することはできません。

そして、作成した暗号化テーブル空間にテーブルやインデックスを作成します。暗号化テーブル空間に作成されたリレーションは自動的に暗号化されます。



例1: 作成時に暗号化テーブル空間を指定

CREATE TABLE my_table (...) TABLESPACE secure_tablespace;

例2: 作成時にはテーブル空間を明示せず、デフォルト・テーブル空間を使用

SET default_tablespace = 'secure_tablespace'; CREATE TABLE my_table (...);

ー時テーブルと一時インデックスを暗号化する場合も同様です。つまり、TABLESPACE句を明示的に指定するか、または temp_tablespacesパラメータに暗号化テーブル空間を列挙し、CREATE TEMPORARY TABLEやCREATE INDEXを実行します。

データベースを作成するときに、CREATE DATABASE文のTABLESPACE句に暗号化テーブル空間を指定すると、明示的にテーブル空間を指定しないでそのデータベースに作成されたリレーションは暗号化されます。さらに、システムカタログも暗号化されるので、 ユーザー定義関数のソースコードも保護されます。



pgAdminのテーブル空間作成画面または、クエリーツールから、暗号化テーブル空間の作成はできません。暗号化テーブル空間を 作成する場合は、[プラグイン(P)]メニューの[PSQL Console]をクリックし、psqlのコンソール画面で暗号化テーブル空間の作成を行っ てください。

5.5 暗号化されているテーブル空間の確認

システムビューpgx_tablespacesは、各テーブル空間が暗号化されているかどうか、および暗号化アルゴリズムについての情報を示します。列の詳細については"C.1 pgx_tablespaces"を参照してください。

次のようなSQL文を実行することで、どのテーブル空間が暗号化されているかを知ることができます。

ただし、アプリケーションの互換性を維持することを考慮し、下記のSQL文中のシステムカタログ(pg_tablespace)を直接参照しないよう にしてください。

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



postgres=# SELECT s spcname	spcname, spcencalgo FROM pg_tablespace ts, pgx_tablespaces tsx WHERE ts.oid = tsx.spctablespace; spcencalgo -+
pg_default	none
pg_global	none
secure_tablespace	AES256
(3 rows)	



アプリケーションの互換性の維持に関する詳細は、"アプリケーション開発ガイド"の"アプリケーションの互換に関する注意事項"を参照してください。

5.6 キーストアの管理

盗難による脅威に備えるための、キーストアやマスタ暗号化キーの管理方法について説明します。

5.6.1 マスタ暗号化キーの変更

同じ暗号化キーを長期間使い続けることは、攻撃者に対して暗号データの解読の機会を与えてしまいます。一定期間ごと、またはキーが危険にさらされたときには、キーを変更することを薦めます。

どのくらいの期間でキーを変更すればよいかについては、暗号化アルゴリズムとキー管理についての業界のベストプラクティスに従っ てください。たとえば、米国のNISTが発行している'NIST Special Publication 800-57'があります。PCI DSSもこれに言及しています。 その中では、マスタ暗号化キーは1年ごとに一度変更することが推奨されています。 マスタ暗号化キーを変更するには、最初に設定したときと同じくpgx_set_master_key関数を実行します。詳細は"5.2 マスタ暗号化キーの設定"を参照してください。

マスタ暗号化キーを変更したら、直ちにキーストアをバックアップしてください。

5.6.2 キーストアのパスフレーズの変更

組織のセキュリティ・ポリシーでは通常、パスフレーズを知るセキュリティ管理者が異動や退職で職務から外れるときには、パスフレーズ を変更することが求められます。また、万一ソーシャル・エンジニアリングのような手口でパスフレーズが危険にさらされたときには、パ スフレーズを変更することをお薦めします。

キーストアのパスフレーズを変更するには、データベースのスーパーユーザーで次のようにSQL関数を実行します。

SELECT pgx_set_keystore_passphrase('old_passphrase', 'new_passphrase');

パスフレーズを変更したら、直ちにキーストアをバックアップしてください。

pgx_set_keystore_passphrase関数の詳細は、"B.2 透過的データ暗号化制御関数"を参照してください。

5.6.3 キーストアの自動オープンの有効化

自動オープン・キーストアを使用すると、パスフレーズを入力することなく、インスタンスの起動時に自動的にキーストアをオープンできます。キーストアの自動オープンを有効にするには、pgx_keystoreコマンドを実行します。

> pgx_keystore ---enable-auto-open C:¥key¥store¥location¥keystore.ks パスフレーズを入力してください: キーストアの自動オープンが有効になりました、



pgx_keystoreコマンドの詳細については、"リファレンス"の"pgx_keystore"を参照してください。

自動オープンを有効にすると、元のキーストアと同じディレクトリに自動オープン・キーストアが作成されます。自動オープン・キーストアのファイル名はkeystore.aksです。keystore.aksは、keystore.ksの復号された内容を難読化したコピーです。このファイルが存在すると、インスタンスの起動時に、キーストアをオープンするためのパスフレーズを入力する必要はありません。

元のキーストアのファイルkeystore.ksを削除しないでください。マスタ暗号化キーやパスフレーズを変更するために必要です。これらを変更すると、元のキーストアのファイルkeystore.ksからkeystore.aksが再作成されます。

キーストアを格納したディレクトリ、およびkeystore.ksとkeystore.aksは、インスタンスを起動するユーザーのみがアクセスできるように保護してください。

なお、これらのファイルを作成するSQL関数およびコマンドは、インスタンスを起動するユーザーのみがアクセスできるようにファイルの 許可モードを設定します。そのため、ファイルをリストアしたときなどに手動で同様の許可モードを設定してください。

許可モードはWindows(R)のエクスプローラの[プロパティ]で設定します。



自動オープン・キーストアは、それが作成されたコンピュータでのみオープンします。

キーストアの自動オープンを無効化するには、keystore.aksを削除してください。



• WebAdminを使用してリカバリを行う際には、キーストアの自動オープンを有効にしてから行ってください。

- ・ 暗号化を有効にしたあと、または設定を変更したあとは、"5.7 データベースのバックアップとリストア/リカバリ"を参照して、データベースをバックアップしてください。
- ・ キーストアの格納先には、以下のディレクトリとは別のディレクトリを指定してください。
 - データ格納先
 - テーブル空間格納先
 - トランザクションログ格納先
 - バックアップデータ格納先

5.6.4 キーストアのバックアップとリカバリ

キーストアの破損や消失に備え、次のときにはキーストアをバックアップしてください。ただし、データベースとキーストアは別々の記憶 媒体に保管してください。両方を同じ記憶媒体に格納した場合、その記憶媒体が盗まれると、暗号化データを解読されるおそれがあ ります。自動オープン・キーストアのオープンにはパスフレーズが必要ないため、特に安全な場所に保管してください。

- ・ 最初にマスタ暗号化キーを設定したとき
- ・ マスタ暗号化キーを変更したとき
- データベースをバックアップするとき
- キーストアのパスフレーズを変更したとき



キーストアをバックアップするときには、古いキーストアを上書きしないようにしてください。なぜなら、データベースをリカバリするときに は、データベースのバックアップを取得した時点のキーストアをリストアする必要があるためです。データベースのバックアップデータが 不要になったら、それに対応するキーストアを削除してください。

🌽 例

・2013年7月1日にデータベースとキーストアをバックアップします。

> pgx_dmpall -D D:¥database¥inst1

> copy C:¥key¥store¥location¥keystore.ks C:¥keybackup¥keystore_20130701.ks

pgx_dmpallコマンドには、以下を指定します。

- -Dオプションは、データ格納先のディレクトリを指定します。-Dオプションを省略した場合、PGDATA環境変数の値が使用されます。
- ・2013年7月5日にマスタ暗号化キーを変更し、キーストアをバックアップします。

> psql -c "SELECT pgx_set_master_key('passphrase')" postgres > copy C:¥key¥store¥location¥keystore.ks C:¥keybackup¥keystore_20130705.ks

psqlコマンドには、以下を指定します。

- -cオプションは、マスタ暗号化キーを設定するSQL関数を指定します。
- 引数は、接続するデータベース名を指定します。

キーストアが破損または消失した場合には、最新のマスタ暗号化キーが含まれるキーストアをリストアしてください。もし最新のマスタ暗 号化キーを含むキーストアがない場合は、データベースのバックアップを取得した時点のキーストアをリストアし、そのデータベースの バックアップからデータベースをリカバリしてください。これにより、キーストアも最新状態にリカバリされます。

💕 例

・ 最新のマスタ暗号化キーが含まれる、2013年7月5日時点のキーストアをリストアします。

> copy C:\#keybackup\#keystore_20130705.ks C:\#key\#store\|ocation\#keystore.ks

 最新のマスタ暗号化キーが含まれるキーストアのバックアップがない場合、2013年7月1日にデータベースとともにバックアップした キーストアをリストアしてリカバリします。

> copy C:¥keybackup¥keystore_20130701.ks C:¥key¥store¥location¥keystore.ks > pgx_rcvall -B E:¥backup¥inst1 -D D:¥database¥inst1 --keystore-passphrase

pgx_rcvallコマンドには、以下を指定します。

- -Dオプションは、データ格納先のディレクトリを指定します。-Dオプションを省略した場合、PGDATA環境変数の値が使用されます。
- -Bオプションは、バックアップデータ格納先のディレクトリを指定します。
- --keystore-passphraseオプションは、キーストアをオープンするためのパスフレーズの入力を促します。

キーストアをリストアした際には、再度キーストアの自動オープンを有効にしてください。これにより、自動オープン・キーストア(keystore.aks)の内容が、リストアしたキーストアの内容と一致します。

自動オープン・キーストアのファイルkeystore.aksをバックアップしないことをお薦めします。万一、自動オープン・キーストアが格納されたバックアップ媒体とデータベースのバックアップ媒体の両方が盗まれてしまった場合、攻撃者はパスフレーズを知らなくてもデータを読み取れてしまうためです。

自動オープン・キーストアが破損または消失した場合には、再び自動オープンを有効にしてください。これにより、keystore.ksから keystore.aksが再作成されます。



pgx_rcvallコマンド、およびpgx_dmpallコマンドの詳細は、"リファレンス"の"pgx_rcvall"および"pgx_dmpall"を参照してください。

psqlコマンドの詳細は、"PostgreSQL文書"の"リファレンス"の"psql"を参照してください。

pgx_set_master_key関数の詳細は、"B.2 透過的データ暗号化制御関数"を参照してください。

キーストアの自動オープンの有効化については、"5.6.3キーストアの自動オープンの有効化"を参照してください。

5.7 データベースのバックアップとリストア/リカバリ

Symfoware Serverでは、以下に示す5つのバックアップ/リカバリの方法が利用できます。いずれの方法を用いる場合も、キーストアを同時にバックアップしてください。

ただし、データベースとキーストアは別々の記憶媒体に保管してください。両方を同じ記憶媒体に格納した場合、その記憶媒体が盗まれると、暗号化データを解読されるおそれがあります。

WebAdminによるバックアップ・リカバリ

・ バックアップ

WebAdminは、暗号化されたデータをバックアップします。

データベースのバックアップ後、キーストアをバックアップしてください。

• リカバリ

データベースのバックアップを取得した時点のキーストアをリストアしてください。詳細は、"5.6.4 キーストアのバックアップとリカバリ"を参照してください。

"5.6.3 キーストアの自動オープンの有効化"の手順に従い、キーストアの自動オープンを有効にしてください。その後、WebAdmin でリカバリを実行してください。

pgx_dmpall、pgx_rcvallコマンドによるバックアップ・リカバリ

・ バックアップ

pgx_dmpallコマンドは、暗号化されたデータをバックアップします。 データベースのバックアップ後、キーストアをバックアップしてください。

• リカバリ

データベースのバックアップを取得した時点のキーストアをリストアしてください。

キーストアの自動オープンは、必要に応じて設定してください。

キーストアの自動オープンを有効にしない場合、--keystore-passphraseオプションを指定して、pgx_rcvallコマンドを実行してください。パスフレーズの入力を促すプロンプトが表示されます。



・ 2013年7月1日にデータベースとキーストアをバックアップします。

> pgx_dmpall -D D:¥database¥inst1

> copy C:¥key¥store¥location¥keystore.ks C:¥keybackup¥keystore_20130701.ks

pgx_dmpallコマンドには、以下を指定します。

- -Dオプションは、データ格納先のディレクトリを指定します。-Dオプションを省略した場合、PGDATA環境変数の値が使用されます。

・2013年7月1日に取得したバックアップから、データベースとキーストアをリカバリします。

> copy C:¥keybackup¥keystore_20130701.ks C:¥key¥store¥location¥keystore.ks

> pgx_keystore --enable-auto-open C:¥key¥store¥location¥keystore.ks (自動オープンを有効にする場合のみ実行)

> pgx_rcvall -B E:¥backup¥inst1 -D D:¥database¥inst1 --keystore-passphrase

pgx_rcvallコマンドには、以下を指定します。

- -Dオプションは、データ格納先のディレクトリを指定します。-Dオプションを省略した場合、PGDATA環境変数の値が使用されます。
- -Bオプションは、バックアップデータ格納先のディレクトリを指定します。

— --keystore-passphraseオプションは、キーストアをオープンするためのパスフレーズの入力を促します。

SQLによるダンプとリストア

・ バックアップ

pg_dumpコマンド、およびpg_dumpallコマンドが出力するデータは暗号化されません。そのため、"5.8 データベースのインポートと エクスポート"を参照して、OpenSSLなどのコマンドを利用してデータを暗号化してください。

データベースのバックアップ後、キーストアをバックアップしてください。

・リストア

OpenSSLなどのコマンドを利用して、バックアップデータを暗号化した場合、そのデータを復号してください。

pg_dumpallコマンドが出力するデータは、デフォルトではテーブル空間の暗号化の指定を含みます。そのため、pg_restoreコマンドは、テーブル空間を暗号化してリストアします。

ファイルシステムレベルのバックアップとリストア

・ バックアップ

インスタンスを停止して、OSのファイルコピーコマンドでデータ格納先のディレクトリやテーブル空間ディレクトリをバックアップします。暗号化テーブル空間のファイルは、暗号化された状態でバックアップされます。

バックアップ実行後、キーストアをバックアップしてください。

・リストア

データベースのバックアップを取得した時点のキーストアをリストアしてください。

インスタンスを停止して、OSのファイルコピーコマンドでデータ格納先のディレクトリやテーブル空間ディレクトリをリストアします。

継続的アーカイブによるバックアップとポイントインタイムリカバリ

・ バックアップ

pg_basebackupコマンドは、暗号化されたデータをバックアップします。 バックアップ実行後、キーストアをバックアップしてください。

• リカバリ

データベースのバックアップを取得した時点のキーストアをリストアしてください。

キーストアの自動オープンは、必要に応じて設定してください。

キーストアの自動オープンを有効にしない場合、"5.3 キーストアのオープン"を参照してpg_ctl --keystore-passphrase startを指定した起動を行ってください。

🐴 参照

以下のコマンドの詳細は、"PostgreSOL文書"の"リファレンス"のそれぞれの項を参照してください。

- psql
- pg_dump
- pg_restore
- pg_basebackup

以下のコマンドの詳細は、"リファレンス"を参照してください。

- pgx_rcvall
- pgx_dmpall
- pg_dumpall

キーストアをリストアした際には、再度キーストアの自動オープンを有効にしてください。これにより、自動オープン・キーストア(keystore.aks)の内容が、リストアしたキーストアの内容と一致します。

キーストアの自動オープンの有効化については、"5.6.3 キーストアの自動オープンの有効化"を参照してください。

5.8 データベースのインポートとエクスポート

COPY TOコマンドが出力するファイルは暗号化されません。そのため、他のシステムにファイルを転送する場合は、OpenSSLのコマンドなどでファイルを暗号化したり、Windows用の暗号化通信を行うファイル転送ソフトウェアを利用して、通信中のデータを暗号化してください。

平文のファイルは、不要になったら安全な方法で削除してください。

ファイルを安全に削除するには、以下の方法が利用できます。

・ fsutilコマンド



テーブルmy_tableの内容をCSV形式のファイルにエクスポートする。 > psql -c "COPY my_table TO 'C:¥WINDOWS¥Temp¥my_table.csv' (FORMAT CSV)" postgres # エクスポートしたファイルを暗号化する。 > C:¥OpenSSL-Win32¥bin¥openssl enc -e -aes256 -in C:¥WINDOWS¥Temp¥my_table.csv -out my_table.csv.enc (ここで暗号化に使うパスフレーズの入力が求められる) # 平文ファイルのサイズを確認し、ゼロパディング後に削除 > dir C:¥WINDOWS¥Temp¥my_table.csv > fsutil file setzerodata offset=0 length=7 C:¥WINDOWS¥Temp¥my_table.csv > del C:¥WINDOWS¥Temp¥my_table.csv # 他のシステムで、暗号化ファイルを復号する。 > C:¥OpenSSL-Win32¥bin¥openssl enc -d -aes256 -in my_table.csv.enc -out my_table.csv (ここで復号に使うパスフレーズの入力が求められる)

.....

COPY FROMコマンドのインポート先となるテーブルやインデックスが暗号化テーブル空間にある場合、インポートされたデータは自動的に暗号化された形で格納されます。

5.9 既存データの暗号化

既存の暗号化されていないテーブル空間を暗号化することはできません。また、暗号化されたテーブル空間を暗号化しないように変更することもできません。

代替手段として、テーブルやインデックスを別のテーブル空間に移動してください。このためには、次のSQLコマンドが利用できます。

ALTER TABLE table_name SET TABLESPACE new_tablespace; ALTER INDEX index_name SET TABLESPACE new_tablespace; ALTER DATABASE database_name SET TABLESPACE new_tablespace;



SQLコマンドについては、"PostgreSQL文書"の"リファレンス"の"SQLコマンド"を参照してください。

5.10 クラスタシステムにおける運用

高可用性システムやストリーミングレプリケーション、データベース多重化機能、Mirroring Controllerオプションのようなクラスタシステム上で、透過的データ暗号化を使用する場合について説明します。

5.10.1 データベース多重化運用を用いないHAクラスタ

データベース多重化運用を使用しないHAクラスタ環境で透過的データ暗号化を使用する場合、以下の点を留意してください。

キーストア・ファイルの配置と自動オープン

キーストア・ファイルの配置方法には以下の2通りあります。

- ・1つのキーストア・ファイルを共有する方法
- キーストア・ファイルのコピーを配置する方法

1つのキーストア・ファイルを共有する方法

プライマリサーバとスタンバイサーバで、同じキーストア・ファイルを使用する方法です。

プライマリサーバが稼働している間はスタンバイサーバは動作しないため、同時にこのファイルをアクセスすることはなく、共有する ことができます。

キーストア・ファイルをさらに安全に管理するためには、安全な場所に隔離されたキー管理サーバまたはキー管理ストレージ上に配置してください。

キーストアの自動オープンの有効化は、プライマリサーバとスタンバイサーバの両方で行ってください。

キーストア・ファイルのコピーを配置する方法

プライマリサーバのキーストア・ファイルのコピーをスタンバイサーバ側に配置する方法です。

プライマリ、スタンバイの両方のサーバからアクセスできる共用のサーバやディスク装置が準備できない場合に、この方法を採用してください。

ただし、プライマリサーバでマスタ暗号化キーやパスフレーズを変更した場合、スタンバイサーバにキーストア・ファイルを再度コピー する必要があります。

キーストア・ファイルをさらに安全に管理するためには、プライマリサーバ、スタンバイサーバそれぞれに対して、安全な場所に隔離 されたキー管理サーバまたはキー管理ストレージを準備し、キーストア・ファイルを配置してください。

キーストアの自動オープンの有効化は、プライマリサーバとスタンバイサーバの両方で行ってください。自動オープン・キーストアファイル(keystore.aks)をスタンバイサーバにコピーしても、キーストアの自動オープンは有効になりません。



フェイルオーバ運用によるクラスタシステムの環境構築については、"クラスタ運用ガイド"を参照してください。

5.10.2 データベース多重化運用

ストリーミングレプリケーションや、ストリーミングレプリケーションを利用したデータベース多重化機能、Mirroring Controllerオプションを 使用する環境で透過的データ暗号化を使用する場合、以下の点を留意してください。

キーストア・ファイルの配置

プライマリサーバのキーストア・ファイルのコピーをスタンバイサーバ側に配置してください。

これは、両方のサーバが同時にキーストア・ファイルをアクセスすることがあり、共有できないためです。

関 ポイント

キーストア・ファイルをさらに安全に管理するためには、安全な場所に隔離されたキー管理サーバまたはキー管理ストレージ上に配置 してください。プライマリ、スタンバイの両方のサーバが使うキーストアは、同一のキー管理サーバまたはキー管理ストレージで管理す ることができます。

ただし、プライマリサーバとスタンバイサーバが使うキーストアは異なるディレクトリを作成して、プライマリサーバのキーストアをスタンバイサーバが使うディレクトリ上にコピーして使用してください。

キーストアの自動オープン

キーストアの自動オープンを必ず有効化してください。

このとき、キーストアの自動オープンの有効化は、データベース多重化運用を構成するすべてのサーバで行ってください。キーストアの自動オープンの設定はサーバ固有の情報を含むので、ファイルをコピーしただけでは、有効化されません。

パスフレーズの変更

パスフレーズの変更は、データベース多重化運用を構成するすべてのサーバに反映されるので、特別な作業は必要ありません。

スタンバイサーバの構築と起動

pg_basebackupコマンドまたはpgx_rcvallコマンドでスタンバイサーバを構築する前に、プライマリサーバからスタンバイサーバにキーストア・ファイルをコピーしておいてください。自動オープン・キーストアを使用する場合は、コピーしたキーストア・ファイルを用いてスタンバイサーバで自動オープンを有効にします。

スタンバイサーバを起動するときにキーストアをオープンしてください。これは、プライマリサーバから受信した暗号化されたWALを復 号し、再生するために必要です。キーストアをオープンするには、pg_ctlコマンドまたはpgx_rcvallコマンドに、--keystore-passphraseを 指定してパスフレーズを入力するか、または自動オープン・キーストアを使用します。 pg_ctlコマンドに--keystore-passphraseを指定する場合については、"5.3 キーストアのオープン"を参照してください。

マスタ暗号化キーとパスフレーズの変更

マスタ暗号化キーとパスフレーズはプライマリサーバで変更します。そのとき、プライマリサーバからスタンバイサーバにキーストアをコ ピーする必要はありません。スタンバイサーバを再起動したり、キーストアを再度オープンする必要もありません。マスタ暗号化キーとパ スフレーズの変更は、スタンバイサーバのキーストアにも反映されます。



pgx_rcvallコマンドの詳細は、"リファレンス"の"pgx_rcvall"を参照してください。

pg_basebackupコマンドの詳細は、"PostgreSQL文書"の"リファレンス"の"pg_basebackup"を参照してください。

ストリーミングレプリケーションを構築する手順については、"PostgreSQL文書"の"サーバの管理"の"高可用性、負荷分散およびレプ リケーション"を参照してください。

.

5.11 セキュリティに関する注意事項

- データベースサーバのメモリ(共有バッファ)内では、復号されたデータがキャッシュされます。その結果、プロセスのメモリダンプであるミニダンプには、暗号化されていないデータが格納されます。したがって、安全な方法でメモリダンプを削除してください。安全にファイルを削除するために、以下のコマンドが利用できます。
 - fsutilコマンド
- OSのスワップ領域には、データベースサーバのメモリから暗号化されていないデータが書き出されることがあります。スワップ領域からの情報漏えいを防ぐには、スワップ領域の使用を無効にするか、またはフルディスク暗号化製品を使ってスワップ領域を暗号化することを検討してください。
- ・サーバログファイルの内容は暗号化されません。そのため、SQL文に定数を指定していると、その値がサーバログファイルに出力 されることがあります。これを防ぐには、log_min_error_statementなどのパラメータの設定を検討してください。
- キーストアをオープンしたりマスタ暗号化キーを変更するSQL関数を実行するときには、パスフレーズを含むそのSQL文がサーバ ログファイルに出力されないように注意する必要があります。そのためには、log_min_error_statementなどのパラメータの設定を検 討してください。もしデータベースサーバとは別のコンピュータからこれらのSQL関数を実行する場合は、クライアントとデータベー スサーバとの間の通信をSSLで暗号化してください。

5.12 構築済みアプリケーションの導入のヒント

透過的データ暗号化では、アプリケーションを変更することなく、アプリケーション全体のデータを容易に暗号化できます。

データベース管理者は、次のように構築済みアプリケーションを導入します。ただし、本手順を実行するとデフォルトテーブル空間に データを格納することになりますので、元々の設計と異なる場合は必要に応じて対処してください。

1. (通常の手順)構築済みアプリケーションのための所有者ユーザーとデータベースを作成します。

CREATE USER crm_admin ...; CREATE DATABASE crm_db ...; 2. (暗号化のための手順)構築済みアプリケーションのデータを格納する暗号化テーブル空間を作成します。

SET tablespace_encryption_algorithm = 'AES256'; CREATE TABLESPACE crm_tablespace LOCATION 'C:¥crm¥data';

3. (暗号化のための手順)構築済みアプリケーションの所有者ユーザーのデフォルト・テーブル空間として、暗号化テーブル空間 を設定します。

ALTER USER crm_admin SET default_tablespace = 'crm_tablespace'; ALTER USER crm_admin SET temp_tablespaces = 'crm_tablespace';

4. (通常の手順) 構築済みアプリケーションのインストールを実行します。アプリケーションのインストーラは、データベースサーバ のホスト名とポート番号、ユーザー名とデータベース名の入力を求めます。インストーラは入力された情報を使ってデータベース サーバに接続し、SQLスクリプトを実行します。インストーラがないアプリケーションでは、データベース管理者がSQLスクリプトを 手動で実行します。

通常、アプリケーションのSQLスクリプトは、ER図から変換されたCREATE TABLEやCREATE INDEX、GRANT/REVOKEなどの論理 定義のSQL文を含んでいます。データベースやユーザー、テーブル空間を作成するSQL文は含みません。そのSQLスクリプトを実行 するユーザーのデフォルトのテーブル空間を設定しておくことで、SQLスクリプトが生成するオブジェクトはそのテーブル空間に配置さ れます。

第6章 定期的な運用操作

本章では、日常のデータベース業務を行うにあたって、定期的に行う必要のある運用操作について説明します。

6.1 ログの設定と監視

Symfoware Serverでは、データベースのエラーや警告などをログファイルに出力できます。

エラー発生有無や、エラー原因を特定するために有益な情報となります。

デフォルトでは、イベントログに出力されます。また、Symfoware Serverを運用する前に、Symfoware Serverのログファイル(log_destination など)のログを取得するように設定することを推奨します。

上記で取得したログファイルを定期的に監視し、エラーが発生していないか監視してください。



- ・ ログの詳細は、"PostgreSQL文書"の"サーバの管理"の"エラー報告とログ取得"を参照してください。
- ・ WebAdminで運用する場合のログの設定は、"導入ガイド(サーバ編)"の"設定パラメータ"を参照してください。

6.2 ディスクの使用量の監視と空き領域の確保

データベースを長期間利用すると、ディスク装置の空き領域を圧迫し、ディスク領域が不足してしまうことがあります。この状態になると、 データベース業務が停止して業務が続行できなくなる可能性があります。

そのため、定期的にディスク容量の使用状況を監視し、同じディスク装置内にある不要なファイルを削除します。

ディスクの使用量は、下記のディレクトリを配置しているディスクに対して監視してください。

- ・ データ格納先のディレクトリ
- ・トランザクションログ格納先のディレクトリ(トランザクションログをデータ格納先のディレクトリとは別に格納している場合)
- ・ バックアップデータ格納先のディレクトリ
- ・ テーブル空間格納先のディレクトリ

6.2.1 ディスクの使用量の監視

ディスクの使用量を確認するには、OSが提供する下記のコマンドを使用して確認してください。

・ fsutil volume diskfree コマンド

また、SQL文でテーブルやインデックスの単位で確認することもできます。

この方法は、"PostgreSQL文書"の"サーバの管理"の"ディスク使用量の決定"を確認してください。

参者

WebAdminによる運用を行っている場合は、ディスクの使用量が全体の80%に達すると警告が表示されます。

6.2.2 ディスクの空き領域の確保

ディスク容量の空き領域は、OSが提供する以下のコマンドを使用して、同じディスク装置内にあるデータベース以外の不要なファイル を削除することにより、確保してください。

・ delコマンド

また、以下を定期的に実行することにより、ディスク容量を確保できます。

・ データ格納ディスクの容量を確保する場合

REINDEX文を実行します。詳細は"6.5 インデックスの再編成"を参照してください。

・ バックアップデータ格納ディスクの容量を確保する場合

WebAdmin、またはpgx_dmpallコマンドによりバックアップを実行します。

6.3 コネクションの自動切断

何らかの原因でアプリケーションがダウンして、アプリケーションが異常終了した場合、データベースサーバ上にアプリケーションとのコネクションが残る可能性があります。この状態が長時間続いた場合、新しいアプリケーションがデータベースサーバに接続してもエラーになったり、テーブルなどが使用中である旨のエラーになることがあります。

そのため、一定時間、アイドル中のコネクションを自動的に切断することを推奨します。

コネクションが自動切断されるまでの時間は、postgresql.confファイルに以下のパラメータを設定してください。

パラメータ名	設定値	説明
tcp_keepalives_idle	keepaliveを送信するまでの時間(秒数) 0の場合はシステムのデフォルト値を使用	指定された秒数間、アイドル中のコネクショ ンに対して、keepaliveを送信します。
	します。	30秒を指定することを推奨します。
tcp_keepalives_interval	keepaliveの送信間隔時間(秒数)	指定された間隔でkeepaliveを送信します。
	0の場合はシステムのデフォルト値を使用 します。	6秒を指定することを推奨します。



デスクトップヒープの設定を変更しない限り、コネクションの最大接続数は、125になります。



パラメータの詳細については、"PostgreSQL文書"の"サーバの管理"の"接続設定"を参照してください。

6.4 アプリケーションの接続状態の監視

Symfoware Serverは、更新、削除されたデータを即座に削除しません。バキュームにより、参照するトランザクションが存在しないと判断した場合、これらの不要データを回収します。

しかし、長時間接続したコネクションや、資源を占有したコネクションが存在している場合、不要データが回収されず、データベースが 膨張して、データベースの性能劣化となる可能性があります。

▶ 参照 バキュームの詳細は、"PostgreSQL文書"の"サーバの管理"の"定常的なバキューム作業"を参照してください。

このような場合、原因となるコネクションを監視することで、データベースの性能劣化を抑止できます。 長時間、待ち状態となっているコネクションを監視するには、以下の2つの方法があります。

- 6.4.1 ビュー(pg_stat_activity)を使用する方法
- 6.4.2 pgAdminを使用する方法

6.4.1 ビュー(pg_stat_activity)を使用する方法

| idle in transaction

| begin;

ビュー(pg_stat_activity)を利用して、長時間、待ち状態のクライアントのコネクションを監視します。



以下の例は、クライアントの待ち状態が、60分以上の場合、該当コネクションを表示します。

ただし、アプリケーションの互換性を維持することを考慮し、下記のSQL文中のシステムカタログを直接参照しないようにしてください。

postgres=# select * from pg_stat_activity where state='idle in transaction' and current_timestamp > cast(query_start + interval '60 minutes' as timestamp); -[RECORD 1]-datid | 16384 | db01 datname pid | 16875 | 10 usesysid usename | symfo application_name | apl01 client_addr | 192.33.44.15 client hostname | 51793 client port 2013-05-31 17:40:24.161826+09 backend_start 2013-05-31 17:40:27.636134+09 xact_start query_start | 2013-05-31 17:40:27.636134+09 | 2013-05-31 17:40:27.636402+09 state_change waiting | f



state

query

アプリケーションの互換性の維持に関する詳細は、"アプリケーション開発ガイド"の"アプリケーションの互換に関する注意事項"
 を参照してください。

・ pg_stat_activityの詳細は、"PostgreSQL文書"の"サーバの管理"の"統計情報コレクタ"を参照してください。

6.4.2 pgAdminを使用する方法

pgAdminの[サーバ状態]を使用して、コネクションを監視する手順を説明します。

1. pgAdminの[ツール]メニューの[サーバ状態]をクリックします。



2. 長時間、待ち状態のクライアントのコネクションを特定します。

「TX 開始」に表示されるトランザクション開始時間から、長時間、待ち状態のコネクションを特定してください。

甲 サー	小〔状態 -sv1 (sv1	:26500)						_			
7741	レ(F) 編集(E)	22-(V) ^	JLプ(H)				_		レニン・チャン・フィー	月本はも月まれ	Calle 2193
1.00				- (6)3	580	•	db01	-	トリンリリションB	的知时间包括	AE 86
PID 7690 772 7690	アプリケーション名 ogAdmin III paql ogAdmin III	データベース postgres 6201 6001	2-9 symfo symfo symfo	051701 Client IP Iocal ppe Client IP	55 20 20 20 20	Hアント開始 13-07-22 09:50 13-07-22 09:50 13-07-22 09:50	勿 1:01+09 1:31+09 1:01+09	(9-1896 TX 1896 2013-07-2	5tate Kie 2 09:51.36+09 Kie in transa Kie	State ch 2013-07 cton 2013-07 2013-07	プロック クエリー SELECT BEG3N; SELECT
											,
070											0×
PID	9-91	(-2	リレーション	ユーザ	XID	ΤХ	方法	許可	1936	クエリー	
772	I			symto	5/6	5/6	ExclusiveL	ock (31,1	2013-07-22 09:51:36+09	BEGIN;	
×											
準備12	1999/10				- b./ 1			F 1.4 3			O X
XID	50-7000				714		9-J =	7 - 7/ "A			

6.5 インデックスの再編成

データベースは通常、テーブルにインデックスを定義しますが、データを頻繁に更新すると、インデックスがディスク装置の空き領域を 効率的に利用できなくなってきます。また、これに伴い、データベースへのアクセス性能が少しずつ低下する可能性があります。

ディスク装置の使用領域を整理し、データベースへのアクセス性能の低下を防ぐために、REINDEXコマンドを定期的に実行してイン デックスを再編成することを推奨します。

ディスクの使用量は、"6.2 ディスクの使用量の監視と空き領域の確保"に説明している方法で、データ格納先のディレクトリに対して確認してください。



REINDEXコマンドの定期的な実行によるインデックスの再編成について、詳細な内容は"PostgreSQL文書"の"サーバの管理"の"定 常的なインデックスの再作成"を参照してください。

関 ポイント

通常は月に1回の頻度で、データベースの保守などを行う時期を見計らって実行してください。インデックスの使用量をSQL文で調べ、 日々増加の傾向が見られる場合には、ディスクの空き容量と比較してインデックス再作成の実行頻度を調整します。

SQL文および実行結果の例を以下に示します。

ただし、アプリケーションの互換性を維持することを考慮し、下記のSQL文中のシステムカタログや関数を直接参照したり使用しないようにしてください。

[SQL文]

[実行結果]

publicpgbench_accounts_pkey0.162208 kB13 MBpg_catalogpg_depend_depender_index0.6224 kB368 kBpg_catalogpg_depend_reference_index0.58216 kB368 kB	schema_name	index_name	index_ratio	index_size	table_size
	public	pgbench_accounts_pkey	0. 16	2208 kB	13 MB
	pg_catalog	pg_depend_depender_index	0. 6	224 kB	368 kB
	pg_catalog	pg_depend_reference_index	0. 58	216 kB	368 kB

💦 参照

アプリケーションの互換性の維持に関する詳細は、"アプリケーション開発ガイド"の"アプリケーションの互換に関する注意事項"を参照してください。

6.6 データベース活動状況の監視

Symfoware Serverでは、データベースの活動状況に関する情報を収集できます。これらを監視することで、データベース状況の変化 を確認できます。

これらの情報は、内部ロックなどの待機情報を含め、性能面でのボトルネック解消に役立つ情報となります。また、当社技術員(サポート)への調査依頼時にも有益な情報となるため、取得することを推奨します。

図6.1 情報収集のイメージ



1. 業務時間帯に一定間隔で統計情報を取得します。

取得した情報をファイルに蓄積します。

可能な限り1つのトランザクション内で、各種の統計情報ビューのデータを取得してください。ある瞬間でのシステムの性能のス ナップショットを取ることができるからです。

取得対象のシステムビューについては、"6.6.1 取得できる情報"を参照してください。

2. 業務終了後の後処理時間帯に、統計情報をリセットします。

リセット方法については、"6.6.3 情報のリセット方法"を参照してください。

3. 蓄積した情報ファイルを保存します。

日々の性能の変化を確認したり、サポートに問い合わせるまで情報が削除されないようにするために、蓄積した情報ファイルは 最低でも2日間は保持してください。

24時間業務の場合には、夜間などの業務負荷の低い時間帯に、統計情報のリセットおよび蓄積ファイルの保存を実施するようにして ください。

🚰 注意

統計情報は日々のデータベース状況を加算していくため、統計情報をリセットしないと、カウンタの値が上限を超えて正確な値を得ら れません。

ここでは、以下について説明します。

- 取得できる情報
- 取得するための設定
- 情報のリセット方法

6.6.1 取得できる情報

取得できる情報は、以下の2つに分類されます。

- ・ PostgreSQL共通の情報
- ・ Symfoware Serverが追加した情報

PostgreSQL共通の情報

💦 参照

PostgreSQL共通の情報については、"PostgreSQL文書"の"サーバの管理"の"データベース活動状況の監視"を参照してください。

Symfoware Serverが追加した情報

Symfoware Serverが追加した情報として、以下を取得することができます。

表6.1 Symfoware Serverが追加した情報

ビュー名	説明
pgx_stat_lwlock	軽量ロックの内容ごとに1行の形で、発生に関する統計情報を示します。これらの情報は ボトルネックの解消につながる情報となります。 詳細については"C.2 pgx_stat_lwlock"を参照してください。
pgx_stat_latch	Symfoware Server内部の待機情報について、内容ごとに1行の形で、発生に関する統計 情報を示します。これらの情報はボトルネックの解消につながる情報となります。 詳細については"C.3 pgx_stat_latch"を参照してください。

ビュー名	説明
pgx_stat_walwriter	WALの書込みに関する統計情報を1行のみで表示します。詳細については"C.4 pgx_stat_walwriter"を参照してください。
pgx_stat_sql	SQL文の種類ごとに1行の形で、SQL文の発生回数に関する情報を示します。詳細については"C.5 pgx_stat_sql"を参照してください。

6.6.2 取得するための設定

情報内容に応じて、取得するための設定方法が異なります。

- PostgreSQL共通の情報
- Symfoware Serverが追加した情報

PostgreSQL共通の情報

💦 参照

PostgreSQL共通の情報については、"PostgreSQL文書"の"サーバの管理"の"データベース活動状況の監視"の"統計情報コレクタ" を参照してください。

Symfoware Serverが追加した情報

Symfoware Serverが追加した情報については、デフォルトで情報を収集します。

情報収集の有効/無効を変更する場合は、postgresql.conf内の設定パラメータを変更してください。以下に、情報収集の有効/無効を変更可能なビューと、設定パラメータを示します。

ビュー名	設定パラメータ
pgx_stat_lwlock	track_waits
pgx_stat_latch	
pgx_stat_sql	track_sql

備考) pgx_stat_walwriterは変更できません。

パラメータの詳細については、"付録A パラメータ"を参照してください。



Symfoware Server V12.0.0/V12.0.0Aで作成したインスタンスにおいて、Symfoware Serverが追加した情報を参照するためには、シス テムビューを追加してください。詳細は"導入ガイド(サーバ編)"の"Symfoware Server移行時の作業手順"を参照してください。

6.6.3 情報のリセット方法

情報のリセット方法を説明します。

Symfoware Serverが追加した情報

Symfoware Serverが追加した情報については、PostgreSQL共通の情報と同様に"pg_stat_reset_shared 関数"でリセットすることができます。

pg_stat_reset_shared関数に、以下のパラメータを設定します。

関数	戻り値の型	説明
pg_stat_reset_shared(text)	void	引数に応じて、クラスタ全体の統計情報カウン タの一部をゼロに戻します(スーパーユーザー 権限が必要です)。
		pg_stat_reset_shared('lwlock')を呼び出すと、 pgx_stat_lwlockで示される値すべてがゼロに なります。
		以下の場合も同様に、それぞれの統計情報カ ウンタの値すべてがゼロになります。
		• pg_stat_reset_shared('latch')を呼び出した 場合:
		pgx_stat_latchで示される値すべて
		• pg_stat_reset_shared('walwriter')を呼び出 した場合:
		pgx_stat_walwriterで示される値すべて
		• pg_stat_reset_shared('sql')を呼び出した場合:
		pgx_stat_sqlで示される値すべて



pg_stat_reset_shared関数のその他のパラメータについては、"PostgreSQL文書"の"サーバの管理"の"データベース活動状況の監 視"の"統計情報関数"を参照してください。

第7章 PL/extJavaの導入と運用

本章では、PL/extJavaの導入と運用について説明します。

7.1 PL/extJavaの概要

PL/extJavaは、データベースサーバ内にアプリケーションサーバを組み込み、Java Virtual Machine (JavaVM)を制御するフレームワー クです。

クライアントがデータベースサーバに対して、頻繁にSQLアクセスする業務では、クライアントとデータベースサーバとの間のネットワークトラフィックが増加するため、該当業務のレスポンスが低下するだけでなく、他の業務システムもレスポンスが低下します。

これらの業務処理時間を改善するには、クライアント側の業務ロジックをストアドファンクションとして、データベースサーバ上で処理させることでネットワークトラフィックを軽減できますが、多くのクライアントが同時アクセスする運用形態の場合、データベースサーバ側で JavaVMを実行すると、JavaVMの多重制御やメモリ資源の考慮が必要となります。

本機能は、上記のように、多くのクライアントが同時アクセスする運用形態の場合でも、アプリケーションサーバでJavaVMの多重制御 やメモリ資源を制御することで効率的にストアドファンクションを実行できます。

図7.1 PL/extJavaの概要



関 ポイント

本章では、データベースに登録したPL/extJavaのストアドファンクションを「Java関数」と呼びます。また、JavaVM上で動作するアプリ ケーションを「Javaアプリケーション」と呼びます。



- ・「Java関数」は、クライアントとデータベースサーバ間のコネクションと異なるコネクションで動作します。このため、「Java関数」に移行する業務はトランザクション単位で運用する必要があります。
- ・ PL/extJavaを使用するデータベースは、データベース名を28バイト以下にする必要があります。また、アプリケーションサーバから データベースに接続するためのロール名は8バイト以下にする必要があります。

7.1.1 PL/extJavaの構成

PL/extJavaの構成について説明します。

図7.2 PL/extJavaの構成



図中の各要素について、以下に説明します。

・ インスタンス

Symfoware Serverのインスタンスです。

クライアントからJava関数が呼び出された場合、Webサーバに対してJavaアプリケーションの実行を依頼します。 Java関数の登録方法は、"7.2.4 Java関数の登録"を参照してください。

・ Webサーバ

Webサーバは、Java関数からJavaアプリケーションの実行要求を受け取ります。 その要求に応じて、コンテナ内のサーバインスタンス(JavaVM)にJavaアプリケーションの実行を依頼します。 コンテナ内に複数のサーバインスタンス(JavaVM)が存在する場合、最適なサーバインスタンス(JavaVM)に振り分けます。

・コンテナ

コンテナは、Javaアプリケーションの実行環境です。

Java関数を実行するインスタンスのデータベースに1対1で対応します。

・ サーバインスタンス(JavaVM)

サーバインスタンス(JavaVM)は、1個のJavaVMです。サーバインスタンス(JavaVM)で、Javaアプリケーションを同時に実行できます。

Javaアプリケーションの最大同時実行数は、サーバインスタンス(JavaVM)を追加することにより拡張できます。

・ドメイン

ドメインは、コンテナを一元的に管理します。

ドメインは、インスタンスに1対1で対応します。ドメインの資源を格納するディレクトリを「ドメインルート」と呼びます。

7.1.2 使用できるアプリケーションサーバ

Symfoware Serverに組み込まれたInterstage Application Server技術をベースとしたアプリケーションサーバを利用することで、以下の効果を得ることができます。

(以降、Symfoware Serverに組み込まれたInterstage Application Server技術をベースとしたアプリケーションサーバを、「Symfoware Server Java アプリケーションサーバ」と呼びます。)

- Symfoware Server Java アプリケーションサーバの設定や制御を簡易化したコマンド(pgx_jadmin)を使用することができるため、 Symfoware Server Java アプリケーションサーバの詳細な知識がなくても、容易にセットアップやチューニングを行うことができます。
- ・以下のような Symfoware Server Java アプリケーションサーバの機能を、ユーザーが意識することなく使うことができるため、安定稼動を容易に実現することができます。
 - 多数のJava関数が実行された場合、最適化された数のJavaVM上でJavaアプリケーションを実行するため、必要最小限のメモリ資源で運用できます。
 - Javaのフル・ガーベジコレクション発生を予知して、当該のJavaVMを切り離すことにより、Javaアプリケーションの予期しない遅延を防止できます。
 - JavaVMのヒープ使用量やガーベジコレクションの発生頻度を監視して、危険な兆候をアラート通知できます。(予兆監視機能)
 - JavaVMの生死監視によりJavaアプリケーションの異常動作をロギングし、異常原因を効率的に調査できます。(タイマ機能)



Symfoware Server Java アプリケーションサーバを利用する場合、フレームワークは、以下の構成になります。



図中のSymfoware Server Java アプリケーションサーバ固有要素について、以下に説明します。

• Webサーバコネクタ

Webサーバコネクタは、Webサーバが受け付けたリクエストをコンテナに転送する役割を持っています。

・ PCMIサービス

Java EE DASサービス、サーバインスタンス(JavaVM)の管理を行うサービスです。 Java EE DASサービス、サーバインスタンス(JavaVM)を起動・停止、監視します。

・ Java EE DASサービス

PCMIサービスと連携して、コンテナの運用管理を行うサービスです。 pgx_jadminコマンドからの運用操作依頼を受け付けます。



Symfoware Server Java アプリケーションサーバを使用する場合、以下の点に注意してください。

- ・ドメインは、1台のマシンに対して2個以上作成できないため、PL/extJavaを利用するデータベースクラスタは、1台のマシンに対して 2個以上作成できません。
- ・1個のサーバインスタンス(JavaVM)が同時に実行できるJavaアプリケーションは64個までです。これ以上の多重度で実行する場合には、pgx_jadminコマンドでサーバインスタンス(JavaVM)を随時追加してください。

7.1.3 ユーザー定義

PL/extJavaで必要となるユーザーについて説明します。

・ OSの管理者ユーザー (OSのデフォルトはAdministrator)

```
OSの管理者ユーザーが、ドメインの管理者となります。また、OSの管理者ユーザーが、アプリケーションサーバのコンテナを構成
するプロセスの実効ユーザーとなるため、Javaアプリケーションによるファイルアクセスなどは、OSの管理者ユーザーの権限で行わ
れます。
```

Symfoware Serverに組み込まれたSymfoware Server Java アプリケーションサーバの機能を利用する場合

- コンテナやpgx_jadminコマンドの実効ユーザーです。
- 本ユーザーは、「pgx_jadmin init-domain」の「--dbadminuser」オプションで指定します。
- データベースに接続するユーザー

Javaアプリケーションが、データベースにアクセスする場合、JDBCコネクションプールを利用します。その認証ユーザーです。

CREATE USER文を使用して、データベースに接続するユーザーを追加してください。

Symfoware Server Java アプリケーションサーバを利用する場合、"7.2.3.1 データベースクラスタの設定"を参照してください。

7.2 PL/extJavaの導入

PL/extJavaの導入について説明します。

Symfoware Server Java アプリケーションサーバを利用する場合、以下の手順でPL/extJavaをセットアップしてください。

G 注意

- ・ PL/extJavaは、WebAdmin、およびpgAdminを使用してセットアップできません。
- ・ IPv6/IPv4デュアルスタック環境での運用のみをサポートします。IPv4を無効にした場合の運用はサポートしません。
- Symfoware Serverのインストール先のパス名の長さが、50バイトを超えるとJava関数の呼び出しに失敗します。エラーメッセージ (16551)が通知され、HTTP status codeが404となっている場合は、イベントログのエラーメッセージを確認してください。エラーメッ セージ(IJServer12042)が出力されている場合は、Symfoware Serverのインストール先のパス名の長さを50バイト以内にしてインス トールし直してください。

セットアップの流れ

PL/extJavaのセットアップは、以下の手順で実施します。

事前にデータベースクラスタを作成してください。

- 1. ポート番号の準備
- 2. ドメインの作成
- 3. PL/extJavaの作成
- 4. Java 関数の登録

7.2.1 ポート番号の準備

PL/extJavaで使用するポート番号について説明します。

ドメイン管理用のポート番号

用途	デフォルト	指定方法	説明
	ポート番号		
ドメイン管理用のポート番号	26530	pgx_jadmin init-domainまたは	ドメイン管理用のポートです。
	26521	pgx_jadmin modify-domain-port	ポート番号は、3個必要です。
	26522		



pgx_jadmin init-domain の詳細については、"7.2.2ドメインの作成"を参照してください。

pgx_jadmin modify-domain-portの詳細については、"7.3.3.3 ポート番号の変更"を参照してください。

サーバインスタンス(JavaVM)管理用のポート番号

サーバインスタンス(JavaVM)単位に以下のポート番号が必要です。

用途	月途 デフォルト 指定方法		説明
	ポート番号		
サーバインスタンス(JavaVM)管理	26531	pgx_jadmin create-container	サーバインスタンス(JavaVM)を操作するために使
用のポート番号	26532	instanceport,	用します。
		pgx_jadmin add-instance	ポート番号は、サーバインスタンス(JavaVM)単位に 2個設定する必要があります。
		instanceport、または	
		pgx_jadmin modify-instance- portinstanceport	

💦 参照

pgx_jadmin create-container の詳細については、"7.2.3.2 コンテナの作成"を参照してください。

pgx_jadmin add-instance の詳細については、"7.3.3.1 サーバインスタンス(JavaVM)の追加と削除"を参照してください。

pgx_jadmin modify-instance-portの詳細については、"7.3.3.3 ポート番号の変更"を参照してください。

G 注意

- ポート番号は、以下の範囲の未使用のポート番号を指定してください。
 - Windows Server(R) 2003またはWindows Server(R) 2003 R2の場合:5001~49151
 - Windows Server(R) 2008、Windows Server(R) 2008 R2、Windows Server(R) 2012、またはWindows Server(R) 2012 R2の場合:1024~49151
- ファイアウォールの設定

pgx_jadminコマンドに指定したポート番号は、ローカルマシン内の通信で使用します。ファイアウォールを利用して、各ポート番号 へのアクセスを制限することにより、不正アクセスを防ぎセキュリティを確保してください。

7.2.2 ドメインの作成

PL/extJavaを利用するためにドメインを作成します。

1. インスタンスの停止

インスタンスを停止します。

停止方法については、"2.1 インスタンスの起動と停止"を参照してください。

インスタンスの停止に失敗した場合は、"8.11インスタンス停止失敗時の対処"を参照してください。

2. hostsファイルの編集

hostsファイルにPL/extJavaを構築するホストの情報を追加します。

3. ドメインの作成

pgx_jadminコマンドのinit-domainサブコマンドで、ドメインを作成します。



ドメインルートが「C:¥domain」の場合

> pgx_jadmin init-domain --shareddir C:¥domain --domainport 26530,26521,26522 --pgdata c:¥database¥inst1



init-domainサブコマンドの詳細については、"リファレンス"の"pgx_jadmin"を参照してください。

.....

7.2.3 PL/extJavaの作成

PL/extJavaの作成について説明します。

7.2.3.1 データベースクラスタの設定

データベースクラスタの定義を設定します。

1. postgresql.confファイル内の"listen_addresses"の設定

Javaアプリケーションがデータベースに接続する場合には、ループバックアドレスを使用して接続します。そのため、"listen_addresses"は、localhostからの接続を許可する設定にしてください。



詳細については、"PostgreSQL文書"の"サーバの管理"の"接続と認証"を参照してください。

2. データベースに接続するユーザーの追加

Javaアプリケーションは、データベースにアクセスする場合、JDBCコネクションプールを利用します。

CREATE USER文を使用して、データベースに接続するユーザーを追加してください。追加したユーザー名は、コンテナを作成 するときに指定します。

🄰 例

.

ユーザー名が「user01」の場合

db01=# create user user01;

3. pg hba.conf ファイルの設定

Javaアプリケーションからデータベースへローカルで接続します。ローカル接続を許可する設定にしてください。

パラメータ名	パラメータの設定値
ТҮРЕ	host
DATABASE	Javaアプリケーションが接続するデータベース名
USER	Javaアプリケーションがデータベースに接続するときに使用するユーザー名
ADDRESS	127.0.0.1/32
AUTH-METHOD	trust認証以外を推奨

💦 参照

```
詳細については、"PostgreSQL文書"の"サーバの管理"の"pg_hba.confファイル"を参照してください。
```

4. postgresql.conf ファイル内の "max_connections"の設定

Symfoware Server Java アプリケーションサーバからデータベースに接続するときに使用する以下の接続数を、既存のmax_connectionsに加算してください。

サーバインスタンス (JavaVM) 数×64

💦 参照

```
詳細については、"導入ガイド(サーバ編)"の"リモート接続の設定"を参照してください。
```

7.2.3.2 コンテナの作成

コンテナは、Java関数を利用するデータベースクラスタ内のデータベースに対して1個作成します。

コンテナを作成すると、コンテナ内に1個のサーバインスタンス(JavaVM)が同時に作成されます。また、Javaアプリケーションからデー タベースにアクセスするためのJDBCコネクションプールやデータソースの設定も行われます。なお、サーバインスタンス(JavaVM)名 は、"データベース名-通番"となります。サーバインスタンス(JavaVM)名は、pgx_jadminコマンドのlist-containerサブコマンドで確認で きます。

💦 参照

list-containerサブコマンドの詳細は、"リファレンス"の"pgx_jadmin"を参照してください。

1. インスタンスの停止

インスタンスを停止します。

停止方法については、"2.1 インスタンスの起動と停止"を参照してください。

インスタンスの停止に失敗した場合は、"8.11インスタンス停止失敗時の対処"を参照してください。

2. コンテナの作成

pgx_jadminコマンドのcreate-containerサブコマンドにより、コンテナを作成します。データベース接続に必要な情報は、後から変更することもできます。





create-containerサブコマンドの詳細は、"リファレンス"の"pgx_jadmin"を参照してください。

- サーバインスタンス(JavaVM)の追加 Javaアプリケーションを同時に65個以上実行したい場合、必要数分のサーバインスタンス(JavaVM)を追加してください。 サーバインスタンス(JavaVM)の追加方法は、"7.3.3.1 サーバインスタンス(JavaVM)の追加と削除"を参照してください。
- 4. インスタンスの起動

インスタンスを起動してください。

起動方法については、"2.1 インスタンスの起動と停止"を参照してください。

インスタンスの起動に失敗した場合は、"8.10インスタンス起動失敗時の対処"を参照してください。

5. PL/extJava拡張のインストール

CREATE EXTENSION文を実行してPL/extJavaの拡張をインストールします。

db01=# CREATE EXTENSION plextjavau;

7.2.4 Java関数の登録

Java関数の登録とJavaアプリケーションの格納手順について説明します。

本手順は、Java関数を作成するたびに行う必要があります。

Java関数の登録

1. データベースに接続

psqlコマンド、pgAdminなどを利用して、作成したコンテナに対応するデータベースにインスタンス管理者で接続します。

2. データベースにファンクションを登録

データベースにファンクションを登録します。

Java関数は呼び出し元のセッションとは独立したセッションで動作します。Java関数を呼び出し元のセッションで動作させる場合は、SET文でplextjava.separate_sessionパラメータを"off"にしてから、ファンクションを登録します。

SET plextjava.separate_session=off

ファンクションをデータベースへ登録する際に使用するCREATE FUNCTION文について説明します。

【構文】

```
CREATE [ OR REPLACE ] FUNCTION

name ( [ IN ] argtype [, ...] )

[ RETURNS rettype ]

AS 'definition'

LANGUAGE /ang_name
```

name

```
Java関数名を指定します。
```

Java関数名とJavaアプリケーションのメソッド名は、一致している必要はありません。

スキーマ名を修飾できます。事前にスキーマを作成してください。

argtype

Java関数の引数のデータ型を指定します。

rettype

Java関数の戻り値のデータ型を指定します。
definition

```
Javaアプリケーションの '[パッケージ名].[クラス名].[メソッド名] 'または '[クラス名].[メソッド名] 'を指定します。
```

メソッド名にメソッドの引数を指定しても無効になります。

lang_name

"plextjavau"を指定します。

酁 例

スキーマ名が「javatest」、Java関数名が「java_addOne」、Javaアプリケーションの[パッケージ名].[クラス名].[メソッド名]が

「org.postgresql.plextjava.example.Parameters.addOne」の場合

db01=# CREATE FUNCTION javatest.java_addOne(INTEGER) RETURNS INTEGER AS 'org.postgresql.plextjava.example.Parameters.addOne' LANGUAGE plextjavau;



ー データ型の詳細は"アプリケーション開発ガイド"の"Java関数のデータ型とJavaアプリケーションのデータ型の関係"を参照 してください。

- CREATE FUNCTIONコマンドの詳細は"PostgreSQL文書"の"SQLコマンド"の"CREATE FUNCTION"を参照してください。
- 3. データベースに接続するユーザーの権限設定

Javaアプリケーションからデータベースにアクセスするときに使用するユーザーに、アクセス対象となるデータベースオブジェクト (テーブル、列、ビューなど)に対する権限を付与します。権限の付与は、GRANT文を使用してください。

🄰 例

テーブル「table01」にデータを追加する権限をユーザー名「user01」に付与する場合

db01=# GRANT INSERT ON table01 to user01;

4. Java関数の権限設定

Java関数を呼び出すユーザーにJava関数の実行権限を付与します。権限の付与は、GRANT文を使用してください。



Java関数の実行権限は、Java関数がアクセスする資源にアクセスすることを許可されたユーザーに対してのみ、付与してください。その他のユーザーに対してJava関数の実行権限を付与すると、資源のアクセス権限を持たないユーザーがJava関数経由で 資源にアクセスすることを、不当に許可してしまいます。



Java関数が「java_addOne」、ユーザー名が「user04」の場合

db01=# GRANT EXECUTE ON FUNCTION javatest.java_addOne(INTEGER) to user04;

5. データベースと切断します。

.....



CREATE FUNCTION文、GRANT文の詳細については"PostgreSQL文書"の"SQLコマンド"を参照してください。

Javaアプリケーションの格納

1. Javaアプリケーションのコピー

Javaアプリケーションをjar形式で、以下のディレクトリに手動でコピーします。

[ドメインルート]¥plextjava¥[データベース名]

7.3 PL/extJavaの運用

PL/extJavaの運用について説明します。

Symfoware Server Java アプリケーションサーバを利用する場合、以降に説明する手順で運用してください。

7.3.1 コンテナの起動・停止

インスタンスが起動、停止するときに、コンテナが自動的に起動、停止します。

7.3.2 PL/extJavaの確認

PL/extJavaの確認方法について説明します。

7.3.2.1 ドメイン情報の確認

pgx_jadminコマンドのlist-domainサブコマンドで、ドメインの情報を確認します。



> pgx_jadmin list-domain

表示例

domain status: running shareddir: C:¥domain domainport: 26530,26521,26522 dbadminuser: symfo datadir: C:¥database¥inst1



list-domainサブコマンドの詳細は、"リファレンス"の"pgx_jadmin"を参照してください。

7.3.2.2 コンテナ情報の確認

pgx_jadminコマンドのlist-containerサブコマンドにより、コンテナの情報を確認します。

データベース名が「db01」の場合

> pgx_jadmin list-container --dbname db01

表示例

```
container status: running
dbname: db01
instance: db01-1 26531,26532
instance: db01-2 26801,26802
dbport: 26011
dbuser: user01
```



```
list-containerサブコマンドの詳細は、"リファレンス"の"pgx_jadmin"を参照してください。
```

7.3.3 PL/extJavaの変更

PL/extJavaを変更する方法について説明します。

7.3.3.1 サーバインスタンス (JavaVM) の追加と削除

サーバインスタンス(JavaVM)の追加、削除について説明します。

サーバインスタンス(JavaVM)は、1個のJavaVMです。1個のサーバインスタンス(JavaVM)で、最大64個のJavaアプリケーションを同時 に実行できます。

Javaアプリケーションの同時実行数は、サーバインスタンス(JavaVM)を追加することで拡張できます。また、Javaアプリケーションの同時実行数を小さくする場合、追加したサーバインスタンス(JavaVM)を削除します。

- 1. インスタンスの停止
 - インスタンスを停止します。

停止方法については、"2.1 インスタンスの起動と停止"を参照してください。

インスタンスの停止に失敗した場合は、"8.11 インスタンス停止失敗時の対処"を参照してください。

2. データベースクラスタの設定変更

インスタンスの最大同時接続数(max_connections)を変更します。

同時接続数の変更方法については、"7.2.3.1 データベースクラスタの設定"を参照してください。

3. サーバインスタンス(JavaVM)の追加と削除

pgx_jadminコマンドのadd-instance、またはdelete-instanceサブコマンドで、サーバインスタンス(JavaVM)を追加、または削除しま す。

🄰 例

[サーバインスタンス(JavaVM)の追加]

> pgx_jadmin add-instance --dbname db01 --instanceport 26801,26802

[サーバインスタンス(JavaVM)の削除]

> pgx_jadmin delete-instance --dbname db01



add-instance、およびdelete-instanceサブコマンドの詳細は、"リファレンス"の"pgx_jadmin"を参照してください。

4. インスタンスの起動

インスタンスを起動します。

起動方法については、"2.1 インスタンスの起動と停止"を参照してください。

インスタンスの起動に失敗した場合は、"8.10インスタンス起動失敗時の対処"を参照してください。

7.3.3.2 データベース接続情報の変更

データベースの接続情報の変更について説明します。

1. インスタンスの停止

インスタンスを停止します。

停止方法については、"2.1 インスタンスの起動と停止"を参照してください。

インスタンスの停止に失敗した場合は、"8.11 インスタンス停止失敗時の対処"を参照してください。

2. データベース接続情報の変更

pgx_jadminコマンドのmodify-container-dbサブコマンドで、データベース接続情報を変更します。

以下のデータベース接続情報を変更できます。

- インスタンスのポート番号
- データベースに接続するユーザー名
- データベースに接続するパスワード



データベースに対する接続ユーザー名を「user02」に変更する場合

> pgx_jadmin modify-container-db --dbname db01 --dbuser user02

💦 参照

modify-container-dbサブコマンドの詳細は、"リファレンス"の"pgx_jadmin"を参照してください。

3. インスタンスの起動

インスタンスの接続情報を更新した後、インスタンスを起動します。 起動方法については、"2.1 インスタンスの起動と停止"を参照してください。 インスタンスの起動に失敗した場合は、"8.10 インスタンス起動失敗時の対処"を参照してください。

7.3.3.3 ポート番号の変更

ドメイン管理用のポート番号、またはサーバインスタンス(JavaVM)管理用のポート番号の変更について説明します。

ドメイン管理用のポート番号の変更

- 1. インスタンスの停止
 - インスタンスを停止します。

停止方法については、"2.1 インスタンスの起動と停止"を参照してください。

インスタンスの停止に失敗した場合は、"8.11 インスタンス停止失敗時の対処"を参照してください。

2. 変更前のポート番号の確認

pgx_jadminコマンドのlist-domainサブコマンドで変更前のポート番号を確認します。

🎾 例

> pgx_jadmin list-domain

💦 参照

list-domainサブコマンドの詳細は、"リファレンス"の"pgx_jadmin"を参照してください。

3. ドメイン管理用ポート番号の変更

pgx_jadminコマンドのmodify-domain-portサブコマンドで、ポート番号を変更します。



ドメイン管理用のポート番号を「26600,26601,26602」に変更する場合

> pgx_jadmin modify-domain-port --domainport 26600, 26601, 26602



modify-domain-portサブコマンドの詳細は、"リファレンス"の"pgx_jadmin"を参照してください。

4. 変更後のポート番号の確認

pgx_jadminコマンドのlist-domainサブコマンドで、ポート番号が変更されたことを確認します。

5. インスタンスの起動

インスタンスを起動します。

起動方法については、"2.1 インスタンスの起動と停止"を参照してください。

インスタンスの起動に失敗した場合は、"8.10インスタンス起動失敗時の対処"を参照してください。

サーバインスタンス(JavaVM)管理用のポート番号の変更

1. インスタンスの停止

インスタンスを停止します。

停止方法については、"2.1 インスタンスの起動と停止"を参照してください。

インスタンスの停止に失敗した場合は、"8.11 インスタンス停止失敗時の対処"を参照してください。

2. 変更前のポート番号の確認

pgx_jadminコマンドのlist-containerサブコマンドで変更前のポート番号を確認します。





3. サーバインスタンス(JavaVM)のポート番号の変更

pgx_jadminコマンドのmodify-instance-portサブコマンドにより、ポート番号を変更します。

🎾 例

サーバインスタンス(JavaVM)管理用のポート番号を「26701,26702」に変更する場合

> pgx_jadmin modify-instance-port --instance db1-1 --instanceport 26701,26702

💦 参照

modify-instance-portサブコマンドの詳細は、"リファレンス"の"pgx_jadmin"を参照してください。

4. 変更後のポート番号の確認

.

pgx_jadminコマンドのlist-containerサブコマンドで、ポート番号が変更されたことを確認します。

5. インスタンスの起動

インスタンスを起動します。

起動方法については、"2.1 インスタンスの起動と停止"を参照してください。

インスタンスの起動に失敗した場合は、"8.10インスタンス起動失敗時の対処"を参照してください。

7.3.4 Java関数の削除

PL/extJavaからJava関数とJavaアプリケーションを削除する手順について説明します。

関 ポイント

インスタンスを起動した状態で、Java関数を削除できます。

Java関数の削除

1. データベースに接続

psqlコマンド、pgAdminなどを利用して、作成したコンテナに対応するデータベースにインスタンス管理者で接続します。

2. Java 関数の削除

データベースに登録したJava関数を削除します。

db01=# DROP FUNCTION Java関数名(引数);

3. データベースから切断します。

💦 参照

詳細は、"PostgreSQL文書"の"リファレンス"の"SQLコマンド"の"DROP FUNCTION"を参照してください。

Javaアプリケーションの削除

1. Javaアプリケーションの削除

以下のディレクトリのjarファイルを削除します。

[ドメインルート]¥plextjava¥[データベース名]

7.3.5 コンテナの削除

コンテナを削除する方法について説明します。

1. Java 関数の削除

作成したコンテナに対応するデータベースにインスタンス管理者で接続します。

データベースに登録したJava関数をすべて削除します。

詳細については、"7.3.4 Java関数の削除"を参照してください。

2. PL/extJava拡張のアンインストール

作成したコンテナに対応するデータベースにインスタンス管理者で接続します。

DROP EXTENSION文を実行してPL/extJavaの拡張をアンインストールします。

db01=# DROP EXTENSION plextjavau;

3. インスタンスの停止

インスタンスを停止します。

停止方法については、"2.1 インスタンスの起動と停止"を参照してください。

インスタンスの停止に失敗した場合は、"8.11 インスタンス停止失敗時の対処"を参照してください。

4. コンテナの削除

pgx_jadminコマンドのdelete-containerサブコマンドで、コンテナを削除します。



> pgx_jadmin delete-container --dbname db01

💦 参照

```
delete-containerサブコマンドの詳細は、"リファレンス"の"pgx_jadmin"を参照してください。
```

7.3.6 ドメインの削除

ドメインを削除する方法について説明します。



ドメインを削除した場合、[ドメインルート]ディレクトリが削除されます。Javaアプリケーションなど、このディレクトリ配下にあるファイルのう ち必要なものは、ドメインを削除する前に退避してください。

1. インスタンスの停止

インスタンスを停止します。

停止方法については、"2.1 インスタンスの起動と停止"を参照してください。

インスタンスの停止に失敗した場合は、"8.11 インスタンス停止失敗時の対処"を参照してください。

2. ドメインの削除

pgx_jadminコマンドのdelete-domainサブコマンドにより、ドメインを削除します。



🐴 参照

delete-domainサブコマンドの詳細は、"リファレンス"の"pgx_jadmin"を参照してください。

7.3.7 バックアップ/リストア

PL/extJavaのバックアップとリストアに関して説明します。

PL/extJavaのドメインやコンテナが配置されたディスクの故障などに備えて、PL/extJavaのバックアップを推奨します。

バックアップ契機としては、以下が想定されます。

- ・ PL/extJavaを変更した場合(コンテナ、サーバインスタンス(JavaVM)など)
- ・Java関数を追加、変更、削除した場合



システムディスクが故障した後、インスタンス管理者を再作成するときには、故障前と同じユーザー名にしてください。ユーザー名が異 なると、バックアップデータと矛盾するため、リストアできません。

7.3.7.1 バックアップ方法

PL/extJavaのバックアップ手順について説明します。

フェイルオーバ運用を利用する場合、バックアップは運用系で取得してください。

データベース多重化運用を利用する場合、バックアップはプライマリサーバで取得してください。

1. インスタンスの停止

インスタンスを停止します。

停止方法については、"2.1 インスタンスの起動と停止"を参照してください。

インスタンスの停止に失敗した場合は、"8.11インスタンス停止失敗時の対処"を参照してください。

2. PL/extJavaのバックアップ格納ディスクの用意

PL/extJavaの資源が格納されているディスクと異なるディスク装置を用意します。ドメインルートの2倍以上の容量があるディスクを用意してください。

3. PL/extJavaのバックアップ

OSの管理者ユーザーでpgx_jadminコマンドのbackupサブコマンドを実行して、PL/extJavaの資源をバックアップします。



PL/extJavaのバックアップ格納ディレクトリには、下記のディレクトリを指定しないでください。

- インスタンスのデータ格納先、バックアップ格納先およびドメインルートと同じディレクトリ
- インスタンスのデータ格納先、バックアップ格納先およびドメインルート配下のディレクトリ
- インスタンスのデータ格納先、バックアップ格納先およびドメインルートが、PL/extJavaのバックアップ格納ディレクトリの配下 となるディレクトリ



PL/extJavaのバックアップ格納ディレクトリが「D:¥backup」の場合

> pgx_jadmin backup --backupdir D:\backup

🐴 参照

```
backupサブコマンドの詳細は、"リファレンス"の"pgx_jadmin"を参照してください。
```

7.3.7.2 リストア方法

PL/extJavaをバックアップした時点にリストアする方法について説明します。

1. インスタンスの停止

インスタンスを停止します。

停止方法については、"2.1 インスタンスの起動と停止"を参照してください。

インスタンスの停止に失敗した場合は、"8.11 インスタンス停止失敗時の対処"を参照してください。

2. インスタンスのリストア

インスタンスのリストアが必要な場合、インスタンスをリストアします。

3. PL/extJavaのリストア

OSの管理者ユーザーでpgx_jadminコマンドのrestoreサブコマンドを実行して、PL/extJavaの資源をリストアします。



PL/extJavaのバックアップ格納ディレクトリが「D:¥backup」の場合

> pgx_jadmin restore --backupdir D:¥backup



restoreサブコマンドの詳細は、"リファレンス"の"pgx_jadmin"を参照してください。

.....

第8章 異常時の対処

Symfoware Serverの運用中における、データベースやアプリケーションに異常が発生した場合の対処方法について説明します。

異常の種類に応じて、データベースクラスタをリカバリする必要があります。リカバリは、以下の資源をリカバリします。

- データ格納先
- ・ トランザクションログ格納先(データ格納先と別のディスクに格納している場合)
- バックアップデータ格納先

🌀 注意

実際にはディスクが故障していなくても、故障している場合と同じ入出力異常のメッセージが出力されることがあります。これらは、それ ぞれリカバリの方法が異なります。

ハードウェアの状態を確認して、以下のいずれかを選択してください。

・ ディスクが故障している場合

"8.1 ディスク障害(ハードウェア)からのリカバリ"を参照して対処してください。

・ ディスクが故障していない場合

"8.13 ディスク障害以外の入出力異常"を参照して対処してください。

例えば、以下のような場合があります。

- 外部ディスクとの間のネットワーク異常
- 電源未投入やマウントによる異常

異常原因の特定方法

異常が発生した場合は、WebAdminのメッセージやイベントログおよびサーバログを参照して異常の原因を特定してください。

💦 参照

サーバログの詳細は、"導入ガイド(サーバ編)"の"設定パラメータ"を参照してください。

リカバリの目安時間

各ディレクトリ配下にある資源のリカバリ目安時間の導出式を示します。

・ データ格納先、またはトランザクションログ格納先

リカバリ時間 = (データ格納先の使用量 + トランザクションログ格納先の使用量)÷ ディスク書込み性能 × 1.5

- データ格納先の使用量:データベースクラスタのディスク使用量
- トランザクションログ格納先の使用量:トランザクションログをデータベースクラスタの外に格納している場合のトランザクションログが消費しているディスク使用量
- ディスク書込み性能:運用を行うシステム環境における、1秒間あたりに書き込み可能な最大のデータ量(バイト/秒)の実測値
- 1.5:最も時間のかかるディスク書き込み以外の時間を見込んだ係数

• バックアップデータ格納先

リカバリ時間 = バックアップデータ格納先の使用量 ÷ ディスク書込み性能 × 1.5

- バックアップデータ格納先の使用量:バックアップデータのディスク使用量
- ディスク書込み性能:運用を行うシステム環境における、1秒間あたりに書き込み可能な最大のデータ量(バイト/秒)の実測値
- 1.5:最も時間のかかるディスク書き込み以外の時間を見込んだ係数

8.1 ディスク障害(ハードウェア)からのリカバリ

データ格納ディスク、またはバックアップデータ格納ディスクでハードウェア障害が発生した場合に、障害が発生する直前の状態にデー タベースクラスタをリカバリする方法について説明します。

以下の2つの方法でリカバリできます。

- 8.1.1 WebAdminを使用する場合
- ・8.1.2 サーバコマンドを使用する場合

関 ポイント

データベースクラスタをリカバリした後、バックアップすることを推奨します。バックアップすることで、不要になったアーカイブログ(バック アップデータ格納先にコピーされたトランザクションログ)が削除されるため、ディスク容量の確保やリカバリ時間の短縮につながります。

8.1.1 WebAdminを使用する場合

障害が発生したディスクに応じて、以下のリカバリ手順でデータベースクラスタをリカバリしてください。

データ格納ディスク、またはトランザクションログ格納ディスクに障害が発生した場合

以下の手順でデータ格納ディスク、またはトランザクションログ格納ディスクをリカバリしてください。

1. アプリケーションの停止

データベースを利用しているアプリケーションを停止してください。

2. インスタンスの停止

インスタンスを停止します。停止方法については、"2.1.1 WebAdminを使用する場合"を参照してください。インスタンスを停止せずにデータベースクラスタのリカバリをした場合は、WebAdminが自動的にインスタンスを停止します。

3. 障害ディスクのリカバリ

ディスク交換を行ったあと、ボリュームの構成情報をリカバリします。

4. テーブル空間のディレクトリ作成

バックアップを実施した以降にテーブル空間を定義している場合、そのディレクトリを作成します。

5. キーストアのリカバリとキーストアの自動オープンの有効化

データベースに格納するデータを暗号化している場合は、以下を行ってください。

- データベースのバックアップを取得した時点のキーストアをリストアしてください。
- キーストアの自動オープンを有効にしてください。
- 6. データベースクラスタのリカバリ

WebAdminにログインし、[監視]メニュー画面のエラーメッセージの[対処]ボタンをクリックします。

Symfoware			ユーザー: symfo ログアウト FUITSU
監視 設定			₽ EE 372-274.
インスタンス一覧	インスタンス名: inst1		
新規作成			
inst1			Trut
	インスタンス	データ格納先	バックアップデータ格納先
	稼働状態: 停止	状態: 異栄	状態: 正常
	 インスタンス名: inst1 ボート番号: 26500 エンコード: UTF8	パス: D.WsymfoWinst1Wdata	パス: E¥symfo¥inst1¥backup
	起動	データ復日 ひらックア	ップ日時> 2013-10-17 15:14
	▼ メッセージ		
	(1) 2013-10-17 16:09:50 データ格線	特先の資源に異常を検出しました。	<u> </u>
	<u>▲</u> 2013-10-17 16:10:04 インスタン	スに問題が発生しました。対処ボタンを押してく	たさい。 2010年1月20日 2010年1月1日 2010年1月11日 2010年1月110月11日 2010年1月110月110月110月110月110月110月110月110月11

7. リカバリの実行

[リカバリ]ダイアログが表示されますので、[実行]ボタンをクリックしてください。

[監視]メニュー画面に「リカバリ中」と表示され、リカバリが実行されます。リカバリが正常に完了すると、インスタンスが自動的に起動されます。





WebAdminは、ハッシュインデックスのリカバリをサポートしません。ハッシュインデックスを使用している場合には、リカバリの実行後に、REINDEXコマンドで該当するハッシュインデックスを再構築してください。なお、ハッシュインデックスの使用は推奨されていません。

8. アプリケーションの再開

データベースを利用しているアプリケーションを再開してください。

関 ポイント

ディスク異常の発生の仕方によってはWebAdminで異常が検出できない場合があります。

その場合は"8.10.4 その他の異常"を参照してリカバリしてください。

バックアップデータ格納ディスクに障害が発生した場合

以下の手順でバックアップデータ格納ディスクをリカバリしてください。

1. 障害ディスクのリカバリ

ディスク交換を行ったあと、ボリュームの構成情報をリカバリします。

2. バックアップデータのリカバリ

WebAdminにログインし、[監視]メニュー画面のエラーメッセージの[対処]ボタンをクリックします。

*Symfoware			ユーザー: symfo <u>ログアウト</u>	ໜຶ່ກຮຸ
監視 設定			😌 🎩 📓 🛛	TaZIk.
インスタンス一覧	インスタンス名: inst1			
新規作成		(
inst1		1500		
	インスタンス	データ格納先	バックアップデータ格納先	
	稼働状態: 停止	状態: 正常	状態: 異常	
	 インスタンス名: inst1 ポート番号: 20500 エンコード: UTF8	バス: D.WsymdoWinst1Wdata	バス: E¥symfo¥inst1¥backup	
	起動	デー頻度目 ひいっつアっ	プ日時> 2013-10-17 1458	
	▼ メッセージ			
	 2013-10-17 15:07:52 バックアップ 	データ格納先の資源に異常を検出しました。		<u>_</u>
	A 2013-10-17 15:08:56 インスタンス	に問題が発生しました。対処ボタンを押してくた	さい。 約 対処	2
		[yty	<u>ル]</u> をクリック	~

3. バックアップの実行

バックアップデータをリカバリするために、バックアップを実行します。[バックアップ]ダイアログが表示されますので、[実行]ボタ ンをクリックしてください。[監視]メニュー画面に「バックアップ中」と表示され、バックアップが実行されます。バックアップを実行す ると、インスタンスが自動的に起動されます。

バックアップ 利用可能	1/23/2/2	バック 「 データ格納先」	フ アップ
		状態を再確認する	実行 キャンセル



[状態を再確認する]をクリックした場合、データ格納先およびバックアップデータ格納先の資源を再度検証します。その結果により、以

「のようになります。

• 異常を検出しなかった場合

データ格納先およびバックアップデータ格納先の状態が「正常」に戻り、通常の操作を行えるようになります。

• 異常を検出した場合

メッセージリストに再度エラーメッセージが表示されます。[対処]ボタンをクリックし、ダイアログに記されているエラー原因の解決方 法にしたがって問題を解決してください。

8.1.2 サーバコマンドを使用する場合

障害が発生したディスクに応じて、以下のリカバリ手順でデータベースクラスタをリカバリしてください。

データ格納ディスク、またはトランザクションログ格納ディスクに障害が発生した場合

以下の手順でデータ格納ディスク、またはトランザクションログ格納ディスクをリカバリしてください。

1. アプリケーションの停止

データベースを利用しているアプリケーションを停止してください。

2. インスタンスの停止

インスタンスを停止します。停止方法については、"2.1.2 コマンドを利用する場合"を参照してください。 インスタンスの停止に失敗した場合は、"8.11 インスタンス停止失敗時の対処"を参照してください。

3. 障害ディスクのリカバリ

ディスク交換を行ったあと、ボリュームの構成情報をリカバリします。

- 4. 格納先のディレクトリの作成
 - データ格納ディスクに障害が発生した場合
 データ格納先のディレクトリを作成します。テーブル空間を定義していた場合、そのディレクトリも作成します。
 - トランザクションログ格納ディスクに障害が発生した場合
 トランザクションログ格納先のディレクトリを作成します。

格納先のディレクトリは、Windows(R)のエクスプローラの[プロパティ]を使用して、インスタンス管理者のみがアクセスできるように許可を設定します。([プロパティ]の詳細は、Windows(R)の[ヘルプとサポート]を参照してください。)

💦 参照

格納先のディレクトリの作成については、"導入ガイド(サーバ編)"の"セットアップ"の"資源配置用のディレクトリの準備"を参照 してください。

5. キーストアのリカバリとキーストアの自動オープンの有効化

データベースに格納するデータを暗号化している場合は、データベースのバックアップを取得した時点のキーストアをリストアしてください。キーストアの自動オープンは、必要に応じて設定してください。

6. データベースクラスタのリカバリ

バックアップデータを使用して、データベースクラスタをリカバリします。

pgx_rcvallコマンドには、以下を指定します。

- -Dオプションは、データ格納先のディレクトリを指定します。-Dオプションを省略した場合、PGDATA環境変数の値が使用されます。
- -Bオプションは、バックアップデータ格納先のディレクトリを指定します。

例

> pgx_rcvall -D D:¥database¥inst1 -B E:¥backup¥inst1

G 注意

リカバリに失敗した場合は、表示されるエラーメッセージに従ってエラーの原因を取り除き、pgx_rcvallコマンドを再実行してください。

「pgx_rcvall:リカバリ中にエラーが発生しました」のメッセージがある場合は、その後にリカバリ実行時のログが出力されます。ロ グの最後の十数行以内にエラーの原因が出力されますので、そのメッセージに従ってエラーの原因を取り除き、pgx_rcvallコマ ンドを再実行してください。

また、リカバリ中に表示される以下のメッセージは、pgx_rcvallコマンドの正常な動作によって出力されるメッセージのため、ユーザーが意識する必要はありません。

FATAL: データベースシステムは起動しています

7. インスタンスの起動

インスタンスを起動します。インスタンスの起動方法については、"2.1.2 コマンドを利用する場合"を参照してください。

8. アプリケーションの再開

データベースを利用しているアプリケーションを再開してください。

バックアップデータ格納ディスクに障害が発生した場合

バックアップデータ格納ディスクをリカバリする手順を説明します。 対処方法には以下の2つの方法があります。

- ・ インスタンスを起動したままリカバリする方法
- ・ インスタンスを停止してリカバリする方法

インスタンス停止の有無による、実行する手順の違いを以下に示します。

No	手順	インスタンス	停止の有無
		なし	あり
1	トランザクションログ二重化の停止確認	0	—
2	アーカイブログ出力の停止	0	_
3	アプリケーションの停止	—	0

No	手順	インスタンス停止の有無		
		なし	あり	
4	インスタンスの停止	_	0	
5	障害ディスクのリカバリ	0	0	
6	バックアップデータ格納先のディレクトリ作成	0	0	
7	アーカイブログ出力の再開	0	—	
8	トランザクションログ二重化の再開	0	_	
9	インスタンスの起動	_	0	
10	バックアップの実行	0	0	
11	アプリケーションの再開	—	0	

○:実施

-:実施不要

手順を以下に示します。

インスタンスを起動したままリカバリする方法

1. トランザクションログ二重化の停止確認

トランザクションログの二重化が停止されているかを、次のSQL関数で確認します。

postgres=# SELECT pgx_is_wal_multiplexing_paused();
pgx_is_wal_multiplexing_paused
-----t
(1 row)

トランザクションログの二重化が停止していない場合は、次のSQL関数で停止してください。

postgres=# SELECT pgx_pause_wal_multiplexing(); LOG: multiplexing of transaction log files has been stopped pgx_pause_wal_multiplexing

(1 行)

2. アーカイブログ出力の停止

バックアップ格納ディスクを交換するまでに時間がかかると、トランザクションログが蓄積されていきます。そして、データ格納 ディスクまたはトランザクションログ格納ディスクが満杯になると、運用が継続できなくなる危険性があります。

これを回避するために、以下の方法でアーカイブログの出力を停止してください。

- archive_commandの変更

アーカイブログを出力したものとみなすよう、必ず正常終了するコマンドを指定します。

echoを指定すると、サーバログにメッセージが出力されるため、調査を行う上での指標になる可能性があります。

- 設定ファイルの再読込み

pg_ctl reloadコマンド、またはSQL関数pg_reload_confを実行します。

なお、運用が継続できなくなる危険性が無く、単にエラーを出力させたくない場合は、archive_commandに空文字列(")を指定して設定ファイルの再読み込みを実施してください。

3. 障害ディスクのリカバリ

ディスク交換を行ったあと、ボリュームの構成情報をリカバリします。

4. バックアップデータ格納先のディレクトリ作成

バックアップデータ格納先を作成します。

バックアップデータ格納先のディレクトリは、Windows(R)のエクスプローラの[プロパティ]でインスタンス管理者のみがアクセ スできるように許可を設定します。([プロパティ]の詳細は、Windows(R)の[ヘルプとサポート]を参照してください。)

バックアップデータ格納先のディレクトリの作成については、"3.2.2 サーバコマンドを使用する場合"を参照してください。

5. アーカイブログ出力の再開

archive_commandの設定を元に戻し、設定ファイルを再読込みします。

6. トランザクションログ二重化の再開

SQL関数pgx_resume_wal_multiplexingを実行します。

例

SELECT pgx_resume_wal_multiplexing()

7. バックアップの実行

```
pgx_dmpallコマンドを使用して、データベースクラスタをバックアップします。
```

pgx_dmpallコマンドには、以下を指定します。

- -Dオプションは、データ格納先のディレクトリを指定します。-Dオプションを省略した場合、PGDATA環境変数の値が使用されます。

例

> pgx_dmpall -D D:\u00e4database\u00e4inst1

インスタンスを停止してリカバリする場合

1. アプリケーションの停止

データベースを利用しているアプリケーションを停止してください。

2. インスタンスの停止

インスタンスを停止します。停止方法については、"2.1.2 コマンドを利用する場合"を参照してください。 インスタンスの停止に失敗した場合は、"8.11 インスタンス停止失敗時の対処"を参照してください。

3. 障害ディスクのリカバリ

ディスク交換を行ったあと、ボリュームの構成情報をリカバリします。

4. バックアップデータ格納先のディレクトリ作成

バックアップデータ格納先を作成します。

バックアップデータ格納先のディレクトリは、Windows(R)のエクスプローラの[プロパティ]でインスタンス管理者のみがアクセ スできるように許可を設定します。([プロパティ]の詳細は、Windows(R)の[ヘルプとサポート]を参照してください。) 詳細については、"3.2.2 サーバコマンドを使用する場合"を参照してください。

5. インスタンスの起動

インスタンスを起動します。起動方法については、"2.1.2 コマンドを利用する場合"を参照してください。

6. バックアップの実行

pgx_dmpallコマンドを使用して、データベースクラスタをバックアップします。

pgx_dmpallコマンドには、以下を指定します。

- -Dオプションは、データ格納先のディレクトリを指定します。-Dオプションを省略した場合、PGDATA環境変数の値が使用されます。

例

> pgx_dmpall -D D:¥database¥inst1

7. アプリケーションの再開

データベースを利用しているアプリケーションを再開してください。



- ・ pgx_rcvallコマンド、およびpgx_dmpallコマンドの詳細は、"リファレンス"の"pgx_rcvall"および"pgx_dmpall"を参照してください。
- ・ archive_modeの詳細は、"PostgreSQL文書"の"サーバの管理"の"ログ先行書き込み(WAL)"を参照してください。

・ pgx_resume_wal_multiplexingの詳細は、"B.1 WAL二重化制御関数"を参照してください。

8.2 データ破壊からのリカバリ

ディスク内のデータが論理的に破壊され、データベースが正常に動作しない場合、バックアップを取得した時点の状態にデータベー スクラスタをリカバリできます。

以下の2つの方法でリカバリできます。

- 8.2.1 WebAdminを使用する場合
- 8.2.2 pgx_rcvallコマンドを使用する場合



 データベースクラスタをリカバリした後、バックアップすることを推奨します。バックアップすることで、不要になったアーカイブログ (バックアップデータ格納先にコピーされたトランザクションログ)が削除されるため、ディスク容量の確保やリカバリ時間の短縮につ ながります。 過去の時点に復旧した場合、その復旧時点を起点とする新たな時系列(データベース更新の歴史)が始まります。リカバリが完了したときには、その復旧時点が新たな時系列における最新地点です。以後、最新状態にリカバリする場合には、この新たな時系列上のデータベース更新が再実行されます。

8.2.1 WebAdminを使用する場合

WebAdminを使用する場合、バックアップデータを利用してデータ破壊直前にリカバリしてください。

詳細は、"8.1.1 WebAdminを使用する場合"を参照してください。

8.2.2 pgx_rcvallコマンドを使用する場合

pgx_rcvallコマンドでバックアップ取得日時を指定してデータベースクラスタをリカバリします。そのあと、必要に応じてトランザクションを 再実行し、データを復旧します。

以下の手順でデータ格納ディスクをリカバリしてください。

1. アプリケーションの停止

データベースを利用しているアプリケーションを停止してください。

2. インスタンスの停止

インスタンスを停止します。停止方法については、"2.1.2 コマンドを利用する場合"を参照してください。 インスタンスの停止に失敗した場合は、"8.11 インスタンス停止失敗時の対処"を参照してください。

3. バックアップ取得日時の確認

業務ログまたはイベントログの内容を元に、データが破壊される前の日時を特定します。

4. キーストアのリカバリとキーストアの自動オープンの有効化

データベースに格納するデータを暗号化している場合は、データベースのバックアップを取得した時点のキーストアをリストアしてください。キーストアの自動オープンは、必要に応じて設定してください。

5. データベースクラスタのリカバリ

pgx_rcvallコマンドを使用して、データベースクラスタを復旧します。

pgx_rcvallコマンドには、以下を指定します。

- -Dオプションは、データ格納先のディレクトリを指定します。-Dオプションを省略した場合、PGDATA環境変数の値が使用されます。
- -Bオプションは、バックアップデータ格納先のディレクトリを指定します。
- -eオプションは、復旧する日時を指定します。時間は秒単位で指定します。

例

復旧時点に '2013年7月20日 10時00分00秒 'を指定した場合の実行例を示します。

> pgx_rcvall -D D:¥database¥inst1 -B E:¥backup¥inst1 -e "2013-07-20 10:00:00"



リカバリに失敗した場合は、表示されるエラーメッセージに従ってエラーの原因を取り除き、pgx_rcvallコマンドを再実行してください。

「pgx_rcvall:リカバリ中にエラーが発生しました」のメッセージがある場合は、その後にリカバリ実行時のログが出力されます。ロ グの最後の十数行以内にエラーの原因が出力されますので、そのメッセージに従ってエラーの原因を取り除き、pgx_rcvallコマ ンドを再実行してください。

また、リカバリ中に表示される以下のメッセージは、pgx_rcvallコマンドの正常な動作によって出力されるメッセージのため、ユーザーが意識する必要はありません。

FATAL: データベースシステムは起動しています

6. インスタンスの起動

インスタンスを起動します。インスタンスの起動については、"2.1.2 コマンドを利用する場合"を参照してください。

必要に応じて、指定した復旧時点からトランザクション処理を再度実行した上で、データベースの運用を再開します。



pgx_rcvallコマンドは、ハッシュインデックスを正しく復旧できません。 ハッシュインデックスを使用している場合、インスタンス起動 後、該当するインデックスに対して、REINDEXコマンドを実行してください。

7. アプリケーションの再開

データベースを利用しているアプリケーションを再開してください。



pgx_rcvallコマンドの詳細は、"リファレンス"の"pgx_rcvall"を参照してください。

8.3 ユーザーの誤操作からのリカバリ

ユーザーの誤操作によりデータ破壊が発生した場合に、データベースクラスタをリカバリする方法について説明します。

以下の2つの方法でリカバリできます。

- 8.3.1 WebAdminを使用する場合
- 8.3.2 pgx_rcvallコマンドを使用する場合



 データベースクラスタをリカバリした後、バックアップすることを推奨します。バックアップすることで、不要になったアーカイブログ (バックアップデータ格納先にコピーされたトランザクションログ)が削除されるため、ディスク容量の確保やリカバリ時間の短縮につ ながります。

- 過去の時点に復旧した場合、その復旧時点を起点とする新たな時系列(データベース更新の歴史)が始まります。リカバリが完了したときには、その復旧時点が新たな時系列における最新地点です。以後、最新状態にリカバリする場合には、この新たな時系列上のデータベース更新が再実行されます。
- ・ 有効なリストアポイントは、バックアップを取得した時系列上で作成したものです。つまり、過去の時点に復旧した場合、以降に設定したリストアポイントは利用できません。したがって、望みの過去データを復元できたら、バックアップを取得してください。

8.3.1 WebAdminを使用する場合

WebAdminを使用してバックアップ時点にリカバリすることができます。 以下の手順でデータ格納ディスク内のデータをリカバリしてください。

1. アプリケーションの停止

データベースを利用しているアプリケーションを停止してください。

2. インスタンスの停止

インスタンスを停止します。停止方法については、"2.1.1 WebAdminを使用する場合"を参照してください。

3. キーストアのリカバリとキーストアの自動オープンの有効化

データベースに格納するデータを暗号化している場合は、以下を行ってください。

- データベースのバックアップを取得した時点のキーストアをリストアしてください。
- キーストアの自動オープンを有効にしてください。
- 4. データベースクラスタのリカバリ

WebAdminにログインし、[監視]メニュー画面の[データ復旧]ボタンをクリックします。

Symfoware			ユーザー: symfo ログアウト	FUJITSU
監視 設定			Q 🎫 📓 🛛	EaZlk.
- インスタンス一覧	- インスタンス名: inst1			
新規作成				
inst1		J.	לעידתע	
	インスタンス	データ格納先	バックアップデータ格納先	
	稼働状態: 停止	状態: 正常	状態: 正常	
	インスタンス名: inst1 ボート番号: 26500 エンコード: UTF8	バス: D¥symfo¥inst1¥data	バス: E¥symio¥inst1¥backup	
	起動	デーダ製田 いいっか	アップ日時> 2013-10-17 14:58	
	▼メッセージ	×//		~
	(1) 2013-10-17 1459:18 12X32X (1) 2013-10-17 1459:24 インスタンス	を停止しました。 【ラ	テータ復日]をクリック	
				X

5. バックアップ時点へのリカバリの実行

[リカバリ]ダイアログが表示されますので、[実行]ボタンをクリックしてください。

[監視]メニュー画面に「リカバリ中」と表示され、リカバリが実行されます。リカバリが正常に完了すると、インスタンスが自動的に起動されます。

リカバリ 利用不可能	ビ インスタンス	「 データ格納先	リカバリ ・ ・ ・ バックアップデータ 格納先
<u>注意</u> データベースクラスタをバック	アップ時点ヘリカバリします	ţ.	実行 キャンセル



WebAdminは、ハッシュインデックスを正しく復旧できません。ハッシュインデックスを使用している場合、リカバリの実行後、該当 するインデックスに対して、REINDEXコマンドを実行してください。

6. データベースの運用の再開

必要に応じて、バックアップ時点から誤操作を行うまでのトランザクション処理を再度実行したうえで、データベースの運用を再 開します。

8.3.2 pgx_rcvallコマンドを使用する場合

pgx_rcvallコマンドでは、サーバコマンドで作成したリストアポイントの時点にデータベースクラスタをリカバリします。リストアポイントの作成方法については、"3.2.2 サーバコマンドを使用する場合"の"リストアポイントの設定"を参照してください。

以下の手順でデータ格納ディスク内のデータをリカバリしてください。

1. アプリケーションの停止

データベースを利用しているアプリケーションを停止してください。

2. インスタンスの停止

インスタンスを停止します。停止方法については、"2.1.2 コマンドを利用する場合"を参照してください。 インスタンスの停止に失敗した場合は、"8.11 インスタンス停止失敗時の対処"を参照してください。

3. リストアポイントの確認

"3.2.2 サーバコマンドを使用する場合"で任意のファイルなどに記録したリストアポイントから、誤操作が行われる前のリストアポ イントを特定します。

4. キーストアのリカバリとキーストアの自動オープンの有効化

データベースに格納するデータを暗号化している場合は、データベースのバックアップを取得した時点のキーストアをリストアしてください。キーストアの自動オープンは、必要に応じて設定してください。

5. データベースクラスタのリカバリ

pgx_rcvallコマンドを使用して、データベースクラスタを復旧します。

pgx_rcvallコマンドには、以下を指定します。

- -Dオプションは、データ格納先のディレクトリを指定します。-Dオプションを省略した場合、PGDATA環境変数の値が使用されます。
- -Bオプションは、バックアップデータ格納先のディレクトリを指定します。
- -nオプションは、指定したリストアポイントの時点にデータをリカバリします。

例

リストアポイントが"batch_20130703_1"の場合の実行例を示します。

> pgx_rcvall -D D:¥database¥inst1 -B E:¥backup¥inst1 -n batch_20130703_1



リカバリに失敗した場合は、表示されるエラーメッセージに従ってエラーの原因を取り除き、pgx_rcvallコマンドを再実行してください。

「pgx_rcvall:リカバリ中にエラーが発生しました」のメッセージがある場合は、その後にリカバリ実行時のログが出力されます。ロ グの最後の十数行以内にエラーの原因が出力されますので、そのメッセージに従ってエラーの原因を取り除き、pgx_rcvallコマ ンドを再実行してください。

また、リカバリ中に表示される以下のメッセージは、pgx_rcvallコマンドの正常な動作によって出力されるメッセージのため、ユーザーが意識する必要はありません。

FATAL: データベースシステムは起動しています

6. インスタンスの起動

インスタンスを起動します。

インスタンスの起動方法については"2.1.2 コマンドを利用する場合"を参照してください。



pgx_rcvallコマンドは、ハッシュインデックスを正しく復旧できません。ハッシュインデックスを使用している場合、インスタンス起動後、該当するインデックスに対して、REINDEXコマンドを実行してください。

7. データベースの運用の再開

必要に応じて、指定した復旧時点から誤った操作を行うまでのトランザクション処理を再度実行した上で、データベースの運用 を再開します。



pgx_rcvallコマンドの詳細は、"リファレンス"の"pgx_rcvall"を参照してください。

8.4 アプリケーション異常の対処

長時間、待ち状態のクライアントのコネクションが存在する場合、原因となるコネクションを切断することで、データベースの性能劣化を 抑止できます。

以下の方法で切断するコネクションを特定できます。

- ・ビュー(pg_stat_activity)("8.4.1ビュー(pg_stat_activity)を使用する場合"参照)
- pgAdmin ("8.4.2 pgAdminを使用する場合"参照)

コネクションの切断は、システム管理関数(pg_terminate_backend)を使用します。

8.4.1 ビュー(pg_stat_activity)を使用する場合

ビュー(pg_stat_activity)を使用する場合、以下の手順でコネクションを切断します。

1. psqlコマンドで、データベースpostgresに接続します。

> psql postgres	
psql (9.2.8)	
Type "help" for help.	

2. 長時間、待ち状態のクライアントのコネクションを切断します。

pg_terminate_backend()を利用して、長時間接続中のコネクションを切断します。

ただし、アプリケーションの互換性を維持することを考慮し、下記のSQL文中のシステムカタログや関数を直接参照したり使用しないようにしてください。

例

以下の例は、クライアントの待ち状態が、60分以上の場合、コネクションを切断します。

select pid,usename,application_name,client_hostname,pg_terminate_backend(pid) from pg_stat_activity where							
state='idle in transaction' and current_timestamp > cast(query_start + interval '60 minutes' as timestamp);							
-[RECORD 1]+							
pid	4684						
usename	symfo						
application_name	apl1						
client_addr	slient_addr 192.11.11.1						
pg_terminate_backend	t						



・ pg_terminate_backendの詳細は、"PostgreSQL文書"の"SQL言語"の"システム管理関数"を参照してください。

 アプリケーションの互換性の維持に関する詳細は、"アプリケーション開発ガイド"の"アプリケーションの互換に関する注意事項" を参照してください。

.....

8.4.2 pgAdminを使用する場合

pgAdminを使用する場合、以下の手順でコネクションを切断します。

1. pgAdminの[ツール(T)]メニューの[サーバ状態(S)]をクリックします。

🌳 pgAdmin III		
ファイル(F) 編集(E) プラグイン(P) ビュー(V)	ルール有効?	
1 🔗 🦰 🔜 🗔 📄 🕅 👘	トリガ有効?	
	トリガの無効	
オブジェクトブラウザ × プロ	トリガ有効	
目 サーバグループ	Validate foreign key	
B sv1(sv1:26500)	Validate check constraint	en and an
in the second seco	Validate domain check constraint	
10月末	Slowed July-San	
	5101yレクラク - ショク	
	Disconnect conver	
(000 S)	Disconnect database	
🧰 S:	H-ビスの開始	
i 🕅 St		
(IIII)		
No.	Paduse replay of WAL	
[Add pamod restors point	
	Add named restore point	
	デバッグ(D)	•
	クエリーツール(Q) Ctrl+E	
	スクリプト	•
	データビュー(D)	·
and the second se	レポート(R)	
SQLCD	メンテナンス(M)	
	バックアップ(B)	
	全体バックアップ(B)	
	サーババックアップ(B)	
	リストアー(R)	
	Import	
	認可ウイザード(G)	
	サーバ構成	
		333
The second se	実行(R)	

2. 長時間、待ち状態のクライアントのコネクションを切断します。

「TX 開始」に表示されるトランザクション開始時間から、長時間、待ち状態のコネクションを選択します。その後、赤色の四角ボタンをクリックして、コネクションを切断します。

ith ID	アラリケーションダ	レーデーカベース	ユーザ	クライアント	05(70)8		クエリー開始	···· TX 開始		State	ei con State chang	e Jo	ت مىر م
1511 1550 1642	pgAdmin III pgAdmin III	db01 postgres db01	symfo symfo symfo	ocal pon 10.249.95.52: 10.249.95.52:	2013-07-1 2013-07-1 2013-07-1	9 18:09:00+09 9 18:11:44+09 9 18:19:33+09		2013-07-	19 18:19:57+09	de in transac ide ide	2013-07-19 2013-07-19 2013-07-19	18:19 18:11 18:33	SELE SELE
C Louis	a local level level		(and the second second						n Markada aya				
27													
PID	データベース	リレーション	2-9	XID	TX	方法	時町、ついている	開始 ()	019-				
1511			symfo	2/26	2/26	ExclusiveLock	(\$1)	2013-07	BEGIN;				
1642	db01	pg_class	symfo		5/558	AccessShar	はい	2013-07	SELECT pg_stat	_get_backend_	pid(svrid) AS pid, (SELECT datna	me FROM pg
1642	db01	pg_class	symfo		5/558	AccessShar	(\$1.)	2013-07	SELECT pg_stat	_get_backend_	pid(svrid) AS pid, (SELECT datna	me FROM pg
1642	db01	pg_locks	symfo		5/558	AccessShar	(\$1.)	2013-07	SELECT pg_stat	_get_backend_	pid(svrid) AS pid, (SELECT datna	me FROM pg
1642			symto	5/558	5/558	ExclusiveLock	(2()	2013-07	SELECT pg_stat	_get_backend_	pid(svrid) AS pid, (SELECT datna	me FROM pg
1643		pg_cata	symfo		5/558	Accesssnar	(80)	2013-07	SELECT pg_stat	_get_backend_	pid(sviid) AS pid, (pid(outid) AS pid, (SELECT dates	me FROM pg
1642	db01	ng_cata	symfo		5/558	AccessShar	(21.5	2013-07	SELECT on stat	_get_backend_	pid(wild) AS pid, (nid(wild) AS nid, (SELECT dates	me FROM pg
1642		pg_data	symfo		5/558	AccessShar	(BL)	2013-07	SELECT pg_stat	get_backend	pid(svrid) AS pid, (SELECT datna	me FROM pg
						-			100000000000000000000000000000000000000				
- Sector	White and										A COLOR OF COLOR		-
DOBL NO.								3575235555	0.0000000000000			12/22/22/22/22	11111111

8.5 アクセス異常の対処

アクセスが拒否された場合、以下のディレクトリにインスタンス管理者が操作できる権限を与え、再度操作を実行してください。また、 ディスク異常などによりファイルシステムが読み取り専用でマウントされていないか、イベントログおよびサーバログを参照して確認して ください。読み取り専用でマウントされている場合は、正しくマウントし再度操作を実行してください。

- データ格納先
- テーブル空間格納先
- トランザクションログ格納先
- バックアップデータ格納先



8.6 データ格納先の容量不足時の対処

データ格納先の容量が不足した場合は、まずディスク上に不要なファイルがないかを確認し、不要なファイルを削除して業務を継続で きるようにしてください。

不要なファイルを削除しても問題を解消できない場合は、容量の大きなディスクへのデータの移行が必要になります。

データの移行には、以下の2つの方法があります。

- 8.6.1 テーブル空間を使用する方法
- ・ 8.6.2 容量の大きいディスクにディスク交換する方法

8.6.1 テーブル空間を使用する方法

Symfoware Serverでは、テーブル空間を使用して、テーブルやインデックスなどのデータベースオブジェクトの格納先を別のディスク に変更できます。

以下に、手順を示します。

1. テーブル空間の作成

CREATE TABLESPACEコマンドを使用して、新たに用意したディスクにテーブル空間を作成します。

2. テーブル空間の変更

ALTER TABLEコマンドを使用して新たに定義したテーブル空間にテーブルを変更します。



CREATE TABLESPACEコマンドおよびALTER TABLEコマンドの詳細については、"PostgreSQL文書"の"リファレンス"の"SQLコマンド"を参照してください。

8.6.2 容量の大きいディスクにディスク交換する方法

容量の大きいディスクへの交換を実施する場合は、バックアップおよびリカバリ機能を使用してデータ格納先の資源を移行する必要があります。

バックアップとリカバリ操作は、以下の2つの方法で行うことができます。

- 8.6.2.1 WebAdminを使用する場合
- ・ 8.6.2.2 サーバコマンドを使用する場合

以降で、それぞれの操作でディスク交換とデータ格納先の資源の移行を行う場合の手順について説明します。



- ・ ディスク交換を実施する前に、データベースを利用しているアプリケーションおよびインスタンスを停止してください。
- リカバリ操作後は、データベースクラスタのバックアップを推奨します。バックアップすることで、不要になったアーカイブログ(バック アップデータ格納先にコピーされたトランザクションログ)が削除されるため、ディスク容量の確保やリカバリ時間の短縮につながり ます。

8.6.2.1 WebAdminを使用する場合

WebAdminを使用してディスク交換とデータ格納先の資源の移行を行う場合の手順を以下に示します。

1. ファイルの退避

データ格納先のディスク配下に必要なファイルが存在する場合、ファイルを退避します。データ格納先の退避は不要です。

2. アプリケーションの停止

データベースを利用しているアプリケーションを停止してください。

3. データベースクラスタのバックアップ

最新のデータ格納先の資源をバックアップします。バックアップ方法については、"3.2.1 WebAdminを使用する場合"を参照してください。

4. インスタンスの停止

インスタンスを停止します。停止方法については、"2.1.1 WebAdminを使用する場合"を参照してください。

5. 容量の大きいディスクへの交換

ディスクを交換します。その後、ボリュームの構成情報をリカバリします。

6. データベースクラスタのリカバリ

WebAdminにログインし、リカバリ操作を行います。操作方法については、"8.1.1 WebAdminを使用する場合"の"データ格納ディスク、またはトランザクションログ格納ディスクに障害が発生した場合"の手順4"テーブル空間のディレクトリ作成"から手順7"リカバリの実行"を参照してください。リカバリが正常に完了すると、インスタンスが自動的に起動されます。

7. アプリケーションの再開

データベースを利用しているアプリケーションを再開してください。

8. ファイルの復元

手順1"ファイルの退避"で退避したファイルを元に戻します。

8.6.2.2 サーバコマンドを使用する場合

サーバコマンドを使用してディスク交換とデータ格納先の資源の移行を行う場合の手順を以下に示します。

1. ファイルの退避

データ格納先のディスク配下に必要なファイルが存在する場合、ファイルを退避します。データ格納先の退避は不要です。

2. アプリケーションの停止

データベースを利用しているアプリケーションを停止してください。

3. データベースクラスタのバックアップ

サーバコマンドを使用して最新のデータ格納先資源をバックアップします。バックアップ方法については、"3.2.2 サーバコマンド を使用する場合"を参照してください。

4. インスタンスの停止

バックアップ完了後、インスタンスを停止します。インスタンスの停止方法については、"2.1.2 コマンドを利用する場合"を参照してください。

インスタンスの停止に失敗した場合は、"8.11 インスタンス停止失敗時の対処"を参照してください。

5. 容量の大きいディスクへ交換

ディスクを交換します。その後、ボリュームの構成情報をリカバリします。

6. データ格納先のディレクトリ作成

データ格納先のディレクトリを作成します。テーブル空間を定義していた場合、そのディレクトリも作成してください。

データ格納先のディレクトリは、Windows(R)のエクスプローラの[プロパティ]でインスタンス管理者のみがアクセスできるように許可を設定します。([プロパティ]の詳細は、Windows(R)の[ヘルプとサポート]を参照してください。)

7. キーストアのリカバリとキーストアの自動オープンの有効化

データベースに格納するデータを暗号化している場合は、データベースのバックアップを取得した時点のキーストアをリストアしてください。キーストアの自動オープンは、必要に応じて設定してください。

8. データベースクラスタのリカバリ

pgx_rcvallコマンドを使用して、データベースクラスタを復旧します。

- -Dオプションは、データ格納先のディレクトリを指定します。-Dオプションを省略した場合、PGDATA環境変数の値が使用されます。
- -Bオプションは、バックアップデータ格納先のディレクトリを指定します。

例

> pgx_rcvall -D D:¥database¥inst1 -B E:¥backup¥inst1



リカバリに失敗した場合は、表示されるエラーメッセージに従ってエラーの原因を取り除き、pgx_rcvallコマンドを再実行してください。

「pgx_rcvall:リカバリ中にエラーが発生しました」のメッセージがある場合は、その後にリカバリ実行時のログが出力されます。ロ グの最後の十数行以内にエラーの原因が出力されますので、そのメッセージに従ってエラーの原因を取り除き、pgx_rcvallコマ ンドを再実行してください。

また、リカバリ中に表示される以下のメッセージは、pgx_rcvallコマンドの正常な動作によって出力されるメッセージのため、ユーザーが意識する必要はありません。

.....

FATAL: データベースシステムは起動しています



9. インスタンスの起動

インスタンスを起動してください。

.

インスタンスの起動方法については、"2.1.2 コマンドを利用する場合"を参照してください。



pgx_rcvallコマンドは、ハッシュインデックスを正しく復旧できません。 ハッシュインデックスを使用している場合、インスタンス起動 後、該当するインデックスに対して、REINDEXコマンドを実行してください。

10. アプリケーションの再開

データベースを利用しているアプリケーションを再開してください。

11. ファイルの復元

手順1"ファイルの退避"で退避したファイルを元に戻します。

8.7 バックアップデータ格納先の容量不足時の対処

バックアップデータ格納先の容量が不足した場合は、まずディスク上に不要なファイルがないかを確認し、不要なファイルを削除する ようにしてください。その後、必要であればバックアップを実施してください。

不要なファイルを削除しても問題を解消できない場合は、以下の対処を実施してください。

- 8.7.1 バックアップデータを一時退避する方法
- ・ 8.7.2 容量の大きいディスクにディスク交換する方法

8.7.1 バックアップデータを一時退避する方法

ー時的にバックアップデータを別のディレクトリに移動して退避し、バックアップデータ格納先のディスク容量を確保して、バックアップ を正常に取得できるようにする方法です。

容量の大きいディスクを準備するまでに時間がかかる場合は、この方法を採用してください。

バックアップデータ格納先の容量が不足すると、アーカイブログをバックアップデータ格納先に格納できなくなります。これに伴い、データ格納先またはトランザクションログ格納先にトランザクションログが溜まり続けます。

そのため対処に時間がかかると、トランザクションログの格納先が満杯になり、業務が継続できない状態に陥ることがあります。

このような状態に陥らないようにするため、バックアップデータ格納先の容量を確保して、アーカイブログを格納できるようにします。 対処には以下の2つの方法があります。

- 8.7.1.1 WebAdminを使用する場合
- ・8.7.1.2 サーバコマンドを使用する場合

8.7.1.1 WebAdminを使用する場合

以下の手順でバックアップデータ格納ディスクをリカバリしてください。

1. バックアップデータの一時退避

バックアップデータを別のディレクトリに移動し、一時退避するとともにバックアップデータ格納先ディレクトリの容量を確保します。

退避を行う理由は、万が一、復旧するまでの間にデータ格納先のデータが破損した場合でも復旧できるようにするためです。退 避先のディスクがなく、データ格納先の破損の危険性がないと判断した場合は、バックアップデータを削除してください。

下記に、バックアップデータ格納先のディレクトリ(E:¥backup¥inst1)にあるバックアップデータをF:¥mnt¥usb¥backup配下に退避 する場合の例を示します。

例



備考)一時退避先のディレクトリは、OS資源やSymfoware Serverの資源に影響しないよう配置してください。

2. バックアップデータのリカバリ

WebAdminにログインしてバックアップデータのリカバリを開始します。

[監視]メニュー画面のエラーメッセージの[対処]ボタンをクリックします。

4 Symfoware			ユーザー: symto ログアウト FUITSU
監視 設定			€ III 🗿 7= <u>27</u> 1/
インスタンス一覧	インスタンス名: inst1		
新規作成			
inst1			107-03
	インスタンス	データ格納先	バックアップデータ格納先
	稼働状態: 停止	状態: 正常	状態: 異栄
	 インスタンス名: inst1 ポート番号: 20500 エンコード: UTF8	パス: Di¥symdo¥inst1¥data	バス: E¥symfo¥inst1¥backup
	足動	デー外製旧 ロバックフ	アップ日時> 2013-10-17 16:21
	▼ メッセージ		
	 2013-10-17 1622:04 バックアップ 	データ格納先のディスク容量が残りわずかで	す。(E:)
	 2013-10-17 1623:07 バックアップラ 	データ格納先の資源に異常を検出しました。	
	🧘 2013-10-17 16:23:59 インスタンスト	こ問題が発生しました。対処ボタンを押してく	ださい。 創始 3月 3日 100 100 100 100 100 100 100 100 100 10

上記画面は、バックアップデータを移動して退避したことにより、WebAdminがバックアップデータ格納先の内容が消えたことを 検知した場合の画面です。別の方法で退避を行い異常を検知していない場合は、[監視]メニュー画面の「バックアップ」と記され た[→]ボタンをクリックしてください。

3. バックアップの実行

バックアップデータをリカバリするために、バックアップを実行します。[バックアップ]ダイアログが表示されますので、[実行]ボタ ンをクリックしてください。[監視]メニュー画面に「バックアップ中」と表示され、バックアップが実行されます。 バックアップを実行す ると、インスタンスが自動的に起動されます。



4. 一時退避したバックアップデータの削除

バックアップが正常に完了した場合、一時退避したバックアップデータは不要になるため削除します。

下記に、F:¥mnt¥usbに一時退避したバックアップデータを削除する場合の例を示します。

```
例
```

> rmdir /S /Q F:¥mnt¥usb¥backup

8.7.1.2 サーバコマンドを使用する場合

バックアップデータ格納ディスクをリカバリする手順を説明します。

対処方法には以下の2つの方法があります。

- ・ インスタンスを起動したままリカバリする方法
- ・ インスタンスを停止してリカバリする方法

インスタンス停止の有無による、実行する手順の違いを以下に示します。

No	手順	インスタンス停止の有無	
		なし	あり
1	トランザクションログ二重化の停止	0	_
2	アーカイブログ出力の停止	0	_
3	アプリケーションの停止	—	0
4	インスタンスの停止	—	0
5	バックアップデータの一時退避	0	0
6	アーカイブログ出力の再開	0	—
7	トランザクションログ二重化の再開	0	—

No	手順	インスタンス停止の有無	
		なし	あり
8	インスタンスの起動	_	0
9	バックアップの実行	0	0
10	アプリケーションの再開	—	0
11	一時退避したバックアップデータの削除	0	0

○:実施

-:実施不要

手順を以下に示します。

インスタンスを起動したままリカバリする方法

1. トランザクションログ二重化の停止

トランザクションログの二重化を停止します。

postgres=# SELECT pgx_pause_wal_multiplexing(): LOG: multiplexing of transaction log files has been stopped pgx_pause_wal_multiplexing

(1 行)

2. アーカイブログ出力の停止

バックアップ格納ディスクを交換するまでに時間がかかると、トランザクションログが蓄積されていきます。そして、データ格納ディ スクまたはトランザクションログ格納ディスクが満杯になると、運用が継続できなくなる危険性があります。

これを回避するために、以下の方法でアーカイブログの出力を停止させてください。

- archive_commandパラメータの変更

アーカイブログを出力したものとみなすよう、必ず正常終了するコマンドを指定します。

echoを指定すると、サーバログにメッセージが出力されるため、調査を行う上での指標になる可能性があります。

- 設定ファイルの再読込み

pg_ctl reloadコマンド、またはSQL関数pg_reload_confを実行します。

なお、運用が継続できなくなる危険性が無く、単にエラーを出力させたくない場合は、archive_commandに空文字列(")を指定して設定ファイルの再読み込みを実施してください。

3. バックアップデータの一時退避

バックアップデータを別のディレクトリに移動し、退避するとともにバックアップデータ格納先ディレクトリの容量を確保します。

退避を行う理由は、万が一、次の手順を行うまでの間にデータ格納先のデータが破損した場合でも復旧できるようにするためで す。退避先のディスクがなく、データ格納先の破損の危険性がないと判断した場合は、バックアップデータを削除してください。

下記に、バックアップデータ格納先のディレクトリ(E:¥backup¥inst1)にあるバックアップデータをF:¥mnt¥usb¥backup配下に退避 する場合の例を示します。

例

```
> mkdir F:¥mnt¥usb¥backup
> move E:¥backup¥inst1¥* F:¥mnt¥usb¥backup
```

備考)一時退避先のディレクトリは、OS資源やSymfoware Serverの資源に影響しないよう配置してください。

4. アーカイブログ出力の再開

archive_commandの設定を元に戻し、設定ファイルを再読込みします。

5. トランザクションログ二重化の再開

SQL関数pgx_resume_wal_multiplexingを実行します。

例

SELECT pgx_resume_wal_multiplexing()

6. バックアップの実行

```
pgx_dmpallコマンドを使用して、データベースクラスタをバックアップします。
```

pgx_dmpallコマンドには、以下を指定します。

- -Dオプションは、データ格納先のディレクトリを指定します。-Dオプションを省略した場合、PGDATA環境変数の値が使用されます。

例

> pgx_dmpall -D D:\u00e4database\u00e4inst1

7. 一時退避したバックアップデータの削除

バックアップが正常に完了した場合、一時退避したバックアップデータは不要になるため削除します。

下記に、F:¥mnt¥usbに一時退避したバックアップデータを削除する場合の例を示します。

例

> rmdir /S /Q F:¥mnt¥usb¥backup

インスタンスを停止してリカバリする方法

1. アプリケーションの停止

データベースを利用しているアプリケーションを停止してください。

2. インスタンスの停止

インスタンスを停止します。停止方法については、"2.1.2 コマンドを利用する場合"を参照してください。 インスタンスの停止に失敗した場合は、"8.11 インスタンス停止失敗時の対処"を参照してください。

3. バックアップデータの一時退避

バックアップデータを別のディレクトリに移動し、退避するとともにバックアップデータ格納先ディレクトリの容量を確保します。 退避を行う理由は、万が一、次の手順を行うまでの間にデータ格納先のデータが破損した場合でも復旧できるようにするためで す。退避先のディスクがなく、データ格納先の破損の危険性がないと判断した場合は、バックアップデータを削除してください。

下記に、バックアップデータ格納先のディレクトリ(E:¥backup¥inst1)にあるバックアップデータをF:¥mnt¥usb¥backup配下に退避 する場合の例を示します。
例

> mkdir F:¥mnt¥usb¥backup > move E:¥backup¥inst1¥* F:¥mnt¥usb¥backup

備考)一時退避先のディレクトリは、OS資源やSymfoware Serverの資源に影響しないよう配置してください。

4. インスタンスの起動

インスタンスを起動します。起動方法については、"2.1.2 コマンドを利用する場合"を参照してください。

5. バックアップの実行

pgx_dmpallコマンドを使用して、データベースクラスタをバックアップします。

pgx_dmpallコマンドには、以下を指定します。

- -Dオプションは、データ格納先のディレクトリを指定します。-Dオプションを省略した場合、PGDATA環境変数の値が使用されます。

例

> pgx_dmpall -D D:¥database¥inst1

6. アプリケーションの再開

データベースを利用しているアプリケーションを再開してください。

7. 一時退避したバックアップデータの削除

バックアップが正常に完了した場合、一時退避したバックアップデータは不要になるため削除します。

下記に、F:¥mnt¥usbに一時退避したバックアップデータを削除する場合の例を示します。

例

> rmdir /S /Q F:\u00e4mnt\u00e4usb\u00e4backup



・ pgx_rcvallコマンド、およびpgx_dmpallコマンドの詳細は、"リファレンス"の"pgx_rcvall"および"pgx_dmpall"を参照してください。

- ・ archive_commandの詳細は、"PostgreSQL文書"の"サーバの管理"の"ログ先行書き込み(WAL)"を参照してください。
- pgx_is_wal_multiplexing_paused、およびpgx_resume_wal_multiplexingの詳細は、"B.1 WAL二重化制御関数"を参照してください。

8.7.2 容量の大きいディスクにディスク交換する方法

再び容量が不足することのないように、バックアップデータ格納先のディスクを容量の大きいディスクに交換する方法です。 ディスクを交換した後にバックアップを行い、正しいバックアップを取得します。

バックアップ操作は、以下の2つの方法で行うことができます。

• 8.7.2.1 WebAdminを使用する場合

・8.7.2.2 サーバコマンドを使用する場合



ディスク交換を実施する前に、データベースを利用しているアプリケーションを停止してください。

8.7.2.1 WebAdminを使用する場合

以下の手順でバックアップデータ格納ディスクをリカバリしてください。

1. ファイルの退避

バックアップデータ格納先のディスク配下に必要なファイルが存在する場合、ファイルを退避します。

2. バックアップデータの一時退避

バックアップデータを別のディレクトリに退避します。

退避を行う理由は、万が一、次の手順を行うまでの間にデータ格納先のデータが破損した場合でも復旧できるようにするためで す。退避先のディスクがなく、データ格納先の破損の危険性がないと判断した場合は、バックアップデータを削除してください。

下記に、バックアップデータ格納先のディレクトリ(E:¥backup¥inst1)にあるバックアップデータをF:¥mnt¥usb¥backup配下に退避 する場合の例を示します。

例

> mkdir F:¥mnt¥usb¥backup > move E:¥backup¥inst1¥* F:¥mnt¥usb¥backup

備考)一時退避先のディレクトリは、OS資源やSymfoware Serverの資源に影響しないよう配置してください。

3. 容量の大きいディスクへの交換

ディスクを交換します。その後、ボリュームの構成情報をリカバリします。

4. バックアップの実行

WebAdminにログインし、リカバリ操作を行います。"8.1.1 WebAdminを使用する場合"の"バックアップデータ格納ディスクに障害が発生した場合"の手順2"バックアップデータのリカバリ"および手順3"バックアップの実行"を参照してください。

5. ファイルの復元

手順1"ファイルの退避"で退避したファイルを元に戻します。

6. 一時退避したバックアップデータの削除

バックアップが正常に完了した場合、一時退避したバックアップデータは不要になるため削除します。

下記に、F:¥mnt¥usbに一時退避したバックアップデータを削除する場合の例を示します。

例

> rmdir /S /Q F:\u00e4mnt\u00e4usb\u00e4backup

8.7.2.2 サーバコマンドを使用する場合

バックアップデータ格納ディスクをリカバリする手順を説明します。 対処方法には以下の2つの方法があります。

- ・ インスタンスを起動したままリカバリする方法
- ・ インスタンスを停止してリカバリする方法

インスタンス停止の有無による、実行する手順の違いを以下に示します。

No	手順	インスタンス停止の有無						
		なし	あり					
1	ファイルの退避	0	0					
2	バックアップデータの一時退避	0	0					
3	トランザクションログ二重化の停止確認	0	_					
4	アーカイブログ出力の停止	0	—					
5	アプリケーションの停止	_	0					
6	インスタンスの停止	_	0					
7	容量の大きなディスクに交換	0	0					
8	バックアップデータ格納先のディレクトリ作成	0	0					
9	アーカイブログ出力の再開	0	—					
10	トランザクションログ二重化の再開	0	—					
11	インスタンスの起動	_	0					
12	バックアップの実行	0	0					
13	アプリケーションの再開	—	0					
14	ファイルの復元	0	0					
15	一時退避したバックアップデータの削除	0	0					

○:実施

-:実施不要

手順を以下に示します。

インスタンスを起動したままリカバリする方法

1. ファイルの退避

バックアップデータ格納先のディスク配下に必要なファイルが存在する場合、ファイルを退避します。 バックアップデータ格納 先の退避は不要です。

2. バックアップデータの一時退避

バックアップデータを別のディレクトリに退避します。

退避を行う理由は、万が一、次の手順を行うまでの間にデータ格納先のデータが破損した場合でも復旧できるようにするためです。 退避先のディスクがなく、データ格納先の破損の危険性がないと判断した場合は、バックアップデータを削除してく ださい。

下記に、バックアップデータ格納先のディレクトリ(E:¥backup¥inst1)にあるバックアップデータをF:¥mnt¥usb¥backup配下に退避する場合の例を示します。

例

- > mkdir F:¥mnt¥usb¥backup
 > move E:¥backup¥inst1¥* F:¥mnt¥usb¥backup
- 3. トランザクションログ二重化の停止確認

トランザクションログの二重化が停止されているかを、次のSQL関数で確認します。

postgres=# SELECT pgx_is_wal_multiplexing_paused(): pgx_is_wal_multiplexing_paused t (1 row)

トランザクションログの二重化が停止していない場合は、次のSQL関数で停止してください。

postgres=# SELECT pgx_pause_wal_multiplexing(); LOG: multiplexing of transaction log files has been stopped pgx_pause_wal_multiplexing

(1 行)

4. アーカイブログ出力の停止

バックアップ格納ディスクを交換するまでに時間がかかると、トランザクションログが蓄積されていきます。そして、データ格納 ディスクまたはトランザクションログ格納ディスクが満杯になると、運用が継続できなくなる危険性があります。

これを回避するために、以下の方法でアーカイブログの出力を停止してください。

- archive_commandパラメータの変更

アーカイブログを出力したものとみなすよう、必ず正常終了するコマンドを指定します。

echoを指定すると、サーバログにメッセージが出力されるため、調査を行う上での指標になる可能性があります。

- 設定ファイルの再読込み

pg_ctl reloadコマンド、またはSQL関数pg_reload_confを実行します。

なお、運用が継続できなくなる危険性が無く、単にエラーを出力させたくない場合は、archive_commandに空文字列(")を指定して設定ファイルの再読み込みを実施してください。

5. 容量の大きいディスクへの交換

ディスクを交換します。その後、ボリュームの構成情報をリカバリします。

6. バックアップデータ格納先のディレクトリ作成

バックアップデータ格納先のディレクトリを作成します。

バックアップデータ格納先のディレクトリは、Windows(R)のエクスプローラの[プロパティ]でインスタンス管理者のみがアクセ スできるように許可を設定します。([プロパティ]の詳細は、Windows(R)の[ヘルプとサポート]を参照してください。) 詳細については、"3.2.2 サーバコマンドを使用する場合"を参照してください。

7. アーカイブログ出力の再開

archive_commandの設定を元に戻し、設定ファイルを再読込みします。

8. トランザクションログ二重化の再開

```
SQL関数pgx_resume_wal_multiplexingを実行します。
```

例

SELECT pgx_resume_wal_multiplexing()

9. バックアップの実行

pgx_dmpallコマンドを使用して、データベースクラスタをバックアップします。

pgx_dmpallコマンドには、以下を指定します。

- -Dオプションは、データ格納先のディレクトリを指定します。-Dオプションを省略した場合、PGDATA環境変数の値が使用されます。

例

> pgx_dmpall -D D:¥database¥inst1

10. ファイルの復元

手順1"ファイルの退避"で退避したファイルを元に戻します。

11. 一時退避したバックアップデータの削除

バックアップが正常に完了した場合、一時退避したバックアップデータは不要になるため削除します。

下記に、F:¥mnt¥usbに一時退避したバックアップデータを削除する場合の例を示します。

例

> rmdir /S /Q F:\u00e4mnt\u00e4usb\u00e4backup

インスタンスを停止してリカバリする方法

1. ファイルの退避

バックアップデータ格納先のディスク配下に必要なファイルが存在する場合、ファイルを退避します。 バックアップデータ格納 先の退避は不要です。

2. バックアップデータの一時退避

バックアップデータを別のディレクトリに退避します。

退避を行う理由は、万が一、次の手順を行うまでの間にデータ格納先のデータが破損した場合でも復旧できるようにするためです。 退避先のディスクがなく、データ格納先の破損の危険性がないと判断した場合は、バックアップデータを削除してく ださい。

下記に、バックアップデータ格納先のディレクトリ(E:¥backup¥inst1)にあるバックアップデータをF:¥mnt¥usb¥backup配下に退 避する場合の例を示します。

例

> mkdir F:¥mnt¥usb¥backup > move E:¥backup¥inst1¥* F:¥mnt¥usb¥backup

備考) 一時退避先のディレクトリは、OS資源やSymfoware Serverの資源に影響しないよう配置してください。

3. アプリケーションの停止

データベースを利用しているアプリケーションを停止してください。

4. インスタンスの停止

インスタンスを停止します。インスタンスの停止方法については、"2.1.2 コマンドを利用する場合"を参照してください。 インスタンスの停止に失敗した場合は、"8.11 インスタンス停止失敗時の対処"を参照してください。

5. 容量の大きいディスクへの交換

ディスクを交換します。その後、ボリュームの構成情報をリカバリします。

6. バックアップデータ格納先のディレクトリ作成

バックアップデータ格納先のディレクトリを作成します。

バックアップデータ格納先のディレクトリは、Windows(R)のエクスプローラの[プロパティ]でインスタンス管理者のみがアクセ スできるように許可を設定します。([プロパティ]の詳細は、Windows(R)の[ヘルプとサポート]を参照してください。) 詳細については、"3.2.2 サーバコマンドを使用する場合"を参照してください。

7. インスタンスの起動

インスタンスを起動します。起動方法については、"2.1.2 コマンドを利用する場合"を参照してください。

8. バックアップの実行

pgx_dmpallコマンドを使用して、データベースクラスタをバックアップします。

pgx_dmpallコマンドには、以下を指定します。

- -Dオプションは、データ格納先のディレクトリを指定します。-Dオプションを省略した場合、PGDATA環境変数の値が使用されます。

例

> pgx_dmpall -D D:¥database¥inst1

9. アプリケーションの再開

データベースを利用しているアプリケーションを再開してください。

10. ファイルの復元

手順1"ファイルの退避"で退避したファイルを元に戻します。

11. 一時退避したバックアップデータの削除

バックアップが正常に完了した場合、一時退避したバックアップデータは不要になるため削除します。

下記に、F:¥mnt¥usbに一時退避したバックアップデータを削除する場合の例を示します。

例

> rmdir /S /Q F:\u00e4mnt\u00e4usb\u00e4backup



- ・ pgx_rcvallコマンド、およびpgx_dmpallコマンドの詳細は、"リファレンス"の"pgx_rcvall"および"pgx_dmpall"を参照してください。
- archive_modeの詳細は、"PostgreSQL文書"の"サーバの管理"の"ログ先行書き込み(WAL)"を参照してください。
- pgx_is_wal_multiplexing_paused、およびpgx_resume_wal_multiplexingの詳細は、"B.1 WAL二重化制御関数"を参照してください。

8.8 トランザクションログ格納先の容量不足時の対処

トランザクションログ格納先の容量が不足した場合は、まずディスク上に不要なファイルがないかを確認し、不要なファイルを削除して 業務を継続できるようにしてください。

不要なファイルを削除しても問題を解消できない場合は、容量の大きなディスクへのデータの移行が必要になります。

8.8.1 容量の大きいディスクにディスク交換する方法

容量の大きいディスクへの交換を実施する場合は、バックアップおよびリカバリ機能を使用してトランザクションログ格納先の資源を移 行する必要があります。

バックアップとリカバリ操作は、以下の2つの方法で行うことができます。

- 8.8.1.1 WebAdminを使用する場合
- ・ 8.8.1.2 サーバコマンドを使用する場合

以降で、それぞれの操作でディスク交換とトランザクションログ格納先の資源の移行を行う場合の手順について説明します。



- ・ ディスク交換を実施する前に、データベースを利用しているアプリケーションおよびインスタンスを停止してください。
- リカバリ操作後は、データベースクラスタのバックアップを推奨します。バックアップすることで、不要になったアーカイブログ(バック アップデータ格納先にコピーされたトランザクションログ)が削除されるため、ディスク容量の確保やリカバリ時間の短縮につながり ます。

8.8.1.1 WebAdminを使用する場合

WebAdminを使用してディスク交換とトランザクションログ格納先の資源の移行を行う場合の手順を以下に示します。

1. ファイルの退避

トランザクションログ格納先のディスク配下に必要なファイルが存在する場合、ファイルを退避します。トランザクションログ格納先の退避は不要です。

2. データベースクラスタのバックアップ

最新のデータ格納先の資源、およびトランザクションログ格納先の資源をバックアップします。バックアップ方法については、"3.2.1 WebAdminを使用する場合"を参照してください。

3. アプリケーションの停止

データベースを利用しているアプリケーションを停止してください。

4. インスタンスの停止

インスタンスを停止します。停止方法については、"2.1.1 WebAdminを使用する場合"を参照してください。インスタンスを停止せずにデータベースクラスタのリカバリをした場合は、WebAdminが自動的にインスタンスを停止します。

5. 容量の大きいディスクへの交換

ディスクを交換します。その後、ボリュームの構成情報をリカバリします。

- テーブル空間のディレクトリ作成 バックアップを実施した以降にテーブル空間を定義している場合、そのディレクトリを作成します。
- 7. キーストアのリカバリとキーストアの自動オープンの有効化

データベースに格納するデータを暗号化している場合は、以下を行ってください。

- データベースのバックアップを取得した時点のキーストアをリストアしてください。
- キーストアの自動オープンを有効にしてください。
- 8. データベースクラスタのリカバリ

WebAdminにログインし、リカバリ操作を行います。操作方法については、"8.1.1 WebAdminを使用する場合"の"データ格納ディスク、またはトランザクションログ格納ディスクに障害が発生した場合"の手順4"テーブル空間のディレクトリ作成"から手順7"リカバリの実行"を参照してください。リカバリが正常に完了すると、インスタンスが自動的に起動されます。

9. アプリケーションの再開

データベースを利用しているアプリケーションを再開してください。

10. ファイルの復元

手順1"ファイルの退避"で退避したファイルを元に戻します。

8.8.1.2 サーバコマンドを使用する場合

サーバコマンドを使用してディスク交換とトランザクションログ格納先の資源の移行を行う場合の手順を以下に示します。

1. ファイルの退避

トランザクションログ格納先のディスク配下に必要なファイルが存在する場合、ファイルを退避します。トランザクションログ格納先の退避は不要です。

2. データベースクラスタのバックアップ

サーバコマンドを使用して最新のデータ格納先資源、およびトランザクションログ格納先資源をバックアップします。バックアップ 方法については、"3.2.2 サーバコマンドを使用する場合"を参照してください。 3. アプリケーションの停止

データベースを利用しているアプリケーションを停止してください。

4. インスタンスの停止

バックアップ完了後、インスタンスを停止します。インスタンスの停止方法については、"2.1.2 コマンドを利用する場合"を参照してください。

インスタンスの停止に失敗した場合は、"8.11 インスタンス停止失敗時の対処"を参照してください。

5. 容量の大きいディスクへ交換

ディスクを交換します。その後、ボリュームの構成情報をリカバリします。

6. トランザクションログ格納先のディレクトリ作成

トランザクションログ格納先のディレクトリを作成します。テーブル空間を定義していた場合、そのディレクトリも作成してください。

トランザクションログ格納先のディレクトリは、Windows(R)のエクスプローラの[プロパティ]でインスタンス管理者のみがアクセスできるように許可を設定します。([プロパティ]の詳細は、Windows(R)の[ヘルプとサポート]を参照してください。)

7. キーストアのリカバリとキーストアの自動オープンの有効化

データベースに格納するデータを暗号化している場合は、データベースのバックアップを取得した時点のキーストアをリストアしてください。キーストアの自動オープンは、必要に応じて設定してください。

8. データベースクラスタのリカバリ

pgx_rcvallコマンドを使用して、データベースクラスタを復旧します。

- -Dオプションは、データ格納先のディレクトリを指定します。-Dオプションを省略した場合、PGDATA環境変数の値が使用されます。
- -Bオプションは、バックアップデータ格納先のディレクトリを指定します。

例

> pgx_rcvall -D D:¥database¥inst1 -B E:¥backup¥inst1



リカバリに失敗した場合は、表示されるエラーメッセージに従ってエラーの原因を取り除き、pgx_rcvallコマンドを再実行してください。

「pgx_rcvall:リカバリ中にエラーが発生しました」のメッセージがある場合は、その後にリカバリ実行時のログが出力されます。ロ グの最後の十数行以内にエラーの原因が出力されますので、そのメッセージに従ってエラーの原因を取り除き、pgx_rcvallコマ ンドを再実行してください。

また、リカバリ中に表示される以下のメッセージは、pgx_rcvallコマンドの正常な動作によって出力されるメッセージのため、ユーザーが意識する必要はありません。

FATAL: データベースシステムは起動しています



pgx_rcvallコマンドの詳細は、"リファレンス"の"pgx_rcvall"を参照してください。

9. インスタンスの起動

インスタンスを起動してください。

インスタンスの起動方法については、"2.1.2 コマンドを利用する場合"を参照してください。



pgx_rcvallコマンドは、ハッシュインデックスを正しく復旧できません。 ハッシュインデックスを使用している場合、インスタンス起動 後、該当するインデックスに対して、REINDEXコマンドを実行してください。

10. アプリケーションの再開

データベースを利用しているアプリケーションを再開してください。

11. ファイルの復元

手順1"ファイルの退避"で退避したファイルを元に戻します。

8.9 各格納先ディスクの異常

各格納先ディスクで異常および資源破壊が発生した場合は、イベントログおよびサーバログから異常原因を特定して原因を取り除いてください。

ディスクの異常が以下の組み合わせで発生した場合は、リカバリすることはできません。

インスタンスを再作成して運用環境を再構築してください。

データ 格納ディスク	トランザクションログ 格納ディスク	バックアップデータ 格納ディスク				
異常	_	異常				
_	異常	異常				



インスタンスの作成と運用環境の構築については、"導入ガイド(サーバ編)"の"セットアップ"を参照してください。

8.10 インスタンス起動失敗時の対処

インスタンスの起動に失敗した場合は、イベントログおよびサーバログを参照して原因を特定してください。

WebAdminを利用する場合は、原因の対処を実施した後に[対処]ボタンをクリックして[状態を再確認する]をクリックし、インスタンスが 正常な状態にあることを確認してください。

以降によくある原因と対処を説明します。

8.10.1 設定ファイルの誤り

設定ファイルをテキストエディタで直接編集した場合や、WebAdminで設定内容の変更を行った場合は、イベントログおよびサーバログを参照して以下のファイルに関するメッセージが出力されていないかを確認してください。

- postgresql.conf
- pg_hba.conf

💦 参照

設定ファイルのパラメータの詳細については、以下を参照してください。

- ・ "導入ガイド(サーバ編)"の"設定パラメータ"
- "付録A パラメータ"
- ・ "PostgreSQL 文書"の"サーバの管理"の"サーバの設定"および"クライアント認証"

8.10.2 電源未投入やマウントによる異常

各格納先ディスクのディスク装置の電源を投入し忘れたり、自動マウントの設定を忘れていたために、サーバの再起動などにより、マウントが解除されている場合、インスタンスの起動が失敗します。

"8.13.2 電源未投入やマウントによる異常"を参照して対処してください。

8.10.3 PL/extJavaの起動失敗による異常

PL/extJavaの起動失敗によるインスタンス起動が異常となった場合は、以下のログに出力されたエラーメッセージを確認し、"メッセージ集"の"PL/extJavaが出力するメッセージ"を参照して起動失敗の原因を調査してください。これらのログの場所などの詳細は、"付録 GPL/extJavaのログ情報"を参照してください。

- ・ イベントログ
- Symfoware Serverのサーバログ
- コンテナのサーバログ
- ・ コンテナのJavaVMログ
- ドメインのサーバログ
- ・ ドメインのJavaVMログ

以下に代表的な原因とその対処を示します。

・ドメインルートが配置されているディスクの故障や、誤操作によって破壊されている場合は、以下のように復旧します。

PL/extJavaのバックアップを取得している場合

"7.3.7.2 リストア方法"を参照し、PL/extJavaをリストアしてください。

PL/extJavaのバックアップを取得していない場合

以下の手順を実施してください。

1. アプリケーションの停止

データベースを利用しているアプリケーションを停止してください。

2. インスタンスの停止

インスタンスを停止します。停止方法については、"2.1 インスタンスの起動と停止"を参照してください。インスタンスの停止 に失敗した場合は、"8.11 インスタンス停止失敗時の対処"を参照してください。 3. 障害ディスクのリカバリ

ディスクが故障している場合は、ディスク交換を行ったあと、ボリュームの構成情報をリカバリします。

4. ドメインルートのディレクトリの削除

ドメインルートのディレクトリを削除します。

5. postgresql.confファイルの設定変更

postgresql.confファイル内の以下のパラメータを削除します。

- plextjava.http_port
- plextjava.start_command
- plextjava.stop_command
- plextjava.forcible_stop_command
- 6. ドメインルート情報の削除

サービスに「FUJITSU PCMI(isje6-<コンポーネントID>)」が残存している場合は、scコマンドを以下のように実行して、サービス「FUJITSU PCMI(isje6-<コンポーネントID>)」を削除します。

sc delete "FUJITSU PCMI(isje6-<コンポーネントID>)"

以下のファイルが存在する場合は、ファイルを削除します。

[Symfoware Serverのインストールディレクトリ]¥java¥etc¥domain.conf [Symfoware Serverのインストールディレクトリ]¥java¥etc¥domain1_container.conf

🂕 例

<コンポーネントID>は、以下の形式で設定します。

"STFSYMOIDB"+"VL"+"アーキテクチャの先頭1桁"

- VL:Symfoware Serverのバージョン
- アーキテクチャの先頭1桁:製品のアーキテクチャ(32ビット版:3、64ビット版:6)

Symfoware Serverのバージョンが、"1210",製品のアーキテクチャが64ビット版の場合、コンポーネントIDは、以下のように設定されます。

STFSYMOIDB12106

7. PL/extJavaの再作成

PL/extJavaをpgx_jadminコマンドによる操作、および、Javaアプリケーションの格納のみ実施して再作成します。PL/extJavaの作成は、"第7章 PL/extJavaの導入と運用"を参照してください。

8. インスタンスの起動

インスタンスを起動します。起動方法については、"2.1 インスタンスの起動と停止"を参照してください。インスタンスの起動 に失敗した場合は、"8.10 インスタンス起動失敗時の対処"を参照してください。

9. アプリケーションの再開

データベースを利用しているアプリケーションを再開してください。

 コンテナのサーバログ、またはドメインのサーバログに、アドレスを割り当てられないことを示すメッセージが出力されている場合は、 指定されたポート番号を他のサービスが使用していることが考えられます。"7.2.1 ポート番号の準備"、および"7.3.3.3 ポート番号の変更"を参照して、ポート番号を変更してから、インスタンスを再起動してください。 • Symfoware Serverのサーバログには、Java EE DASサービスの起動失敗を示すメッセージが出力されているが、ドメインのサーバ ログには原因を示すメッセージが出力されていない場合には、キャッシュ情報が不整合を起こしている可能性があるため、以下の ファイルを削除して、インスタンスを再起動してください。

[ドメインルート]¥domains¥domain1¥osgi-cache¥felix

・ PL/extJavaのプロセスがすでに起動済みであることを示すメッセージが出力されていた場合には、過去にこれらのプロセスを監視 するPCMIサービスが異常終了したことで、プロセスが残存した可能性があります。以下の手順を実施して不要なプロセスを強制 終了してからインスタンスを再起動してください。

1. サーバインスタンス(JavaVM)のプロセス回収

以下のpidファイル内に記載されているプロセスが存在する場合には、OSの機能を用いてプロセスを強制終了し、pidファイルを削除してください。これはすべてのサーバインスタンス(JavaVM)に対して実施してください。

[ドメインルート]¥nodes¥localhost-domain1¥<サーバインスタンス(JavaVM)名¥config¥pid

2. Java EE DASサービスのプロセス回収

以下のpidファイル内に記載されているプロセスが存在する場合には、OSの機能を用いてプロセスを強制終了し、pidファイルを削除してください。

[ドメインルート]¥domains¥domain1¥config¥pid

3. Webサーバのプロセス回収

以下のpidファイル内に記載されているプロセスが存在する場合には、OSの機能を用いてプロセスを強制終了し、pidファイルを削除してください。

[Symfoware Serverインストールディレクトリ]¥java¥ahs22¥plextjava¥domain1¥logs¥httpd.pid

また、PCMIサービスのプロセスが残存している場合は、以下の手順を実施して不要なプロセスを強制終了してからインスタンスを再起動してください。

4. PCMIサービスのプロセス回収

以下の手順でプロセスが残存しているかを確認してください。プロセスが存在する場合には、OSの機能を用いてプロセスを 強制終了してください。

- Windowsのタスクマネージャ

8.10.4 その他の異常

イベントログおよびサーバログを参照しても対処できない場合やインスタンスが起動できない場合のリカバリ手順を説明します。

以下の2つの方法でリカバリを行うことができます。

- 8.10.4.1 WebAdminを使用する場合
- 8.10.4.2 サーバコマンドを使用する場合

ただし、バックアップデータ格納先に異常がある場合はリカバリできません。解決できない場合は当社技術員(サポート)に連絡してください。

8.10.4.1 WebAdminを使用する場合

以下の手順でリカバリしてください。

1. データ格納先ディレクトリおよびトランザクションログ格納先ディレクトリの削除

データ格納先ディレクトリおよびトランザクションログ格納先ディレクトリを退避してから削除します。

2. 状態の再確認

WebAdminにログインし、[監視]メニュー画面のエラーメッセージの[対処]ボタンをクリックします。 [状態を再確認する]をクリックし、各格納先の資源を再度検証します。

3. リカバリの実行

WebAdminが異常を検出したあと、データベースクラスタを復旧します。 詳細は、"8.2.1 WebAdminを使用する場合"を参照してください。

8.10.4.2 サーバコマンドを使用する場合

以下の手順でリカバリしてください。

- データ格納先ディレクトリおよびトランザクションログ格納先ディレクトリの削除
 データ格納先ディレクトリおよびトランザクションログ格納先ディレクトリを退避してから削除します。
- 2. リカバリの実行

pgx_rcvallコマンドを使用して、データベースクラスタを復旧します。 詳細は、"8.2.2 pgx_rcvallコマンドを使用する場合"を参照してください。

8.11 インスタンス停止失敗時の対処

インスタンスの停止に失敗した場合は、イベントログおよびサーバログを参照して原因を特定してください。 対処を実施してもインスタンスの停止ができない場合は以下の操作によりインスタンスを停止してください。 以下の2つの方法があります。

- 8.11.1 WebAdminを使用する場合
- ・ 8.11.2 サーバコマンドを使用する場合

8.11.1 WebAdminを使用する場合

[監視]メニュー画面から[停止]ボタンをクリックし、停止モードの"Fastモード"または"Immediateモード"を選択してインスタンスを停止します。インスタンスが停止できない場合は、WebAdminがサーバプロセスを強制停止します。

停止モードの詳細については、"2.1.1 WebAdminを使用する場合"を参照してください。

8.11.2 サーバコマンドを使用する場合

以下の3つの方法があります。

• Fastモードによる停止

バックアップが実行中であった場合はバックアップを終了させ、すべての実行中のトランザクションをロールバックして、クライアントとの接続を強制的に切断した後、インスタンスを停止します。

- ・ Immediateモードによる停止 インスタンスを即座に強制的に終了させます。インスタンスの再起動時にはクラッシュリカバリが実行されます。
- サーバプロセスの強制停止
 他の方法を実施しても停止できない場合に、サーバプロセスを確実に停止させます。

8.11.2.1 Fastモードによる停止

pg_ctlコマンドに"-m fast"を指定してインスタンスを停止します。

この方法を実施しても停止が失敗する場合は、"8.11.2.2 Immediateモードによる停止"、または"8.11.2.3 サーバプロセスの強制停止" に従った停止を実施してください。

》例 > pg_ctl stop -D D:¥database¥inst1 -m fast

8.11.2.2 Immediateモードによる停止

pg_ctlコマンドに"-m immediate"を指定してインスタンスを停止します。

この方法を実施しても停止が失敗する場合は、"8.11.2.3 サーバプロセスの強制停止"に従って停止を行ってください。

🄰 例

> pg_ctl stop -D D:¥database¥inst1 -m immediate

8.11.2.3 サーバプロセスの強制停止

Fastモード、Immediateモードのいずれの方法でも停止できない場合は、pg_ctlコマンドのkillパラメータを使用してサーバプロセスを強制停止します。

手順を以下に示します。

1. wmicコマンドを実行して、サーバプロセスのプロセスIDを特定します。

c:¥>wmic wmic:root¥cli>process where "name = ¥"postgres exe¥"" get Commandline Name ProcessId		
CommandLine	Name	ProcessId
~ ″C:¥Program Files¥Fujitsu¥symfoserver64¥bin¥postgres.exe″ -D ″D:¥database¥inst1″	postgres.exe	896
~		

-Dオプションの引数に、該当するインスタンスのデータ格納先のディレクトリが示されているpostgres.exeのプロセスID(896)が、サーバプロセスになります。

2. サーバプロセスを強制停止します。

インスタンス管理者で、pg_ctlコマンドによりサーバプロセスの強制停止を行います。

c:¥>pg_ctl kill QUIT 896

8.11.3 PL/extJavaの停止失敗による異常

PL/extJavaの停止で異常が発生しインスタンスの停止に失敗した場合、以下の対処を行ってください。

・ 以下のログにエラーメッセージが出力されていないかを確認します。

出力されたエラーメッセージは、"メッセージ集"の"PL/extJavaが出力するメッセージ"を参照して停止失敗の原因を調査してください。

- ー イベントログ
- Symfoware Serverのサーバログ
- コンテナのサーバログ
- ー コンテナのJavaVMログ
- ー ドメインのサーバログ
- ー ドメインのJavaVMログ

また、停止時の異常によって、不正にプロセスが残存する場合があります。本現象が発生した場合は、"8.10.3 PL/extJavaの起動失敗による異常"を参照して、残存しているプロセスを回収してください。

上記の調査を行っても原因を特定できない場合は、FJQSSコマンドで資料採取を行った上で当社技術員(サポート)に連絡してください。

8.12 分散トランザクションの異常時の対処

.NETのトランザクションスコープなど、分散トランザクションを利用したアプリケーションの運用時に、サーバがダウンするなどのシステム 異常が発生すると、トランザクションがインダウト状態になることがあります。このとき、トランザクションで占有した資源がロックされ、他の トランザクションから当該資源へのアクセスがブロックされて利用不可になります。

以降にインダウトトランザクションの確認方法と解決方法を説明します。

インダウトトランザクションの確認方法

分散トランザクションを利用したアプリケーションの動作中に、サーバまたはクライアントがダウンした場合、インダウトトランザクションが 発生している可能性があります。

確認方法を以下に示します。

サーバがダウンした場合

1. サーバの再起動時のログに、以下のようなメッセージが出力されていると、インダウトトランザクションが発生していると判断できます。

例

LOG: 準備されたトランザクション2103を復旧しています

2. システムビューpg_prepared_xactsを参照して、準備されたトランザクションに関する情報を取得します。

準備されたトランザクションの一覧にあるトランザクション識別子 (pg_prepared_xacts のtransactionカラム)が、再起動時のログから取得したインダウトトランザクションのトランザクション識別子と一致している場合、その行がインダウトトランザクションに関する情報です。

例

transactionカラムが2103の行に、インダウトトランザクションに関する情報が出力されています。

クライアントがダウンした場合

クライアントが1つも接続していないときに、pg_prepared_xacts に準備されたトランザクションが存在した場合、そのトランザクション はインダウト状態であると判断できます。

クライアントが1つ以上接続しているときに、pg_prepared_xactsに準備されたトランザクションが存在しても、インダウト状態かどうか判 断できません。この場合、下記のクエリで、取得したデータベース名、ユーザー名、PREPARE TRANSACTION を実行した時刻と アクセス先のテーブル名の情報から、インダウトトランザクションを特定してください。

select gid, x. database, owner, prepared, l. relation::regclass as relation from pg_prepared_xacts x left join pg_locks | on l.virtualtransaction = '-1/' ||x.transaction and l.locktype='relation';

これだけでは特定できない場合は、十分に時間が経過した後に、再度pg_prepared_xactsを調べてください。

前回調べたときから継続しているトランザクションがあれば、そのトランザクションがインダウト状態である可能性が高いと判断できます。

関 ポイント

ここで説明したように、確実にインダウトトランザクションを特定する汎用的な方法は存在しません。

これまでに述べた方法で特定できるように、何らかの補助的な情報の採取(例:クライアント側でのロギング)や、運用方法(例:業務 毎にデータベースユーザーを割り当てる)を検討してください。

インダウトトランザクションの解決方法

前述のシステムビューpg_prepared_xactsからインダウトトランザクションのグローバルトランザクション識別子 (pg_prepared_xacts のgidカ ラム)を取得し、ROLLBACK PREPARED文またはCOMMIT PREPARED文を発行することで、インダウトトランザクションを解決しま す。

.....

길 例

• インダウトトランザクションをロールバックする場合

postgres=# rollback prepared '374cc221-f6dc-4b73-9d62-d4fec9b430cd'; ROLLBACK PREPARED

インダウトトランザクションをコミットする場合

postgres=# commit prepared '374cc221-f6dc-4b73-9d62-d4fec9b430cd'; COMMIT PREPARED

8.13 ディスク障害以外の入出力異常

実際にはディスクが故障していなくても、故障している場合と同じ入出力異常のメッセージが出力されることがあります。 これには、以下のような場合があります。それぞれについて対処を示します。

- ・ 8.13.1 外部ディスクとの間のネットワーク異常
- ・ 8.13.2 電源未投入やマウントによる異常

8.13.1 外部ディスクとの間のネットワーク異常

外部ディスクとの間のネットワーク経路に発生する異常です。

イベントログやサーバログの内容、さらにディスク装置のランプ、ネットワーク配線、およびネットワークカードの状態などから異常の原因 を特定したあと、問題のある機器の取り替えなどの処置を行って、異常を取り除いてください。

8.13.2 電源未投入やマウントによる異常

ディスク装置の電源を投入し忘れたり、ディスクの自動マウントの設定を忘れていた、あるいはマウントを誤って解除してしまったために 発生する異常です。

イベントログおよびサーバログの内容や、ディスク装置の電源ランプ、ドライブのマウント状態などを確認し、問題があった場合は、対処 を行ってください。

マウントが解除されてしまっている場合は、誤ってマウントを解除してしまったか、あるいは、既定の設定(OSの起動時に自動でマウントする)を、マウントしない設定に変更している可能性があります。この場合は、自動でマウントを行うように設定してください。

8.14 PL/extJavaの運用操作の異常

pgx_jadminコマンドがエラーとなった場合には、出力されるエラーメッセージと、"メッセージ集"の"PL/extJavaが出力するメッセージ" を参照して問題を解決してください。また、必要に応じてPL/extJavaのログを参照してください。PL/extJavaのログの詳細は、"付録G PL/ extJavaのログ情報"を参照してください。

pgx_jadminコマンドはPL/extJavaを起動しているため、発生する異常のほとんどは、PL/extJavaに起因したインスタンス起動時の異常と同じです。代表的な例については、"8.10.3 PL/extJavaの起動失敗による異常"を参照してください。

8.15 アプリケーションの運用に関する異常(PL/extJava運用時)

8.15.1 Java関数の異常

Javaアプリケーションの実行に失敗した場合には、Java関数がSQLSTATE 39000でエラー復帰します。Symfoware Serverのサーバログを確認してから、必要に応じて以下のログにIJServer12000番台のエラーが出ていないかを確認してください。

- ・ イベントログ
- ドメインのサーバログ
- コンテナのサーバログ
- ・ コンテナのJavaVMログ
- Webサーバのトレースログ
- Webサーバの内部ログ

以下にいくつかの例を示します。

Java EE DASサービスの異常終了

Java EE DASサービスが異常終了しても自動的に再起動されるため業務は継続可能ですが、異常が継続して発生している場合や、 再度発生しないように、以下のログファイルに出力されるメッセージ番号に「IJServer」を含むメッセージに対して、"メッセージ集"の"PL/ extJavaが出力するメッセージ"を参照して原因を調査してください。

- ・ イベントログ
- ドメインのサーバログ

PCMIサービスの異常終了

PCMIサービスが監視しているサーバインスタンス(JavaVM)、およびJava EE DASサービスが残存するため、"8.10.3 PL/extJavaの起動失敗による異常"を参照して、これらのプロセスを強制終了して、原因を取り除いた後、インスタンスを再起動してください。

Javaヒープ不足

以下のログにメモリ不足を示すエラーメッセージが出力され、Javaアプリケーションが実行できない場合は、コンテナ内のサーバインスタンス(JavaVM)のヒープ領域サイズおよびPerm領域サイズを変更する必要があります。

- ・ コンテナのサーバログ
- ・コンテナ内のサーバインスタンス(JavaVM)のログ



コンテナ内のサーバインスタンス(JavaVM)のヒープ領域およびPerm領域のデフォルト値としては、以下の値が設定されています。

種類	設定値
サーバインスタンス(JavaVM)のヒープ領域サイズ	512メガバイト
サーバインスタンス(JavaVM)のPerm領域サイズ	192メガバイト

以下の手順で、コンテナ内のサーバインスタンス(JavaVM)のヒープ領域サイズおよびPerm領域サイズを変更してください。

1. pgx_jadminコマンドのlist-jvm-optionsサブコマンドで、現在のサーバインスタンス(JavaVM)のヒープ領域サイズとPerm領域サイズを確認します。

🔰 例

> pgx_jadmin list-jvm-options --dbname db01

ヒープ領域サイズおよびPerm領域サイズは、以下の確認項目の出力結果で確認してください。

オプション	確認項目
ヒープ領域サイズ	-Xmx
Perm領域サイズ	-XX:MaxPermSize



list-jvm-optionsサブコマンドの詳細については、"リファレンス"の"pgx_jadmin"を参照してください。

2. インスタンスを停止します。

インスタンスを停止します。

.

停止方法については、"2.1 インスタンスの起動と停止"を参照してください。

インスタンスの停止に失敗した場合は、"8.11 インスタンス停止失敗時の対処"を参照してください。

3. pgx_jadminコマンドのmodify-jvm-optionsサブコマンドで、サーバインスタンス(JavaVM)のヒープ領域サイズおよびPerm領域サイズの設定値を変更します。

🂕 例

サーバインスタンス(JavaVM)のヒープ領域サイズ「1024(メガバイト)」およびPerm領域サイズ「384(メガバイト)」に変更する場合

.

> pgx_jadmin modify-jvm-options --dbname dbO1 --heapsize 1024 --permsize 384

.

🐴 参照

modify-jvm-optionsサブコマンドの詳細については、"リファレンス"の"pgx_jadmin"を参照してください。

4. インスタンスを起動します。

インスタンスを起動します。

起動方法については、"2.1 インスタンスの起動と停止"を参照してください。

インスタンスの起動に失敗した場合は、"8.10インスタンス起動失敗時の対処"を参照してください。

ドメインルートのディスク故障

ドメインルートが配置されたディスクの故障や、誤操作によりドメインルート配下の資源が壊れた場合の対処については、"8.10.3 PL/ extJavaの起動失敗による異常"を参照してください。

8.15.2 Java関数の無応答

Java関数が長時間応答を返さない場合は、以下の可能性が考えられます。それぞれの対処を行ってください。

Java関数の呼び出し元が更新したレコードをJavaアプリケーションでも更新しようとしている

Java関数の呼び出し元とJavaアプリケーションとは、異なるトランザクションで動作しているため、Java関数が既に更新したレコードをJava アプリケーションが更新しようとすると、ロックの獲得処理の待ちが復帰しなくなります。このような状況は、pg_locksビューで確認してください。詳細は、"PostgreSQL 文書"内、"サーバの管理"の"データベース活動状況の監視"の"ロックの表示"を参照してください。

Javaアプリケーションの処理において、時間のかかる処理を行っている

以下のファイルに記載されたプロセスIDに対して、jstackコマンドを使用してフルスレッドダンプを採取して実行中の処理内容を確認してください。jstackコマンドの詳細については、"F.2 jstackを利用する場合の注意"を参照してください。

[ドメインルート]¥domains¥domain1¥config¥pid

付録A パラメータ

Symfoware Serverのpostgresql.confファイルに設定するパラメータについて説明します。postgresql.confファイルは、データ格納先に配置されています。

• core_directory(文字列)

コアファイルを出力するディレクトリを指定します。本パラメータを指定しない場合は、データ格納先のディレクトリを指定したものと みなされます。本パラメータは、インスタンス起動時にパラメータ指定することでのみ設定できます。インスタンス起動中に動的変更 することはできません。

• core_contents(文字列)

コアファイルに含める内容を指定します。

- full:サーバプロセスのメモリの内容をすべてコアファイルに出力します。
- none:コアファイルを出力しません。
- minimum:サーバプロセスのうち、共有メモリ以外の内容をコアファイルに出力します。コアファイルのサイズは小さくなります。
 ただし、場合によってはコアファイルが出力された原因の調査に必要な情報が十分に得られないことがあります。

本パラメータを指定しない場合は、"minimum"を指定したものとみなされます。本パラメータは、インスタンス起動時にパラメータ指定することでのみ設定できます。インスタンス起動中に動的変更することはできません。

• keystore_location(文字列)

キーストアのファイルを格納するディレクトリを指定します。他のデータベースクラスタとは異なる場所を指定してください。本パラ メータは、インスタンス起動時にパラメータ指定することでのみ設定できます。インスタンス起動中に動的変更することはできません。

• tablespace_encryption_algorithm(文字列)

作成するテーブル空間の暗号化アルゴリズムを指定します。有効な値は AES128、AES256、または none です。noneを指定した 場合、暗号化は行いません。デフォルト値は none です。暗号化を行う場合、AES256の指定を推奨します。スーパーユーザーの みこの設定を変更できます。

• backup_destination(文字列)

pgx_dmpallコマンドがバックアップデータを格納するディレクトリを絶対パスで指定します。他のデータベースクラスタとは異なる場所を指定してください。本パラメータは、インスタンス起動時にパラメータ指定することでのみ設定できます。インスタンス起動中に動的変更することはできません。

このディレクトリは、バックアップするデータ格納先のディレクトリ、テーブル空間ディレクトリ、およびトランザクションログ格納先の ディレクトリの外に配置してください。このディレクトリの内容はデータベースシステムが管理するため、利用者は任意のファイルを 格納しないように注意してください。

• search_path(文字列)

Oracleデータベース互換のSUBSTR関数を利用する場合は、"search_path"パラメータに、"oracle"および"pg_catalog"を設定して ください。"oracle"は、"pg_catalog"よりも前に指定する必要があります。



search_path = '"\$user", public, oracle, pg_catalog'



- search_pathは、スキーマ検索パスの優先順位を指定する機能です。Oracleデータベース版のSUBSTR関数はoracleスキーマ に定義されます。

- search_pathについては、"PostgreSQL文書"の"サーバの管理"の"文の動作"を参照してください。

• track_waits(文字列)

pgx_stat_lwlockおよびpgx_stat_latchに対し、統計情報の収集を有効にします。

- on:統計情報の収集を有効にします。
- off:統計情報の収集を無効にします。

本パラメータを指定しない場合は、"on"を指定したものとみなされます。

スーパーユーザーのみ、この設定を変更できます。

• track_sql (文字列)

pgx_stat_sqlに対し、統計情報の収集を有効にします。

- on:統計情報の収集を有効にします。
- off:統計情報の収集を無効にします。
- 本パラメータを指定しない場合は、"on"を指定したものとみなされます。

スーパーユーザーのみ、この設定を変更できます。



パスを指定する場合は、以下に注意してください。

- ・パスの区切り文字は¥¥を指定する。
- ・パスに空白を含む場合は""で囲む。



•••	•••	•••	•••	•••	••	•••	•••	••	••	••	••	••	••	••	••	••	••	••	••	••	••	••	•••	• • •	•••	••	••	••	••	••	•	• •	••	••	••	••	••	••	••	••	••	••	••	••	•••	•••	••
F	記り	以外	0	pos	tgre	sql	.co	nf⊘	ン	(ラ)	メー	ータ	に	21	17	には	. '	'Po	ostg	gre	SÇ)L	文書	書"	の	"+	+-	-バ	の)管	理	"0)"	サ・	-)	ŇГ.	こ設	t定	"?	を参	≷照	RL'	7<	くだ	さい	۰°	
•••					• •	• •		• •	•••	••	••	••	••	••	••	••	••	• •	•••	•••	•••	• •		• • •	••	••	••	• •	•••	• •	•	• •	• •	••	••	••	••	••	••	•••	• •	••	•••	••	•••	•••	••

付録B システム管理関数

Symfoware Serverのシステム管理関数について説明します。

🚉 参照

その他のシステム管理関数の詳細は、"PostgreSQL文書"の"SQL言語"の"システム管理関数"を参照してください。

B.1 WAL二重化制御関数

以下の表は、WAL二重化に基づくバックアップ/リカバリに使用できる関数を示しています。

表B.1 WAL二重化制御関数

名前	戻り型	説明
pgx_pause_wal_multiplexing()	void	WAL多重化を停止する
pgx_resume_wal_multiplexing()	void	WAL多重化を再開する
pgx_is_wal_multiplexing_paused()	boolean	WALの多重化が停止されていれば真を返す

WALの多重化が構成されていない場合、これらの関数はエラーを返却します。postgresql.confのbackup_destinationパラメータを設定 すると、WALの多重化が構成されます。

これらの関数は、スーパーユーザーのみ実行可能です。

B.2 透過的データ暗号化制御関数

以下の表は、透過的データ暗号化に使用できる関数を示しています。

表B.2 透過的データ暗号化制御関数

名前	戻り型	説明
pgx_open_keystore(passphrase text)	void	キーストアをオープンする
pgx_set_master_key(passphrase text)	void	マスタ暗号化キーを設定する
pgx_set_keystore_passphrase(old_passphrase text, new_passphrase text)	void	キーストアのパスフレーズを変更する

pgx_open_keystoreは、指定したパスフレーズを使ってキーストアをオープンします。キーストアをオープンすると、マスタ暗号化キーが データベースサーバのメモリにロードされます。これにより、暗号化データへのアクセスや暗号化テーブル空間を作成できます。キース トアがすでにオープンしている場合、この関数はエラーを返却します。

この関数は、スーパーユーザーのみ実行可能です。また、トランザクションブロック内でこの関数を実行することはできません。

pgx_set_master_keyは、マスタ暗号化キーを生成してキーストアに格納します。キーストアがまだ存在しない場合、キーストアを作成します。すでにキーストアが存在する場合、マスタ暗号化キーを変更します。もしキーストアがオープンされていなければ、キーストアがオープンされます。

パスフレーズは8~200バイトの文字列です。

この関数は、スーパーユーザーのみ実行可能です。また、トランザクションブロック内でこの関数を実行することはできません。キースト アはオープンしていてもオープンしていなくてもかまいません。

pgx_set_keystore_passphraseはキーストアのパスフレーズを変更します。old_passphraseには現在のパスフレーズを、new_passphraseには新しいパスフレーズを指定します。

パスフレーズは8~200バイトの文字列です。

この関数は、スーパーユーザーのみ実行可能です。また、トランザクションブロック内でこの関数を実行することはできません。キーストアはオープンしていてもオープンしていなくてもかまいません。

付録C システムビュー

Symfoware Serverのシステムビューについて説明します。

```
参照
その他のシステムビューの詳細は、"PostgreSQL文書"の"内部情報"の"システムビュー"を参照してください。
```

C.1 pgx_tablespaces

pgx_tablespacesカタログはテーブル空間の暗号化に関する情報を提供します。

名前	型	参照先	説明
spctablespace	oid	pg_tablespace.oid	テーブル空間のOID
spcencalgo	text		テーブル空間の暗号化アルゴリズム

spcencalgo列は次のいずれかの値を示します。

- none :テーブル空間は暗号化されていません
- ・ AES128 :キー長128ビットのAES
- ・ AES256 :キー長256ビットのAES

C.2 pgx_stat_lwlock

pgx_stat_lwlockビューは軽量ロックの内容ごとに1行の形で、発生に関する統計情報を示します。

表C.1 pgx_stat_lwlockビュー

列	型	説明
lwlock_name	name	軽量ロックの名前です。
total_waits	bigint	軽量ロックが原因で待ちが発生した回数です。
total_wait_time	double precision	軽量ロックが原因で待ち期間として費やされた、ミリ秒単 位の総時間です。
stats_reset	timestamp with timezone	これらの統計情報がリセットされた最終時刻です。

C.3 pgx_stat_latch

pgx_stat_latchビューはSymfoware Server内部の待機情報について、内容ごとに1行の形で、発生に関する統計情報を示します。

$100.2 \mu y$ stat $atom \perp$

列	型	説明
latch_name	name	ラッチの名前です。
total_waits	bigint	ラッチが原因で待ちが発生した回数です。
total_wait_time	double precision	ラッチが原因で待ち期間として費やされた、ミリ秒単位 の総時間です。
stats_reset	timestamp with timezone	これらの統計情報がリセットされた最終時刻です。

C.4 pgx_stat_walwriter

pgx_stat_walwriterビューはWALの書込みに関する統計情報を1行のみで表示します。

表C.3 pgx_stat_walwriterビュー

列	型	説明
dirty_writes	bigint	WALレコードの追加時にWALバッファが満杯だったため、古いWALバッファをディスクに書き込んだ回数です。
writes	bigint	WALの書込み回数です。
write_blocks	bigint	WALの書込みブロック数です。
total_write_time	double precision	WALの書込みに費やされた、ミリ秒単位の総時間で す。
stats_reset	timestamp with timezone	これらの統計情報がリセットされた最終時刻です。

C.5 pgx_stat_sql

pgx_stat_sqlビューはSQL文の種類ごとに1行の形で、SQL文の発生回数に関する情報を示します。

表C.4 pgx_stat_sqlビュー

列	型	説明
selects	bigint	SELECT文が実行された回数です。
inserts	bigint	INSERT文が実行された回数です。
deletes	bigint	DELETE文が実行された回数です。
updates	bigint	UPDATE文が実行された回数です。
declares	bigint	DECLARE文が実行された回数です。(カーソルのOPEN 回数)
fetches	bigint	FETCH文が実行された回数です。
checkpoints	bigint	CHECKPOINT文が実行された回数です。
clusters	bigint	CLUSTER文が実行された回数です。
copies	bigint	COPY文が実行された回数です。
reindexes	bigint	REINDEX文が実行された回数です。
truncates	bigint	TRUNCATE文が実行された回数です。
locks	bigint	ロックが発生した回数です。
stats_reset	timestamp with timezone	これらの統計情報がリセットされた最終時刻です。

付録D WebAdminのWebサーバ機能の起動と停止

WebAdminを使用して、Symfoware Serverのインスタンスを作成・管理するには、Symfoware Serverをインストールしたサーバで、あらかじめWebAdminのWebサーバ機能を起動する必要があります。

WebAdminのWebサーバ機能の起動方法、停止方法について説明します。

D.1 WebAdminのWebサーバ機能の起動

以下の手順でWebAdminのWebサーバ機能を起動してください。

1. [サービス]ウィンドウを表示する

[管理ツール]で、[サービス]をクリックします。

2. サービスを起動する

表示名"Symfoware WebAdmin <バージョン>"を選択し、[サービスの開始]ボタンをクリックします。

net start, またはsc startコマンドにWebAdminのWebサーバ機能のサービス名を指定して起動することもできます。

D.2 WebAdminのWebサーバ機能の停止

WebAdminのWebサーバ機能の停止方法について説明します。

以下の手順でWebAdminのWebサーバ機能を停止してください。

1. [サービス]ウィンドウを表示する

[管理ツール]で、[サービス]をクリックします。

2. サービスを停止する

表示名"Symfoware WebAdmin <バージョン>"を選択し、[サービスの停止]ボタンをクリックします。

net stop, またはsc stopコマンドにWebAdminのWebサーバ機能のサービス名を指定して停止することもできます。

付録E 障害調査情報の採取

環境構築や運用中に発生したトラブルの原因が判明しない場合、初期調査のための情報を採取します。 初期調査のための情報の採取方法について説明します。

FJQSS(資料採取ツール)で、初期調査のための情報を採取してください。

💦 参照

FJQSSの使用方法については、以下で表示されるマニュアルを参照してください。

- Windows Server(R) 2012、およびWindows Server(R) 2012 R2の場合
 [アプリ]メニューの[FJQSS(資料採取ツール)]の[FJQSS ユーザーズガイド]
- ・Windows Server(R) 2008 R2以前の場合
 - [スタート]メニューの[FJQSS(資料採取ツール)]の[FJQSS ユーザーズガイド]

🔓 注意

・ FJQSSで、初期調査のための情報を採取する際は、以下の環境変数を設定する入力画面が表示されます。

- PGDATA

データ格納先を設定してください。

- PGPORT

インスタンスのポート番号を設定してください。デフォルトのポート番号(26500)を変更していない場合は設定不要です。

- PGUSER

データベースのスーパーユーザーを設定してください。 設定したデータベースのスーパーユーザーは、クライント認証ができるよう設定してください。 FJQSSはtemplate1データベースに対してTCP/IP接続を行い、データベースの情報を取得します。

また、データベース多重化運用を行う場合には、以下の環境変数も設定してください。

- MCCONTROLDIR

Mirroring Controller管理ディレクトリを指定してください。 Mirroring Controller管理ディレクトリの詳細については、"クラスタ運用ガイド"の"Mirroring Controllerの資源"を参照してください。

・ データベース多重化運用を行う場合、FJQSSの操作はインスタンス管理者ユーザーで行ってください。

付録F PL/extJavaの注意事項

PL/extJavaを利用する際の注意事項を説明します。

F.1 スレッドダンプツールの使い方

スレッドダンプツールとは

スレッドダンプツールは、JavaVMのスレッドダンプ(コマンドプロンプトから[Ctrl]キーを押しながら[Break]キーを押し採取したスレッドダンプ)と同じく、各スレッドの状態を出力し、プログラム全体や、各スレッドの実行状況を解析する助けとなるスレッドダンプを出力する機能です。

スレッドダンプツールは、JavaVMのスレッドダンプ出力に以下の機能を追加しています。

• ファイル出力

オプション(-f)で指定したファイルに、スレッドダンプを出力できます。

• ヘッダー、フッター出力

標準出力や標準エラー出力にスレッドダンプを出力した場合、Javaアプリケーションのログファイル(コンテナの標準出力、あるいは JavaVMの出力を格納するファイルなど)に、そのスレッドダンプが含まれることになります。ヘッダーやフッターは、そのログファイル を参照する場合に、目印となる情報です。(-fオプションを指定した場合も出力します。) ヘッダーには、スレッドダンプの採取日時、コマンドラインおよび、メモリの使用状況の情報が含まれます。

動作環境

JDK 7

2種類のコマンド

スレッドダンプツールには、以下の2種類のコマンドがあります。

- ・ 通常のJavaアプリケーション(ユーザーが起動するプログラム)として動作するthdumpコマンド
- ・Windowsサービス(システム内に常駐して動作するプログラム)として動作するthdumpSVCコマンド

これら2種類のコマンドは、以下にしたがって使い分けてください。

- ・ 通常のJavaアプリケーションとして動作している場合は、thdumpコマンドを使用してください。
- ・Windowsサービスとして動作している場合は、thdumpSVCコマンドを使用してください。

なお、各コマンドには、機能的な相違点はありません。

格納先

• thdumpコマンド:

[Symfoware Serverインストールディレクトリ]¥java¥jdk7¥tools¥thdump

・ thdumpSVCコマンド:

[Symfoware Serverインストールディレクトリ]¥java¥jdk7¥tools¥thdump¥thdumpSVC

F.1.1 thdumpコマンドの使用方法

thdump.exeの格納先ディレクトリを環境変数PATHに設定したうえで実行するか、thdump.exeをフルパスで指定して実行してください。

形式

thdump.exe [-f logfile | -j] -p processid

または

thdump.exe [-f logfile | -j] program name with ".exe"

各オプションの詳細は、"F.1.3 オプション"を参照してください。

使用方法

・ "java.exe"で起動したJavaアプリケーションのスレッドダンプをJavaアプリケーションの標準出力に出力する場合:

C:¥> thdump java.exe

・ プロセスIDが123番のプロセスのスレッドダンプを"a.log"に出力する場合:

C:¥> thdump -f a.log -p 123

Internet Explorer + JBKプラグインで動作しているアプレットのスレッドダンプを、JBKプラグインのエラーログ(jbktrace.0)に出力する場合:

C:¥> thdump -j iexplore.exe

[jbkplugin.propertiesファイル]

jbk.plugin.debug.tracedir=C:¥temp jbk.plugin.debug.showvmmsg=true

F.1.2 thdumpSVCコマンドの使用方法

thdumpSVCコマンドは、Windowsサービスとして動作するスレッドダンプを採取する場合に使用します。 スレッドダンプの採取は、次の手順で操作します。

- 1. Windowsサービス登録
- 2. スレッドダンプツール開始
- 3. スレッドダンプツール停止
- 4. Windowsサービス削除

2と3の操作を繰り返すことにより、複数のスレッドダンプを採取することができます。 スレッドダンプの採取が完了したら、4の"Windowsサービス削除"を行ってください。

- ・ Windowsサービス登録/削除にはWindows(R)の"InstallUtil.exe"を内部的に使用します。
- ・スレッドダンプツールの開始および終了には、"sc.exe"を使用します。
- スレッドダンプツールの操作は、管理者権限で実行したコマンドプロンプトから操作する必要があります。管理者権限でコマンドプロンプトを実行するには、コマンドプロンプトの実行ファイルをエクスプローラー上で右クリックし、"管理者権限で実行する"を選択します。
- ・ thdumpSVCコマンドを使用するためには、Microsoft(R).NET FrameworkV2.0を動作環境にインストールしてください。

Windowsサービスへの登録方法

【コマンド形式】

thdumpSVC -Install

スレッドダンプツールをWindowsサービスとして登録するには、thdumpSVCコマンドの格納先をカレントディレクトリにするか、thdumpSVC コマンドをフルパスで指定して、上記コマンドを実行してください。

なお同じシステムで登録可能なスレッドダンプツールは1つだけです。2つ目以降を登録しようとした場合、コマンド実行時エラーとなります。

【確認方法】

Windowsサービスに登録できたかどうかは、Windows(R)の管理ツール"サービス"で確認できます。名前項目に"Thdump Service"が 表示されていれば、登録に成功しています。 コマンド実行後に、画面にWindows(R)の制御によるメッセージが表示されます。 正常に登録できた場合は、メッセージの最後に次の文字列が表示されます。

コミット段階が正常に終了しました。

トランザクション インストールが完了しました。

異常が発生した場合は、メッセージの最後に次の文字列が表示されます。

トランザクション インストールが完了しました。

インストールが失敗し、ロールバックが実行されました。

すでにスレッドダンプツールの登録が完了している場合には異常が発生しますので、Windows(R)の管理ツール"サービス"でスレッド ダンプツールの有無を確認してください。

スレッドダンプツールの開始方法

【コマンド形式】

sc start thdumpService [-f logfile | -j] -p processed

または

sc start thdumpService [-f logfile | -j] program name with ".exe"

各オプションの詳細は、"F.1.3 オプション"を参照してください。

スレッドダンプツールを開始するには、上記コマンドを実行してください。コマンド実行後に、スレッドダンプ採取が行われます。

なお同じシステムで開始可能なスレッドダンプツールは1つだけです。すでにスレッドダンプツールが動作している場合に、上記コマンドを実行すると、コマンド実行時エラーとなります。

【確認方法】

スレッドダンプツールを開始できたかどうかは、Windows(R)の管理ツール"サービス"で確認できます。状態項目に"開始"が表示されていれば、開始に成功しています。

コマンド実行後に、画面にscコマンドの制御によるメッセージが表示されます。

異常が発生した場合には、次のどれかの文字列が表示されます。なお、異常時のメッセージ以外は、無視しても問題ありません。

"[SC] StartService FAILED nnnn:"(nnnnは数字)

"[SC] StartService: OpenService FAILED nnnn:"(nnnnは数字)

スレッドダンプツールが未登録の場合か、すでにスレッドダンプツールが開始している場合には異常が発生しますので、Windows(R)の管理ツール"サービス"でスレッドダンプツールの状況を確認してください。 異常時のメッセージの出力例を、次に示します。

スレッドダンプツールが未登録の場合: "[SC] StartService: OpenService FAILED 1060:"

すでにスレッドダンプツールが開始している場合: "[SC] StartService FAILED 1056:"

スレッドダンプツールの停止方法

【コマンド形式】

sc stop thdumpService

スレッドダンプツールを停止するには、上記コマンドを実行してください。

【確認方法】

スレッドダンプツールを停止できたかどうかは、Windows(R)の管理ツール"サービス"で確認できます。状態項目が空白であれば、停止に成功しています。

コマンド実行後に、画面にscコマンドの制御によるメッセージが表示されます。

異常が発生した場合には、次のどれかの文字列が表示されます。なお、異常時のメッセージ以外は、無視しても問題ありません。

"[SC] ControlService FAILED nnnn:"(nnnnは数字)

"[SC] OpenService FAILED nnnn:"(nnnnは数字)

スレッドダンプツールが未登録の場合か、すでにスレッドダンプツールが停止している場合には異常が発生しますので、Windows(R)の管理ツール"サービス"でスレッドダンプツールの状況を確認してください。

異常時のメッセージの出力例を、次に示します。

スレッドダンプツールが未登録の場合:

"[SC] OpenService FAILED 1060:"

すでにスレッドダンプツールが停止している場合: "[SC] ControlService FAILED 1062:"

Windowsサービスからの削除方法

【コマンド形式】

thdumpSVC -Install -u

スレッドダンプツールをWindowsサービスから削除するには、thdumpSVCコマンドの格納先をカレントディレクトリにするか、thdumpSVC コマンドをフルパスで指定して、上記コマンドを実行してください。

【確認方法】

Windowsサービスから削除できたかどうかは、Windows(R)の管理ツール"サービス"で確認できます。名前項目に"Thdump Service" が表示されていなれば、削除に成功しています。

~コマンド実行後に、画面にWindowsの制御によるメッセージが表示されます。

~正常に削除できた場合は、メッセージの最後に次の文字列が表示されます。

サービス 'thdumpService' は正常にシステムから削除されました。 アンインストールか完了しました。

異常が発生した場合は、メッセージの最後に次の文字列が表示されます。

アンインストール中に例外が発生しました。この例外は無視され、アンインストール続行します。ただし、アンインストール完了後、アプリケーションは完全にアンインストールされない可能性があります。

すでにスレッドダンプツールが削除されている(未登録の)場合には異常が発生しますので、Windows(R)の管理ツール"サービス"で スレッドダンプツールの有無を確認してください。

エラーログファイル

Windowsサービスにはコンソールがありません。スレッドダンプ採取時にエラーが発生したかどうかを確認するには、エラーログファイルを参照してください。

エラーログファイルの格納先は、thdumpSVCコマンドの格納先ディレクトリです。thdumpSVCコマンドの格納先ディレクトリは書込み可能な状態にしてください。

エラーログファイルは、Windowsサービス登録時に作成され、Windowsサービス削除までの間に発生したエラーを記録します。問題な くスレッドダンプを取得できた場合は、エラーログファイルのサイズは0バイトになります。

F.1.3 オプション

スレッドダンプツールの起動時(開始時)に指定するオプションを、下表に示します。

オプション	説明
-f logfile	スレッドダンプの出力先を指定します。
	省略した場合、標準出力または標準エラー出力に出力されます。
	-jオプションと同時に指定できません。

オプション	説明
-j	ヘッダーおよびフッターをJavaVMの出力ルーチンを用いて出力する場合に指定します。
	JNI(Java Native Interface)のJNI_CreateJavaVM()関数に、vfprintf_hookオプションを指定して起動 したJavaVMに対して指定します。JBKプラグインのjbkplugin.propertiesファイル に、"jbk.plugin.debug.showvmmsg=true"と指定した場合は、このオプションを使います。
	-fオプションと同時に指定できません。
-p processid	対象とするプロセスをプロセスIDで指定します。
	下の「program name with ".exe"」オプションと同時に指定できません。
program name with ".exe"	対象とするプロセスをjava.exeなど".exe"で終わるコマンド名で指定します。
	コマンド名が指定されると、スレッドダンプツールは、現在システムで動作しているすべてのプロセスのうち、指定されたコマンド名により起動されたプロセスから1つを選択して処理します。同じコマンドから複数のプロセスを起動している場合は、-pオプションを使用して、プロセスIDを指定してください。
	-pオプションと同時に指定できません。

「program name with ".exe"」以外の、-f、-j、-pオプションの指定の順序は任意です。

また、-f、-pのオプションを複数回指定した場合は、最後に指定したものが有効となります。

コマンドを実行すると、ヘッダー、スタックトレース、フッターの順に出力されます。

対象プロセスは、プロセスID、または".exe"で終わるプログラム名(java.exeなど)で指定します。プロセスIDはWindows(R)の"タスクマ ネージャ"で確認することができます。

F.2 jstackを利用する場合の注意

JDK 7に含まれるjstackツールをトラブルシューティングツールとして使用する際は、以下に注意してください。

・ -mオプションは使用できません。



jstackツールの詳細については、JDK 7ドキュメントを参照してください。本製品に搭載しているJDK 7のドキュメントは以下のURLにあります。

http://docs.oracle.com/javase/7/docs/

付録G PL/extJavaのログ情報

PL/extJavaのエラーや警告などをログファイルに出力します。

エラー発生有無や、エラー原因を特定するために有益な情報となります。

G.1 ドメイン

ドメインが出力するログに関して説明します。

Java EE DASサービスやコンテナを制御するドメインのログが出力されます。

G.1.1 サーバログ

出力先

サーバログは以下のファイルに出力されます。

[ドメインルート]¥domains¥domain1¥logs¥server.log

出力タイミング

- 世代数
 - 1
- ・ サイズ

1,048,576/バイト

・レコード

可変

• ローテーション条件

ログサイズを超えた場合、古い情報はローテーション日時を付加したファイル名で保存されます。

server.log_2012-01-07T15-23-30

出力例

[#|2014-03-18T18:20:16.653+0900|INF0|||_ThreadID=1:_ThreadName=main:|SEC1010: Entering Security Startup Service|#]

[#|2014-03-18T18:20:16.653+0900|INF0|||_ThreadID=1;_ThreadName=main;|SEC1143: Loading policy provider com. sun. enterprise. security. provider. PolicyWrapper. |#]

[#|2014-03-18T18:20:17.230+0900|INF0|||_ThreadID=1;_ThreadName=main;|SEC1115: Realm [admin-realm] of classtype [com. sun. enterprise. security. auth. realm. file. FileRealm] successfully created. [#]

[#|2014-03-18T18:20:17.261+0900|INF0|||_ThreadID=1:_ThreadName=main;|SEC1115: Realm [file] of classtype [com. sun. enterprise. security. auth. realm. file. FileRealm] successfully created. |#]

[#|2014-03-18T18:20:17.277+0900|INF0|||_ThreadID=1;_ThreadName=main;|SEC1115: Realm [certificate] of classtype [com. sun. enterprise. security. auth. realm. certificate. CertificateRealm] successfully created. [#]

G.1.2 JavaVMログ

出力先

JavaVMログは以下のファイルに出力されます。

[ドメインルート]¥pcmi¥isje6¥logs¥server¥console.log

出力タイミング

- 世代数
- 1
- ・ サイズ

1,048,576バイト

- ・レコード
- 可変
- ローテーション条件

ログサイズを超えた場合、古い情報はローテーション日時を付加したファイル名で保存されます。 例として、ログサイズが1,048,576バイトを超えた場合は以下のファイル名で保存されます。

console.log_2014_05_14-15_56_51

出力例

02/09/2014 17:02:45

C:\Program Files (x86)\Fujitsu\symfoserver32\java\jdk7\bin\java.exe

-XX:+PrintVMTerminatedMessage

-XX:InsufficientMemoryHandler=libpcmiJavaVM-win32-x86

-Xbootclasspath/a:C: <code>Program Files (x86)</code> <code>Fujitsu</code> <code>symfoserver32</code> <code>java</code> <code>java</code> <code>jdk7</code> <code>jre</code> <code>lib</code> <code>fmoni.jar</code>

-Xrunfmoni:detail=y

-javaagent:C:\Program Files x86)\Fujitsu\Symfoserver32\Java\pcmi\lib\pcmi-java-premain.jar=C:\Fyan\domain\domain\promi Yisje6\Work\Process\PID_com.fujitsu.interstage.javaee.gf3adapter.Gf3PCMIAdapter_server -cp

C:¥Program Files (x86)¥Fujitsu¥symfoserver32¥java¥pcmi¥lib¥pcmi-api.jar;C:¥Program Files x86)¥Fujitsu¥symfoserver32¥java ¥pcmi¥lib¥protocol.jar;C:/Program Files (x86)/Fujitsu/symfoserver32/java/fjje6/glassfish/modules/glassfish.jar -XX:PermSize=64m

-XX:MaxPermSize=192m

G.2 コンテナ

コンテナが出力するログに関して説明します。

サーバインスタンス(JavaVM)上で動作するJavaアプリケーションのログが出力されます。

G.2.1 サーバログ

出力先

サーバログは以下のファイルに出力されます。

[ドメインルート]¥nodes¥localhost-domain1¥<サーバインスタンス(JavaVM)名>¥logs¥server.log

出力タイミング

- 世代数
 - 1
- ・ サイズ

```
1,048,576バイト
```

・レコード

可変

• ローテーション条件

ログサイズを超えた場合、古い情報はローテーション日時を付加したファイル名で保存されます。

例として、ログサイズが1,048,576バイトを超えた場合は以下のファイル名で保存されます。

server.log_2012-01-07T15-23-30

出力例

[#|2014-03-18T18:20:16.653+0900|INF0|||_ThreadID=1;_ThreadName=main:|SEC1010: Entering Security Startup Service|#]

[#|2014-03-18T18:20:16.653+0900|INF0|||_ThreadID=1:_ThreadName=main:|SEC1143: Loading policy provider com. sun. enterprise. security. provider. PolicyWrapper. |#]

[#|2014-03-18T18:20:17.230+0900|INF0|||_ThreadID=1;_ThreadName=main;|SEC1115: Realm [admin-realm] of classtype [com. sun. enterprise. security. auth. realm. file. FileRealm] successfully created. [#]

[#|2014-03-18T18:20:17.261+0900|INF0|||_ThreadID=1;_ThreadName=main;|SEC1115: Realm [file] of classtype [com. sun. enterprise. security. auth. realm. file. FileRealm] successfully created. |#]

[#|2014-03-18T18:20:17.277+0900|INF0|||_ThreadID=1;_ThreadName=main;|SEC1115: Realm [certificate] of classtype [com. sun. enterprise. security. auth. realm. certificate. CertificateRealm] successfully created. [#]

メッセージ詳細

PL/extJavaで出力するメッセージについて説明します。

• 出力フォーマット

メッセージは、以下のフォーマットで出力します。

[#|yyyy-MM-ddTHH:mm:ss.msTZD|LOG_LEVEL|||_ThreadID=スレッドID;_ThreadName=スレッド名:|メッセージ本文|#]

- LOG_LEVEL

「INFO」固定です。

ー メッセージ本文

PL/extJavaで出力した文字列を出力します。



以下の条件に該当する場合、上記の出力フォーマットでメッセージは出力されません。

- 利用者のJavaアプリケーションが、例外を検出した場合、例外のクラス名のみを出力します。

- 例外のスタックトレースや例外のメッセージはセキュリティを考慮して出力されません。

・ 出力フォーマット (メッセージ本文)

メッセージ本文は、以下のフォーマットで出力します。

ログレベル: symfo-plextjava: [処理結果コード] メッセージ [backend_process=PID]

ー ログレベル

ログレベルは、「ERROR」固定です。
ログレベル	出力内容
ERROR	異常発生時のエラー情報。

- 処理結果コード

処理結果コードは、以下を出力します。

コード	意味
0	正常終了
1001	内部異常を検出しました。(プロトコルの規約違反)
	富士通技術員に問い合わせてください。
2001	内部異常を検出しました。(プロトコルの規約違反)
	富士通技術員に問い合わせてください。
2002	実行対象のJavaアプリケーションが見つかりません。
	Javaアプリケーションのjarファイルを登録しているか、ファンクションの指定に誤りがないか、作成したJavaアプリケーションに誤りがないか確認してください。
	[確認観点]
	・ファンクションの登録で指定した「パッケージ名.クラス名.メソッド名」または「クラス名.メソッド 名」が正しいか。
	・ファンクションの登録で指定した「(Java関数の)引数」と、作成したJavaアプリケーションの引数の型と数が一致しているか。
	・ファンクションの登録で指定した「戻り値型」と、作成したJavaアプリケーションのメソッドの戻り 値の型が一致しているか。
	・Javaアプリケーションのメソッドが、アクセス修飾子「public」の静的メソッド(static)として作成されているか。
3001	Javaアプリケーションを実行するための権限がありません。
	Javaアプリケーションの実行権限を確認してください。
	[確認観点]
	・Javaアプリケーションのクラスのアクセス修飾子が「public」で作成されているか。
3002	Javaアプリケーションの実行中に例外が発生しました。
	データベースの状態、または、Javaアプリケーションの処理内容を確認してください。
9001	内部異常を検出しました。(環境異常)
	富士通技術員に問い合わせてください。

backend_process

Symfoware ServerプロセスのPIDを出力します。

G.2.2 JavaVMログ

出力先

JavaVMログは以下のファイルに出力されます。

[ドメインルート]¥pcmi¥isje6¥logs¥<サーバインスタンス(JavaVM)名>¥console.log

出力タイミング

- 世代数
 - 1

・ サイズ

1,048,576バイト

・レコード

可変

ローテーション条件

ログサイズを超えた場合、古い情報はローテーション日時を付加したファイル名で保存されます。 例として、ログサイズが1,048,576バイトを超えた場合は以下のファイル名で保存されます。

console.log_2014_05_14-15_56_51

出力例

05/09/2014 16:02:49

C:¥Program Files (x86)¥Fujitsu¥symfoserver32¥java¥jdk7¥bin¥java.exe

- -XX:+PrintVMTerminatedMessage
- -XX:InsufficientMemoryHandler=libpcmiJavaVM-win32-x86

-Xbootclasspath/a:C: <code>Program Files (x86)</code> <code>FujitsuFsymfoserver32FjavaFjdk7FjreFlibFfmoni.jar</code>

-Xrunfmoni∶detail=y

-javaagent:C:¥Program Files (x86)¥Fujitsu¥symfoserver32¥java¥pcmi¥lib¥pcmi-java-premain.jar=F:¥symfo¥domain¥pcmi ¥isje6¥work¥process¥.PID_com.fujitsu.interstage.javaee.gf3adapter.Gf3PCMIAdapter_db01-1

-cp C:¥Program Files (x86)¥Fujitsu¥symfoserver32¥java¥pcmi¥lib¥pcmi-api.jar;C:¥Program Files (x86)¥Fujitsu¥symfoserver32¥java ¥pcmi¥lib¥protocol.jar;C:/Program Files (x86)/Fujitsu/symfoserver32/java/fjje6/glassfish/modules/glassfish.jar -XX:PermSize=64m

-XX:MaxPermSize=192m

G.2.3 HTTPアクセスログ

出力先

HTTPアクセスログは以下のファイルに出力されます。

[ドメインルート]¥nodes¥localhost-domain1¥〈サーバインスタンス(JavaVM)名〉¥logs¥access¥server_access_log.txt

出力タイミング

- 世代数
 - 5
- ・ サイズ

1,048,576バイト

・レコード

可変

• ローテーション条件

```
1日を超えた場合、古い情報はserver_access_log.YYYY_MM_DD-hh_mm_ss.txtとなります。
前回のローテーションから1秒以内にローテーションが実行された場合は、server_access_log.YYYY_MM_DD-hh_mm_ss.通し番
号.txtとなります。
YYYY_MM_DD-hh_mm_ssは、ローテーションが実行された時刻です。
```

- 139 -

出力例

```
"127.0.0.1" "-" "14/May/2014:09:07:53 +0900" "POST /db01/callstored HTTP/1.1" "200" "1" "127.0.0.1" "-" "4207"
"ThreadID=68" "ThreadName=http-thread-pool-26532(1)" "127.0.0.1"
```

G.2.4 HTTPトレースログ

出力先

HTTPトレースログは以下のファイルに出力されます。

[ドメインルート]¥nodes¥localhost-domain1¥くサーバインスタンス (JavaVM) 名>¥logs¥http¥trace.log

出力タイミング

- 世代数
 - 10
- ・ サイズ

10,485,760バイト

・ レコード

可変

• ローテーション条件

```
ログサイズを超えた場合、古い情報はtrace.log_YYYY_MM_DD-hh_mm_ssとなります。
前回のローテーションから1秒以内にローテーションが実行された場合は、trace.log_YYYY_MM_DD-hh_mm_ss.通し番号となり
ます。
```

YYYY_MM_DD-hh_mm_ssは、ローテーションが実行された時刻です。

出力例

```
"15/May/2014:11:21:31.608" "21(Grizzly-kernel-thread(1))" "conn" "127.0.0.1:60073"
"15/May/2014:11:21:31.609" "21(Grizzly-kernel-thread(1))" "qin" "127.0.0.1:60073"
"15/May/2014:11:21:31.609" "41(admin-thread-pool-26521(2))" "qout" "127.0.0.1:60073"
"15/May/2014:11:21:31.610" "41(admin-thread-pool-26521(2))" "recv" "GET /__asadmin/get?Xhelp=true HTTP/1.1"
"15/May/2014:11:21:31.616" "41(admin-thread-pool-26521(2))" "send" "200" "127.0.0.1:60073"
"15/May/2014:11:21:31.616" "41(admin-thread-pool-26521(2))" "send" "200" "127.0.0.1:60073"
"15/May/2014:11:21:31.626" "21(Grizzly-kernel-thread(1))" "k-wt" "127.0.0.1:60073"
"15/May/2014:11:21:31.704" "21(Grizzly-kernel-thread(1))" "qin" "127.0.0.1:60073"
"15/May/2014:11:21:31.704" "48(admin-thread-pool-26521(4))" "qout" "127.0.0.1:60073"
"15/May/2014:11:21:31.704" "48(admin-thread-pool-26521(4))" "recv" "GET /__asadmin/get?DEFAULT=inst1.http-service.isjee-
trace-log.max-history-files HTTP/1.1"
"15/May/2014:11:21:31.707" "48(admin-thread-pool-26521(4))" "send" "200" "127.0.0.1:60073"
"15/May/2014:11:21:31.71" "21(Grizzly-kernel-thread(1))" "k-wt" "127.0.0.1:60073"
"15/May/2014:11:21:31.758" "21(Grizzly-kernel-thread(1))" "qin" "127.0.0.1:60073"
"15/May/2014:11:21:31.758" "37(admin-thread-pool-26521(1))" "qin" "127.0.0.1:60073"
"15/May/2014:11:21:31.758" "37(admin-thread-pool-26521(1))" "qin" "127.0.0.1:60073"
"15/May/2014:11:21:31.758" "37(admin-thread-pool-26521(1))" "qin" "127.0.0.1:60073"<"15/May/2014:11:21:31.758" "21(Grizzly-kernel-thread(1))" "disc" "127.0.0.1:60073"</p>
```

G.3 Webサーバ

Webサーバが出力するログに関して説明します。

データベースとサーバインスタンス(JavaVM)間の通信を制御する、Webサーバのログが出力されます。

G.3.1 エラーログ

Webサーバでエラーが発生したときのエラー状況をログ(エラーログ)として出力します。

出力先

エラーログは以下のファイルに出力されます。

[Symfoware Serverインストールディレクトリ]¥java¥ahs22¥plextjava¥domain1¥logs¥errorlog

出力タイミング

- 世代数
 - 5
- ・ サイズ

1,048,576バイト

・ レコード

可変

• ローテーション条件

ログサイズを超えた場合、古い情報はerrorlog.Nとなります。

Nは通番で、0から始まる連続の番号です。新しいファイルの順に、1番目のファイルの通番は「.0」、N番目のファイルの通番は「. (N-1)」となります。

• 出力内容

エラーログは、Webサーバでエラーが発生した際に出力されます。

エラーログとして出力されるログの形式を以下に示します。

[日時] [ログレベル] [client IPアドレス] (エラー番号)エラー説明: メッセージ本文

出力項目

出力形式に対応する各出力項目を以下に示します。

- 日時
 - エラー発生日時

「[曜日月日時:分:秒西暦]」の形式で出力されます。

・ ログレベル

ログレベル	エラーログの重度	初期設定時の出力
emerg	緊急事態発生のエラー	0
alert	修正しなければ動作不可のエラー	0
crit	直ちに処理すべきエラー	0
error	あまり重要ではなく、無視してもよいエラー	0
warn	警告であり、無視してもよいエラー	0
notice	通常起こりうる特筆すべき通知情報	0
info	notice以外の通知情報 環境定義を編集した場合に出力されます。	×
debug	モジュールの開発/デバッグ中のログ 環境定義を編集した場合に出力されます。	×

・ client IPアドレス

クライアントまたはプロキシサーバなどのIPアドレス クライアントからのWebサーバアクセス時にエラーが発生した場合、出力されます。 本項目は、省略される場合があります。 • エラー番号

オペレーティングシステムのエラー番号 本項目は、省略される場合があります。

- エラー説明
 エラー番号の説明
 本項目は、省略される場合があります。
- ・ メッセージ本文

メッセージの本文

出力例

[Fri Sep 05 16:03:10 2014] [notice] Child 10752: Child process is running [Fri Sep 05 16:03:10 2014] [notice] Child 10752: Acquired the start mutex. [Fri Sep 05 16:03:10 2014] [notice] Child 10752: Starting 3000 worker threads.

G.3.2 トレースログ

Webサーバへの入出力情報をログ(トレースログ)として出力します。

出力先

トレースログは以下のファイルに出力されます。

[Symfoware Serverインストールディレクトリ]¥java¥ahs22¥plextjava¥domain1¥logs¥tracelog

出力タイミング

- 世代数
 - 5
- ・ サイズ

2,097,152バイト

・ レコード

可変

• ローテーション条件

ログサイズを超えた場合、古い情報はtracelog.Nとなります。

Nは通番で、0から始まる連続の番号です。新しいファイルの順に、1番目のファイルの通番は「.0」、N番目のファイルの通番は「. (N-1)」となります。

- 出力内容
 - トレースログは、各トレース情報に対して、以下のタイミングで出力されます。
 - クライアントとの入出力情報
 - TCPコネクション接続時
 - TCPコネクション切断時
 - HTTPリクエスト受信時
 - HTTPレスポンス送信時
 - プラグインモジュールとの入出力情報
 - プラグインモジュールの応答処理関数の呼出し時
 - プラグインモジュールの応答処理関数の復帰時

出力形式

トレースログとして出力されるログの形式を以下に示します。

[日時][プロセスID][スレッドID]イベント 詳細情報

出力項目

出力形式に対応する各出力項目を以下に示します。

日時

トレース情報の出力日時 「[日/月/西暦:時:分:秒.ミリ秒]」の形式で出力されます。

・プロセスID

デーモンプロセスのプロセスID

・ スレッドID

通信スレッドのスレッドID

・ イベント 詳細情報

各出力契機におけるトレース内容のフォーマットを以下に示します。

出力契機	イベント	詳細情報
TCPコネクション接続時	conn	クライアントのIPアドレス:ポート番号=> 着信したWebサーバのIPアドレス:ポート番号
TCPコネクション切断時	disc	
HTTPリクエスト受信時	recv	"リクエスト行" エスケープした文字列に変換して出力されます。
HTTPレスポンス送信時	send	ステータスコード
プラグインモジュール応答処理関数の呼 出し時(注1)	call	モジュールソース名(注2)
プラグインモジュール応答処理関数の復 帰時(注1)	rtn	モジュールソース名(復帰コード)(注2) 復帰コードには、以下が出力されます。
		・ 本モジュールで応答処理を実行していない場合:-1
		・ 本モジュールで応答処理を実行した場合:-1以外

注1)出力対象のプラグインモジュールは、Webサーバの提供モジュール以外のモジュールです。

注2) プラグインモジュールのコンパイル時のソースファイル名です。Servletサービスが提供するWebサーバコネクタのモジュールソース名は、「mod_jk2.c」と出力されます。



トレースログにより、HTTP Keep-Alive機能の接続保持時間を確認できます。出力されたトレースログにおいて、disc (TCPコネクション 切断時のイベント)およびsend (HTTPレスポンス送信時のイベント)の日時を確認し、以下の計算式で算出します。

HTTP Keep-Alive機能の接続保持時間=discの日時-sendの日時

備考)タイムアウト発生時、WebサーバからクライアントへのTCPコネクション切断要求に対して、クライアントからの切断通知を2秒 以内に受信できなかった場合、接続保持時間は、KeepAliveTimeoutディレクティブの設定より2秒長くなることがあります。 出力例

```
[15/May/2014:11:34:34.550] [5144] [5157] conn 127.0.0.1:44217=>127.0.0.1:26530
[15/May/2014:11:34:34.550] [5144] [5157] recv "POST /db01/callstored HTTP/1.1"
[15/May/2014:11:34:34.551] [5144] [5157] call mod_jk2.c
[15/May/2014:11:34:34.576] [5144] [5157] rtn mod_jk2.c(0)
[15/May/2014:11:34:34.576] [5144] [5157] send 400
[15/May/2014:11:34:34.591] [5144] [5157] disc
```

G.3.3 アクセスログ

Webサーバでは、Webブラウザからのアクセス状況をログ(アクセスログ)として出力します。

出力先

アクセスログは以下のファイルに出力されます。

[Symfoware Serverインストールディレクトリ]¥java¥ahs22¥plextjava¥domain1¥logs¥accesslog

出力タイミング

- 世代数
- 5
- ・ サイズ

1,048,576バイト

・レコード

可変

• ローテーション条件

ログサイズを超えた場合、古い情報はaccesslog.Nとなります。

Nは通番で、0から始まる連続の番号です。新しいファイルの順に、1番目のファイルの通番は「.0」、N番目のファイルの通番は「. (N-1)」となります。

• 出力内容

アクセスログは、Webサーバがクライアントからリクエストを受信し、受信したリクエストのレスポンスをクライアントへ送信するときに出力されます。

出力形式

アクセスログの初期値のフォーマット(ahs-analysis)で出力されるログの形式を以下に示します。アクセス時に指定されない項目は、ハイフン「-」で出力されます。

ホスト名 ユーザー名識別 ユーザー名 日時 "リクエスト" ステータスコード データ転送量 WebサーバのIPアドレス:ポート番号 Host ヘッダ プロセスID 処理時間 リクエストID

出力項目

出力形式に対応する各出力項目を以下に示します。

出力項目	出力内容
ホスト名	クライアントのIPアドレス/ホスト名
ユーザー名識別	クライアントから返信されるユーザー個人情報
ユーザー名	クライアントから送信されたユーザー名
日時	クライアントからのリクエストを受け付けた日時 「[日/月/西暦:時:分:秒 GMTとの時差]」の形式で出力されます。

出力項目	出力内容
リクエスト	クライアントが要求したリクエスト内容
ステータスコード	クライアントに返信するコード
データ転送量	クライアントに転送したデータ量
WebサーバのIPアドレス:ポート番号	リクエストを受け付けたWebサーバのIPアドレスおよびポート番号
Hostヘッダ	クライアントから送られてきたHostヘッダの内容
プロセスID	リクエストを処理したプロセスのプロセスID
処理時間	リクエストを受け付けてから処理が完了するまでの時間 フォーマットは、「マイクロ秒」の形式で出力されます。
リクエストID	リクエストごとに付与される一意なID

出力例

```
127.0.0.1 - - [15/May/2014:11:32:17 +0900] "POST /db01/callstored HTTP/1.1" 200 3 127.0.0.1:80 127.0.0.1 5143 26901 -
```

G.3.4 内部ログ

Webサーバコネクタでエラーが発生したときのエラー状況をログとして出力します。

出力先

エラーログは以下のファイルに出力されます。

[Symfoware Serverインストールディレクトリ]¥java¥wsc¥logs¥jk2¥httpd_domain1.conf¥jk2.log

出力タイミング

- 世代数
- 2
- ・ サイズ
 - 1,048,576/バイト
- ・ レコード
- 可変
- ローテーション条件

指定ログサイズ、または指定時間を超えた場合、古い情報はjk2_YY.MM.DD_hh.mm.ss.logとなります。

出力例

[05/09/2014 16:03:10:382 +0900] (info) IJServer12047: Web Server Connector running. conf="C:\Program Files (x86)\Fujitsu \Fujitsu \F

索引

[数字] 2層の暗号化キーとキーストア
[B]
backup_destination(义子列)122
[C]
core_directory(文字列)
[K]
keystore_location(文字列)122
[P]
pgx_dmpall、pgx_rcvallコマンドによるバックアップ・リカバリ41
pgx_stat_latchビュー126
pgx_stat_lwlockビュー126
pgx_stat_sqlビュー127
pgx_stat_walwriterビュー127
PL/extJavaの概要
PL/extJavaの構成
PL/extJavaの導入と運用
PL/extJavaのログ情報135

[S]

search_path(文字列)	122
Symfoware Server Java アプリケーションサーバ	59

[T]

tablespace_encryption_algorithm(文字列)	
track_sql	123
track waits	123

[W]

WebAdminの起動URL	4
WebAdminの利用環境	4
WebAdminへのログイン	5

[あ]

暗号化の仕組み	
暗号化の範囲	
インスタンス起動失敗時の対処	111
インスタンスとの切断	10
インスタンスの稼働状態の確認	16,18
インスタンスの起動	15,18
インスタンスの停止	15,18
インスタンスへの接続	9

[か]

各格納先ディスクの異常	111
監視メニュー	3
記憶領域のゼロ・オーバヘッド	33
強力な暗号化アルゴリズムを利用	33
キーストアの自動オープン	45
キーストア・ファイルの配置	45
キーストア・ファイルの配置と自動オープン	44
継続的アーカイブによるバックアップとポイントインタイムリン	カバリ <mark>42</mark>
コンテナ	58

[5]	
サーバインスタンス(JavaVM)	<u>59</u>
スタンバイサーバの構築と起動	45
設定メニュー	4

[た]

データ格納ディスクまたはトランザクションログ格納ディス	クに障
害が発生した場合	76
データ格納ディスク、またはトランザクションログ格納ディ	スクに
障害が発生した場合	80
データベース活動状況の監視	53
ドメイン	59
ドメインルート	59

[は]

バックアップ周期	24
バックアップ状態	
バックアップデータ格納ディスクに障害が発生した場	合78,81
バックアップの実行	. 25,27,28
バックアップの目安時間	
ハードウェアに基づく暗号化/復号の高速化	
ファイルシステムレベルのバックアップとリストア	

#