

Interstage Application Server



移行ガイド

Windows(64)/Linux(64)

J2UL-1332-02Z0(00)
2011年4月

まえがき

本書の目的

本書は、“Interstage Application Server 移行ガイド”です。以前のバージョン・レベルから本バージョン・レベルへ移行する場合の注意事項について説明します。

前提知識

本書を読む場合、以下の知識が必要です。

- C言語に関する基本的な知識
- C++言語に関する基本的な知識
- COBOLに関する基本的な知識
- Java言語に関する基本的な知識
- インターネットに関する基本的な知識
- オブジェクト指向技術に関する基本的な知識
- 分散オブジェクト技術(CORBA)に関する基本的な知識
- リレーショナルデータベースに関する基本的な知識
- LDAPおよびX.500に関する基本的な知識
- 使用するOSに関する基本的な知識

本書の構成

本書は以下の構成になっています。

第1章 他製品との差異について

他製品との差異について説明しています。

第2章 旧バージョン・レベルからの変更

旧バージョン・レベルからの変更内容、資源の互換の有無について説明しています。

第3章 旧バージョン・レベルからの資源の移行方法

旧バージョン・レベルからの資源の移行方法について説明しています。

第4章 旧機能から新機能への移行方法

旧機能から新機能への移行方法について説明しています。

第5章 Interstage運用操作、環境の移行

Interstage運用操作、環境の移行について説明しています。

第6章 J2EEの移行

J2EEの移行について説明しています。

第7章 OLTP機能の移行

OLTP機能の移行について説明しています。

第8章 その他の機能の移行

その他の機能の移行について説明しています。

付録A V5.1以前のServletサービス環境定義の移行

V5.1以前のServletサービスの環境定義をV6以降のServletサービスへ移行する定義項目の対応を説明しています。

製品の表記について

本書での以下の表記については、それぞれの基本ソフトウェアに対応した製品を示しています。

表記	説明
RHEL-AS4(x86)	Red Hat Enterprise Linux AS (v.4 for x86)を前提基本ソフトウェアとしたInterstage Application Server/Interstage Web Server
RHEL-AS4(EM64T)	Red Hat Enterprise Linux AS (v.4 for EM64T)を前提基本ソフトウェアとしたInterstage Application Server/Interstage Web Server
RHEL-AS4(IPF)	Red Hat Enterprise Linux AS (v.4 for Itanium)を前提基本ソフトウェアとしたInterstage Application Server
RHEL5(x86)	Red Hat Enterprise Linux 5 (for x86)を前提基本ソフトウェアとしたInterstage Application Server/Interstage Web Server
RHEL5(Intel64)	Red Hat Enterprise Linux 5 (for Intel64)を前提基本ソフトウェアとしたInterstage Application Server/Interstage Web Server
RHEL5(IPF)	Red Hat Enterprise Linux 5 (for Intel Itanium)を前提基本ソフトウェアとしたInterstage Application Server
RHEL6(x86)	Red Hat Enterprise Linux 6 (for x86)を前提基本ソフトウェアとしたInterstage Application Server/Interstage Web Server
RHEL6(Intel64)	Red Hat Enterprise Linux 6 (for Intel64)を前提基本ソフトウェアとしたInterstage Application Server/Interstage Web Server

輸出許可

本ドキュメントを非居住者に提供する場合には、経済産業大臣の許可が必要となる場合がありますので、ご注意ください。

著作権

Copyright 2010-2011 FUJITSU LIMITED

2010年10月 初版 2011年 4月 第2版

目 次

第1章 他製品との差異について.....	1
1.1 Windows製品間の差異について.....	1
1.2 Linux製品間の差異について.....	4
1.3 Windows Server(R) x64 Editions, Linux for Intel64上での資源の移行についての注意事項.....	6
第2章 旧バージョン・レベルからの変更.....	7
2.1 主な変更点.....	7
2.2 インストーラの変更点.....	9
2.3 提供を停止した機能.....	10
2.4 旧版マニュアルからの変更点.....	10
2.5 メッセージの変更.....	12
2.5.1 コンソール出力の変更について.....	12
2.5.2 メッセージ番号がDEPで始まるメッセージ.....	12
2.5.3 メッセージ番号がesで始まるメッセージ.....	13
2.5.4 メッセージ番号がihsで始まるメッセージ.....	14
2.5.5 メッセージ番号がIJServer1で始まるメッセージ.....	17
2.5.6 メッセージ番号がIJServer2で始まるメッセージ.....	17
2.5.7 メッセージ番号がIJServer3で始まるメッセージ.....	20
2.5.8 メッセージ番号がisで始まるメッセージ.....	20
2.5.9 メッセージ番号がisj2eeで始まるメッセージ.....	22
2.5.10 メッセージ番号がjmsで始まるメッセージ.....	22
2.5.11 メッセージ番号がodで始まるメッセージ.....	23
2.5.12 メッセージ番号がSvMonで始まるメッセージ.....	24
2.5.13 EJBサービスのメッセージ.....	24
2.5.14 SSL環境設定コマンドのメッセージ.....	26
2.5.15 CORBAサービスのマイナーコード.....	29
2.5.16 Interstage シングル・サインオンのメッセージ.....	29
2.5.17 Interstage ディレクトリサービスのメッセージ.....	31
2.5.18 J2EE使用時に出力される例外情報.....	54
第3章 旧バージョン・レベルからの資源の移行方法.....	55
3.1 移行方法.....	55
3.1.1 移行手順.....	55
3.2 旧バージョン・レベルのユーザ資源の互換.....	55
3.2.1 Interstageの動作環境定義ファイル.....	56
3.2.2 Interstage HTTP Serverの資源.....	57
3.2.3 J2EEの資源.....	57
3.2.4 Servletサービスの資源.....	57
3.2.5 Interstage JMSの資源.....	58
3.2.6 SOAPサービスの資源.....	58
3.2.7 Interstage証明書環境の資源.....	60
3.2.8 Interstage シングル・サインオンの資源.....	61
3.2.9 Interstage ディレクトリサービスの資源.....	61
3.2.10 CORBAサービスの資源.....	64
3.2.11 コンポーネントトランザクションサービスの資源.....	66
3.2.12 データベース連携サービスの資源.....	67
3.2.13 Portable-ORBの資源.....	68
3.2.14 イベントサービスの資源.....	68
第4章 旧機能から新機能への移行方法.....	71
4.1 Interstage HTTP Server (Apache HTTP Server 2.0ベース) への移行.....	71
4.1.1 8.0以前のInterstage HTTP Server (Apache HTTP Server 1.3ベース) からの移行.....	71
4.1.2 V7.0以前のInfoProvider Proからの移行.....	84
4.2 Servletサービス(Tomcat 5.5ベースのサーブレット実行環境)への移行.....	92
4.2.1 Tomcat3.1ベースのServletサービスからの移行について.....	92

4.2.2 Servletサービス移行時の注意	93
4.2.3 アプリケーションの非互換一覧	99
4.3 InfoDirectoryからInterstage ディレクトリサービスへの移行	108
4.3.1 InfoDirectoryとInterstage ディレクトリサービスの機能比較	109
4.3.2 Interstage ディレクトリサービスへ移行できないデータ	112
4.3.3 Interstage ディレクトリサービスへの移行手順	112
4.3.4 C言語アプリケーションの移行手順	113
4.3.5 Java言語(JNDI)アプリケーションの移行	123
4.3.6 LDAPコマンドの移行	123
4.4 EJBサービス(IJServer)への移行方法	128
4.4.1 EJBの高速呼び出し機能とLight EJBコンテナ機能からIJServerへの移行	128
4.4.2 Interstage Application Server V3.xからの移行	130
4.4.3 ロードバランス機能について	132
第5章 Interstage運用操作、環境の移行	133
5.1 Interstage操作の移行	133
5.2 Interstage統合コマンドの移行	135
5.3 Interstage管理コンソールの変更内容	135
5.4 Interstage JMXサービスの移行	139
5.5 マルチサーバ管理環境の移行	140
5.6 Interstage HTTP Serverの移行	142
5.7 ワークユニットの移行	143
5.8 JDK/JRE、Javaツールの変更内容	143
5.9 Interstage証明書環境の移行	145
5.10 Interstage ディレクトリサービスの移行	146
5.11 Interstage シングル・サインオンの移行	148
5.11.1 Interstage Application Server V9.0以降からのサーバの移行	153
5.11.2 Interstage Application Server V7.0/8.0からのサーバの移行	154
5.11.3 Interstage Application Server Standard Edition、Enterprise Edition V6.0からのサーバの移行	157
5.11.4 Interstage Application Server Plus V6.0からのリポジトリサーバの移行	157
5.11.5 Interstage Application Server Plus V6.0からの認証サーバの移行	163
5.11.6 Interstage Application Server Plus V6.0からのリポジトリサーバ、および認証サーバの移行	170
5.11.7 Interstage Application Server Plus、Web-J Edition V6.0からの業務サーバの移行	172
5.11.8 Interstage Application Server V5.1からのサーバの移行	179
5.11.9 Javaアプリケーションの移行	179
第6章 J2EEの移行	183
6.1 J2EEの追加機能	183
6.1.1 J2EE定義コマンド	183
6.1.2 Webサービス機能	184
6.1.3 アプリケーションファイル保護レベル	184
6.1.4 Javaヒープ/Java Permanent領域不足時の制御	185
6.2 J2EEアプリケーションの移行	185
6.2.1 IJServerの移行について	186
6.2.2 コンパイル時・実行時にクラスパスに追加するjarについて	192
6.2.3 XMLパーサのXerces2サポート	192
6.2.4 Webサーバのバーチャルホストの表示変更について	193
6.2.5 データベースについて	193
6.2.6 JDBCドライバ	196
6.2.7 自動再接続機能について	196
6.2.8 SQL実行の通信待ち時間監視について	197
6.2.9 コネクション使用時間監視について	197
6.2.10 クラスローダの変更について	197
6.2.11 返却される例外の詳細文字列について	201
6.2.12 JNDIから返却される例外について	201
6.2.13 J2EEアプリケーションの移行時のその他の注意事項	202
6.3 INTERSTAGE V3.xからJ2EEアプリケーションへの移行	202
6.3.1 サーブレット-EJB連携の移行	203

6.3.2 J2EEアプリケーションクライアントーEJB連携の移行	205
6.4 Servletサービスの移行	208
6.5 EJBサービスの移行	209
6.5.1 Interstage Application Server V9.0.1/V9.1での変更内容	209
6.5.2 Interstage Application Server V9.0での変更内容	211
6.5.3 Interstage Application Server 8.0での変更内容	211
6.5.4 Interstage Application Server V7.0での変更内容	213
6.5.5 Interstage Application Server V6.0での変更内容	216
6.5.6 EJBアプリケーション移行時の注意点	218
6.5.7 Interstage Application Server V5.x以前からの移行	219
6.6 Interstage JMSの移行	219
6.6.1 Interstage Application Server V9.0での変更内容	219
6.6.2 Interstage Application Server V7.0での変更内容	219
6.7 Interstage Webサービスの移行	221
第7章 OLTP機能の移行	222
7.1 ワークユニットの移行	222
7.1.1 Interstage Application Server / Interstage Web Server V9.2での変更内容	222
7.2 CORBAワークユニットのサポートに伴うアプリケーションの移行	224
7.3 CORBAワークユニットの移行	225
7.4 CORBAサービスの移行	225
7.4.1 Interstage Application Server V9.2での変更内容	225
7.4.2 Interstage Application Server / Interstage Web Server V9.1での変更内容	226
7.4.3 Interstage Application Server / Interstage Web Server V9.0での変更内容	229
7.4.4 Interstage Application Server 8.0での変更内容	231
7.4.5 Interstage Application Server V7.0での変更内容	231
7.4.6 Interstage Application Server V6.0での変更内容	235
7.4.7 Interstage Application Server V5.1での変更内容	237
7.4.8 INTERSTAGE Application Server V3.xからの移行	237
7.5 イベントサービスの移行	240
7.5.1 Interstage Application Server V9.0での変更内容	240
7.5.2 Interstage Application Server 8.0での変更内容	240
7.5.3 Interstage Application Server V6.0での変更内容	242
7.6 コンポーネントトランザクションサービスの移行	242
7.7 データベース連携サービスの移行	242
7.7.1 Interstage Application Server V9.0での変更内容	242
7.7.2 Interstage Application Server V7.0での変更内容	243
7.7.3 Interstage Application Server V5.xからの移行方法	243
7.7.4 INTERSTAGE Application Server V3.xからの移行	244
7.8 Portable-ORBの移行	249
7.8.1 Interstage Application Server V9.1での変更内容	249
7.8.2 Interstage Application Server V9.0での変更内容	249
第8章 その他の機能の移行	250
8.1 MessageQueueDirectorの移行	250
8.2 負荷計測エージェントの移行	250
8.3 SOAPサービスの移行	251
8.3.1 J2EEのWebサービス機能への移行	251
8.3.2 SOAPサービス環境の注意	255
8.3.3 Interstage Application Server V6.0以前からの移行方法	256
8.3.4 CORBA/SOAPゲートウェイの移行	257
8.3.4.1 V5.1 RPCアプリケーションの移行	258
8.3.4.2 特定のデータ型を使用するSOAPサーバアプリケーションの対応	259
8.3.4.3 Webサービス情報管理ファイルの移行ーバックアップコマンドによる移行	261
8.3.4.4 Webサービス情報管理ファイルの移行ーsoapdd2wsddコマンドによる移行	263
付録A V5.1以前のServletサービス環境定義の移行	265
A.1 JServlet環境定義の移行	265

A.2 サブレット・ゲートウェイ環境定義の移行.....	268
A.2.1 移行元の環境で使っていたWebサーバがInterstage HTTP Serverの場合.....	268
A.2.2 移行元の環境で使っていたWebサーバがInterstage HTTP Server以外の場合.....	269
A.3 サブレット・コンテナ環境定義の移行.....	270
A.4 JServletプロパティファイルの移行.....	273
索引.....	274

第1章 他製品との差異について

1.1 Windows製品間の差異について

Windows Server(R) for Itanium-based SystemsおよびWindows Server(R) x64 Editionsの製品と、他のWindows製品のInterstage Application Serverでは、いくつかの機能差異があります。以下にサーバ環境における新機能追加以外の差異を示します。クライアント環境については新機能追加以外の差異はありません。

なお、本製品の新機能については“Interstage Application Serverアップデート情報”を参照してください。

サービス	機能	説明	参照マニュアル
Web Package	無償配布用Webサーバ	提供していません。	—
JDK/JRE	JDK/JRE	Java HotSpot(TM) Client VMをサポートしていません。	—
	JDK/JRE 1.4.2	Windows Server(R) x64 Editionsの製品では提供していません。	—
	JDK/JRE 6	Windows Server(R) for Itanium-based Systemsの製品では提供していません。	—
	New世代領域サイズ自動調整機能	提供していません。	—
	Javaヒープ量	32ビットOSでの設定に対して、1.5～2倍のJavaヒープ量が必要になります。また、スタックサイズも2倍になります。	使用上の注意「注意事項」の「JDK/JREの注意事項」
J2EE	データベース	Postgre SQLへの接続をサポートしていません。データベースを使用する場合は、Symfoware Server、OracleまたはSQL Serverを使用してください。SQL Serverは、SQL Server 2005をサポートします。	J2EEユーザーズガイド
	ツール	以下の機能を使用することはできません。 ・ j2eejndisetupコマンド	—
	Tomcat4.1ベースのServletサービス	Windows Server(R) x64 Editionsの製品では提供していません。Windows Server(R) for Itanium-based Systems、および他のWindows製品では提供しています。	—
CORBA サービス	アプリケーション開発	Interstage Application Server Enterprise Editionにおいて、サーバアプリケーション開発で使用可能な言語はC、C++、Java、COBOLです。OOCOBOLを使用することはできません。なお、C++言語のPOAは使用することができません。 IDLファイルから生成されるマッピング言語用のファイルは、アプリケーションが動作するシステムで作成したものを使用してください。	アプリケーション作成ガイド(CORBAサービス編)
	アプリケーション開発(コンパイルオプション)	Interstage Application Server Enterprise Editionでのアプリケーション開発において、32bitアプリケーションでは“構造体メンバのアライメント”の設定値は“4 バイト”ですが、64bitアプリケーションでは設定値に“8 バイト”を指定してください。	アプリケーション作成ガイド(CORBAサービス編)

サービス	機能	説明	参照マニュアル
	キュー制御機能	queue_policy定義ファイルを使用したキューポリシーの設定はできません。	—
	ロードバランス	ネーミングサービスと連携したロードバランスの運用をすることはできません。負荷分散のために、IPCOMを利用してください。	高信頼性システム運用ガイド「IPCOMを利用した負荷分散」
	COM/CORBA連携機能	COM/CORBA連携機能を使用することはできません。	—
ディレクトリサービス	標準データベース	Windows Server(R) for Itanium-based Systemsの製品では提供していません。RDBを使用してください。	ディレクトリサービス運用ガイド
	ディレクトリサービス	Windows Server(R) for Itanium-based Systems向けのInterstage Application Server Standard-J Editionでは提供していません。	—
シングル・サインオン	リポジトリサーバ機能	Windows Server(R) for Itanium-based Systems向けのInterstage Application Server Standard-J Editionでは提供していません。	—
SOAPサービス	CORBA/SOAPゲートウェイ	提供していません。	—
UDDIサービス	UDDIレジストリ	提供していません。	—
コンポーネントトランザクションサービス	トランザクションアプリケーション	サポートしていません。既存のトランザクションアプリケーションを本製品による運用環境に移行する場合はCORBAアプリケーションに変更してください。	アプリケーション作成ガイド(CORBAサービス編)「CORBAワークユニット」の「旧バージョンのアプリケーションからの移行方法」
ラッパーオブジェクト	ラッパーオブジェクト	提供していません。	—
運用操作	Interstage 運用API	提供していません。	—
	Interstage統合コマンド	以下のコマンドにおいて、ロードバランスオブション、CORBA/SOAPクライアントゲートウェイは操作できません。 • isinit • ismodifyservice • isstart • isstat • isstop また、isregistdefコマンドで生成されるInterstage動作環境定義において、LBO USE、SOAP Client GWは使用できません。	リファレンスマニュアル(コマンド編)「Interstage統合コマンド」
ポータル機能	ポータル機能	提供していません。	—
負荷分散	IPCOM連携	以下のInterstage特有の負荷分散ポリシーを使用することができません。 • 最小待ちメッセージ数 • 最小通信バッファ使用率	—

サービス	機能	説明	参照マニュアル
MessageQueueDirector	MessageQueueDirector	提供機能は以下の通りです。詳細機能で未提供の機能もありますので、提供機能の詳細については“MessageQueueDirector説明書”を参照してください。 ・ イベントチャネル連携サービス 以下の機能は提供されません。 ・ プログラムインタフェース ・ メッセージキュー ・ MQDのシステム環境にデータベースを利用する ・ 同報配信サービス ・ ACM連携サービス ・ SMTP連携サービス	MessageQueueDirector説明書「イベントチャネル連携サービス」
ebXML Message Service	ebXML Message Service	提供していません。	—
MQ連携サービス	MQ連携サービス	提供していません。	—
マルチサーバ管理機能	プロビジョニング機能(Systemwalker Resource Coordinator連携)	提供していません。	運用ガイド(基本編)
	アプリケーション配付・世代管理機能(Systemwalker Software Delivery 連携)	提供していません。	運用ガイド(基本編)
データベース連携サービス	アプリケーション開発	Interstage Application Server Enterprise Editionにおいて、サーバアプリケーション開発で使用可能な言語はC、Javaです。C++、COBOLを使用することはできません。 C言語で開発する場合は、Windows 2003 Microsoft Platform SDK コンパイラを使用できます。 Java言語で開発する場合は、JDK1.4.2を使用できます。	アプリケーション作成ガイド(データベース連携サービス編)
Apworksサーバランタイム	Apworksサーバランタイム	提供していません。	—
その他	セッション情報管理機能(SMO)	提供していません。	—
	サーバマシン状態監視機能(SMM)	提供していません。	—

なお、他のWindows製品で構築した環境の資源をバックアップして、本製品で構築した環境にリストアして運用することはできません。

1.2 Linux製品間の差異について

Linux for ItaniumおよびLinux for Intel64の製品と、他のLinux製品のInterstage Application Serverでは、いくつかの機能差異があります。以下にサーバ環境における新機能追加以外の差異を示します。クライアント環境については新機能追加以外の差異はありません。なお、本製品の新機能については“Interstage Application Serverアップデート情報”を参照してください。

サービス	機能	説明	参照マニュアル
Web Package	無償配布用Webサーバ	提供していません。	—
JDK/JRE	JDK/JRE	Java HotSpot(TM) Client VMをサポートしていません。	—
	JDK/JRE 1.4.2	Red Hat Enterprise Linux 5をサポートしていません。	—
	JDK/JRE 6	Linux for Itaniumの製品では提供していません。	—
	New世代領域サイズ自動調整機能	提供していません。	—
	Javaヒープ量	32ビットOSでの設定に対して、1.5～2倍のJavaヒープ量が必要になります。	使用上の注意 「注意事項」の「JDK/JREの注意事項」
J2EE	データベース	SQL ServerおよびPostgreSQLへの接続をサポートしていません。データベースを使用する場合は、Symfoware ServerかOracleを使用してください。	J2EEユーザーズガイド
	Tomcat4.1ベースのServletサービス	Linux for Intel64の製品では提供していません。Linux for Itanium、および他のLinux製品では提供しています。	
CORBAサービス	アプリケーション開発	Interstage Application Server Enterprise Editionにおいて、サーバアプリケーション開発で使用可能な言語はC、C++、Java、COBOLです。OOCOBOLを使用することはできません。なお、C++言語のPOAは使用することができません。 また、CORBAで定義されている基本データ型とC言語でのデータ型の対応について、以下の基本データ型は他のInterstage Application Server製品の場合と差異があります。 - 本製品 CORBA_longはintに対応 CORBA_unsigned_longはunsigned intに対応 - 他のInterstage Application Server CORBA_longはlongに対応 CORBA_unsigned_longはunsigned longに対応	アプリケーション作成ガイド(CORBAサービス編)
	キュー制御機能	キュー制御機能(queue_policy)を使用することはできません。	—
	ロードバランス	ネーミングサービスと連携したロードバランスの運用をすることはできません。負荷分散のために、IPCOMを利用してください。	高信頼性システム運用ガイド 「IPCOMを利用した負荷分散」

サービス	機能	説明	参照マニュアル
ディレクトリサービス	標準データベース	Linux for Itaniumの製品では提供していません。RDBを使用してください。	ディレクトリサービス運用ガイド
	ディレクトリサービス	Linux for Itanium向けのInterstage Application Server Standard-J Editionでは提供していません。	—
シングル・サインオン	リポジトリサーバ機能	Linux for Itanium向けのInterstage Application Server Standard-J Editionでは提供していません。	—
コンポーネントトランザクションサービス	トランザクションアプリケーション	サポートしていません。既存のトランザクションアプリケーションを本製品による運用環境に移行する場合はCORBAアプリケーションに変更してください。	アプリケーション作成ガイド(CORBAサービス編) 「CORBAワークユニット」の「旧バージョンのアプリケーションからの移行方法」
運用操作	Interstage 運用API	提供していません。	—
	Interstage統合コマンド	Linux for Intel64版Interstageでは、以下のコマンドにおいて、ロードバランスオプション、CORBA/SOAPクライアントゲートウェイは操作できません。 • isinit • ismodifyservice • isstart • isstat • isstop また、isregistdefコマンドで生成されるInterstage動作環境定義において、LBO USE、SOAP Client GWは使用できません。 Linux for Itanium版Interstageでは、以下のコマンドにおいて、ロードバランスオプションは操作できません。 • isinit • ismodifyservice • isstart • isstat • isstop また、isregistdefコマンドで生成されるInterstage動作環境定義において、LBO USEは使用できません。	リファレンスマニュアル(コマンド編)「Interstage統合コマンド」
ポータル機能	ポータル機能	提供していません。	—
負荷分散	IPCOM連携	以下のInterstage特有の負荷分散ポリシーを使用することができません。 • 最小待ちメッセージ数 • 最小通信バッファ使用率	—

サービス	機能	説明	参照マニュアル
マルチ サーバ管 理機能	プロビジョニング機能(Systemwalker Resource Coordinator連携)	提供していません。	運用ガイド(基本編)
	アプリケーション配付・世代管理機能(Systemwalker Software Delivery 連携)	提供していません。	運用ガイド(基本編)
SOAP サービス	全機能	Linux for Intel64版は提供していません。	—
その他	セッション情報管理機能(SMO)	提供していません。	—
	サーバマシン状態監視機能(SMM)	提供していません。	—

なお、他のLinux製品で構築した環境の資源をバックアップして、本製品で構築した環境にリストアして運用することはできません。

1.3 Windows Server(R) x64 Editions, Linux for Intel64上での資源の移行についての注意事項

Windows Server(R) x64 Editions, Linux for Intel64上における資源の移行について注意事項を説明します。

- Windows Server(R) x64 Editions (32ビット互換) にインストールしたInterstageからWindows Server(R) x64 Editions版Interstageへの資源の移行はサポートしていません。
- Linux for Intel64 (32ビット互換) にインストールしたInterstageからLinux for Intel64版Interstageへの資源の移行はサポートしていません。

第2章 旧バージョン・レベルからの変更

2.1 主な変更点

ここでは、以下について説明します。

- [Interstage Application Server V9.2での主な変更内容](#)
- [Interstage Application Server V9.0での主な変更内容](#)
- [Interstage Application Server 8.0での主な変更内容](#)
- [Interstage Application Server V7.0での主な変更内容](#)
- [Interstage Application Server V6.0での主な変更内容](#)

■Interstage Application Server V9.2での主な変更内容

Interstage Application Server V9.2での主な変更内容を紹介します。

- **JDK/JRE 6の提供**

JDK/JRE 6を提供します。

ただし、本製品に添付されるJDK/JRE 6には、次の機能を提供していません。

- Mozilla Rhino
- Java DB
- Java Web Start(JWS)



注意

次の環境には、JDK/JRE 6は提供されません。

Windows

Windows Server(R) for Itanium-based Systems

Linux

Linux for Itanium

■Interstage Application Server V9.0での主な変更内容

Interstage Application Server V9.0での主な変更内容を紹介します。

- **Interstage HTTP Serverの変更**

Interstage HTTP Serverは、Apache HTTP Server 1.3ベースからApache HTTP Server 2.0ベースのWebサーバへ変更されました。V8.0以前のInterstage HTTP Server (Apache HTTP Server 1.3ベース) からInterstage HTTP Server (Apache HTTP Server 2.0ベース) への移行については、“[4.1.1 8.0以前のInterstage HTTP Server \(Apache HTTP Server 1.3ベース\)からの移行](#)”を参照してください。

- **ディレクトリサービスの変更**

ディレクトリサービスの名称が、Smart RepositoryからInterstage ディレクトリサービスへ変更されました。名称の変更に伴い、マニュアル名が以下に変更になりました。
“ディレクトリサービス運用ガイド”

Windows

名称の変更に伴い、サービス名が以下に変更になりました。

Interstage Directory Service(リポジトリ名)

- ・ デフォルトJDKの変更

JDK/JRE 5.0がデフォルトになりました。

JDK/JRE 5.0の新機能などについては、JDK 5.0ドキュメントの“機能ガイド”の“新機能と拡張機能”を参照してください。
本製品に添付されるJDK/JRE 5.0には、Java Web Start(JWS)を提供していません。

また、JDK/JRE 1.4も提供されますが、JDK/JRE 1.3は提供されません。



注意

次の環境には、JDK/JRE 1.4は提供されません。

Windows

Windows Server(R) 2008

Linux

RHEL5(x86)、RHEL(Intel64) およびRHEL5(IPF)

- ・ JDK/JRE 1.3から5.0への移行方法

JDK/JRE 1.3から5.0への移行方法については、オンラインマニュアルCDの次の場所にある“Java プラットフォーム移行ガイドバージョン1.3から5.0へ”を参照してください。

— ¥ApplicationServer¥JavaplatformMigration¥jm_white_paper_r6a-jp.pdf

- ・ Servlet2.4/JSP2.0のサポート

Servlet2.4/JSP2.0を実装したTomcat5.5ベースのサーブレット実行環境を提供します。

以前のサーブレット実行環境からの移行については、“[4.2 Servletサービス\(Tomcat 5.5ベースのサーブレット実行環境\)への移行](#)”を参照してください。

■Interstage Application Server 8.0での主な変更内容

Interstage Application Server 8.0での主な変更内容を紹介します。

- ・ ディレクトリサービスの変更

ディレクトリサービスが、InfoDirectoryからInterstage ディレクトリサービスへ変更されました。InfoDirectoryからInterstage ディレクトリサービスへの移行については、“[4.3 InfoDirectoryからInterstage ディレクトリサービスへの移行](#)”を参照してください。

■Interstage Application Server V7.0での主な変更内容

Interstage Application Server V7.0での主な変更内容を紹介します。

- ・ マルチサーバ管理機能の提供

複数のサーバ(マルチサーバ)で構成されたシステム上で運用するInterstageやInterstageの業務を統合的に運用管理できる機能が提供されました。複数サーバ上に配置された業務を仮想的な1つの業務として一括操作することや、複数サーバ上で運用しているInterstageの各サービスや業務を運用操作することができます。マルチサーバ管理機能の詳細は、“運用ガイド(基本編)”の“マルチサーバ管理機能”を参照してください。

■Interstage Application Server V6.0での主な変更内容

Interstage Application Server V6.0での主な変更内容を紹介します。

- ・ Interstage管理コンソールの提供

J2EEアプリケーション、Webサービス、CORBAワークユニットの環境設定・運用管理操作を“Interstage管理コンソール(GUI)”で行えます。これまで個々のツールで行っていた各アプリケーションの環境設定や運用管理等の操作を、Interstage管理コンソールだけで行えるようになります。(注)

また、運用中の資源消費・アプリケーションの処理時間などのモニタリングも、Interstage管理コンソールで行えます。

Interstage管理コンソールの詳細は、Interstage管理コンソールのヘルプを参照してください。

Windows

注) Interstage管理コンソールは、Administratorsグループ権限でのみ操作可能です。

標準提供のWebサーバの変更 Windows Solaris

標準インストールによって提供されるWebサーバがInfoProvider ProからInterstage HTTP Serverに変更されました。InfoProvider ProからInterstage HTTP Serverへの移行については、“[4.1.2 V7.0以前のInfoProvider Proからの移行](#)”を参照してください。

サーブレット実行環境の変更

Tomcat 4.1ベースのサーブレット実行環境を提供します。



注意

Interstage Application Server V9.0以降では、Tomcat5.5ベースのServletサービスを提供しています。V5.1以前のサーブレット実行環境からの移行については、“[4.2 Servletサービス\(Tomcat 5.5ベースのサーブレット実行環境\)への移行](#)”を参照してください。

J2EE実行環境の統合

Webアプリケーション(Servlet/JSP)とEJBアプリケーションの実行環境を統合し、1つのJava VMで動作可能となりました。この実行環境をIJServerと呼んでいます。

2.2 インストーラの変更点

ここでは、以下について説明します。

- ・ [コマンド、DLLファイルのインストール先の変更](#)
- ・ [インストール時のIJServer作成](#)
- ・ [Fujitsu XMLプロセッサのインストール](#)

■コマンド、DLLファイルのインストール先の変更 Windows

Interstage Application Server V7.0(Interstage V7.0)では、各サービスが提供するコマンド、DLLファイルの格納先が変更されました。Interstage Application Server V6.0(Interstage V6.0)以前の格納先の構成を意識した処理を行っている場合、新格納先に合わせて修正する必要があります。また、Interstage V6.0以前の格納先と同じ構成としたい場合、issetcompatiblepathコマンドを使用し、Interstage V6.0以前の格納先にコマンド、DLLファイルを複写できます。issetcompatiblepathコマンドの詳細は、“[リファレンスマニュアル\(コマンド編\)](#)”を参照してください。

以下に、コマンド、DLLファイルの格納先(C:\¥Interstageにインストールした場合の格納先の例)を示します。

サービス名	Interstage V6.0以前	Interstage V7.0以降
J2EE	C:\¥Interstage¥J2EE¥bin	C:\¥Interstage¥bin
SOAPサービス	C:\¥Interstage¥F3FMsoap¥bin	C:\¥Interstage¥bin
データベース連携サービス	C:\¥Interstage¥ots¥bin	C:\¥Interstage¥bin
Webサーバ(Interstage HTTP Server)	C:\¥Interstage¥F3FMIhs¥bin	C:\¥Interstage¥bin
コンポーネントトランザクションサービス	C:\¥Interstage¥extp¥bin C:\¥Interstage¥td¥bin	C:\¥Interstage¥bin
JMSサービス	C:\¥Interstage¥jms¥bin	C:\¥Interstage¥bin
UDDIレジストリサービス	C:\¥Interstage¥F3Fmuddis¥bin	C:\¥Interstage¥bin
シングル・サインオン	C:\¥Interstage¥F3Fmsso¥ssoatcsv¥bin C:\¥Interstage¥F3Fmsso¥ssoatzag¥bin C:\¥Interstage¥F3Fmsso¥ssocm¥bin	C:\¥Interstage¥bin

サービス名	Interstage V6.0以前	Interstage V7.0以降
イベントサービス	C:\¥Interstage¥eswin¥bin	C:\¥Interstage¥bin
Interstage EJB サービス	C:\¥Interstage¥ejb¥bin	C:\¥Interstage¥bin

■インストール時のIJServer作成 Solaris Linux

Interstage V7.0以降では、インストール時にIJServerが作成されません。インストール完了後に、必要に応じてInterstage管理コンソールからIJServerを作成してください。

■Fujitsu XMLプロセッサのインストール Windows

Interstage Application Server V5.x以前において、自動インストールされていたFujitsu XMLプロセッサは、標準インストールまたはカスタムインストールでは、自動インストールされなくなりました。

そのため、Fujitsu XMLプロセッサを使用する場合は、別途インストール作業が必要です。

詳細は、“インストールガイド”の“Fujitsu XMLプロセッサのインストールについて”を参照してください。

2.3 提供を停止した機能

旧製品で提供していた以下の機能は、提供を停止しました。

- ・ 負荷計測エージェント

2.4 旧版マニュアルからの変更点

■マニュアル構成の変更

Interstage Application Server V5.0(Interstage V5.0)以降では、マニュアルの構成を大幅に変更しました。ここでは、旧版マニュアルの主な記載内容が、本製品のマニュアルのどこに記載されているかを示します。

旧版マニュアル		本製品のマニュアル	提供の有無 ○:有 ×:無		
マニュアル名(*1)	内容	マニュアル名(*1)	EE (*2)	SJE (*3)	WS (*4)
ガイド デザインガイド オペレーションガイド	・ 環境設定/運用(統合コマンド) ・ バックアップ/リストア	運用ガイド(基本編)	○	○	○
	・ OLTPサーバ(CORBA アプリ、コンポーネント トランザクションサービス、データベース連携 サービス)の運用設計 ・ ワークユニット運用 (Interstage 運用 API、性能監視ツール含む) ・ 分散トランザクション機能(リソース管理プログラム)	OLTPサーバ運用ガイド	○	×	×
	・ 高性能/高信頼システム(IPCOM連携、クラスダシステム、ロードバラン ス機能)	高信頼性システム運用ガイド	○	×	×

旧版マニュアル		本製品のマニュアル	提供の有無 ○:有 ×:無		
マニュアル名(*1)	内容	マニュアル名(*1)	EE (*2)	SJE (*3)	WS (*4)
	・ セキュリティシステム (SSL/認証/アクセス制御/プロキシ連携等)	セキュリティシステム運用ガイド	○	○	○
	・ トラブル対処 (全コンポーネントのトラブル集を統合)	トラブルシューティング集	○	○	○
	・ チューニング (Interstage環境詳細設定)	チューニングガイド	○	○	○
インストールガイド	・ 資源(ディスク/メモリ)	チューニングガイド	○	○	○
	・ ソフトウェア条件	使用上の注意	○	○	○
	・ 注意事項/制限事項				
分散アプリケーション作成ガイド(CORBAサービス編)	・ 追加機能/プログラム修正一覧	アップデート情報	○	○	○
J2EEユーザーズガイド JServletユーザーズガイド EJBユーザーズガイド 分散アプリケーション作成ガイド(EJBサービス編)	(J2EEに関する情報を1冊に統合)	J2EEユーザーズガイド	○	○	○
分散アプリケーション作成ガイド(CORBAサービス編)	・ CORBAサービス	アプリケーション作成ガイド(CORBAサービス編)	○	×	×
	・ イベントサービス	アプリケーション作成ガイド(イベントサービス編)	○	×	×
リファレンスマニュアル(定義編)	・ Interstage システム 定義	運用ガイド(基本編)	○	○	○
	・ Interstage動作環境定義				
	・ CORBAサービス環境定義	チューニングガイド	○	○	○
	・ コンポーネントトランザクションサービス環境				
	・ データベース連携サービス環境定義				
	・ ワークユニット定義	OLTPサーバ運用ガイド	○	×	×
リファレンスマニュアル(API編)	・ IDL定義	アプリケーション作成ガイド(CORBAサービス編)	○	×	×
	・ Java以外のAPI	リファレンスマニュアル(API編)	○	×	×
	・ Java-API	Javadoc集	○	○	○
メッセージ集	・ 例外情報/マイナーコード	メッセージ集	○	○	○
	(全コンポーネントのメッセージ)	メッセージ集	○	○	○

*1)

先頭の製品名は省略しています。

*2)

Interstage Application Server Enterprise Edition

*3)

Interstage Application Server Standard-J Edition

*4)

Interstage Web Server

■Web公開されているマニュアル

本製品のマニュアルは以下のURLで参照できます。

<http://interstage.fujitsu.com/jp/manual/>

2.5 メッセージの変更

2.5.1 コンソール出力の変更について Solaris

Interstage Application Server V7.0以降では、コンソール(/dev/console)に出力されるメッセージが出力されないよう変更しました。旧バージョン・レベルと互換が必要な場合は、以下の環境変数を設定後、Interstageを起動してください。

```
-----  
# IS_ISV_CONSOLE_OUTPUT=ON  
# export IS_ISV_CONSOLE_OUTPUT  
-----
```



- RCプロシジャからのInterstage起動およびワークユニット起動を行う場合は、環境変数IS_ISV_CONSOLE_OUTPUTは設定しないか“OFF”を設定してください。詳細は、“運用ガイド(基本編)”の“Interstage統合コマンドによる運用操作”を参照してください。
- システムメッセージの設定(/etc/syslog.conf)によっては、コンソールにメッセージが出力されます。これは、システムログの機能によるものです。
システムメッセージの設定については、OSのマニュアルを参照してください。

2.5.2 メッセージ番号がDEPで始まるメッセージ

ここでは、以下について説明します。メッセージの詳細は、“メッセージ集”を参照してください。

- [Interstage Application Server V9.2での変更](#)
- [Interstage Application Server V9.0.1/V9.1での変更](#)
- [Interstage Application Server V9.0での変更](#)
- [Interstage Application Server V8.0での変更](#)
- [Interstage Application Server V7.0での変更](#)

■Interstage Application Server V9.2での変更

旧メッセージID	新メッセージID	変更内容
DEP1881	変更なし	ユーザの対処に内容を追記し、V9.0以降のIJServerの新規作成/定義更新時に不正なXMLパーザを指定した場合に出力されるように変更しました。

■Interstage Application Server V9.0.1/V9.1での変更

旧メッセージID	新メッセージID	変更内容
DEP1761	DEP1913	可変情報の“タグ名”を除いたDEP1913に変更しました。
DEP1764	DEP1914	可変情報の“タグ名”を除いたDEP1914に変更しました。
DEP1765	DEP1915	可変情報の“タグ名”を除いたDEP1915に変更しました。
DEP1767	DEP1916	可変情報の“タグ名”を除いたDEP1916に変更しました。

■Interstage Application Server V9.0での変更

旧メッセージID	新メッセージID	変更内容
DEP1099	DEP1877	メモリ不足が発生した場合のメッセージを、DEP1877に変更しました。
	DEP1878	制御ファイルが破損した場合のメッセージを、DEP1878に変更しました。

■Interstage Application Server 8.0での変更

旧メッセージID	新メッセージID	変更内容
DEP5100	変更なし	[ijsdeploymentコマンドから出力されるメッセージ] [ijsundeploymentコマンドから出力されるメッセージ] メッセージ本文を変更しました。 旧:“本操作は管理対象サーバでは実行できません” 新:“本操作はアプリケーション操作種別が一括操作モードに設定されている管理対象サーバでは実行できません”

■Interstage Application Server V7.0での変更

旧メッセージID	新メッセージID	変更内容
DEP4122	変更なし	出力メッセージに詳細情報を付加しました。
DEP5051	—	[ijsundeploymentコマンドから出力されるメッセージ] -aオプションで配備されていないアプリケーションのみを指定した場合に出力されましたが、出力されません。
DEP5051	DEP5072	[ijsundeploymentコマンドから出力されるメッセージ] アプリケーションの配備解除処理が正常に終了した場合に出力されるメッセージを左記のように変更し、アプリケーション単位に出力されるように変更しました。
DEP5073	変更なし	[ijsundeploymentコマンドから出力されるメッセージ] アプリケーションの配備解除処理に失敗した場合、左記のメッセージがアプリケーション単位に出力されるように変更しました。

2.5.3 メッセージ番号がesで始まるメッセージ

ここでは、以下について説明します。メッセージの詳細は、“メッセージ集”を参照してください。

- [Interstage Application Server 8.0での変更](#)

■Interstage Application Server 8.0での変更

旧メッセージID	新メッセージID	変更内容
es10055	変更なし	可変情報に“イベントチャネルグループ名::イベントチャネル名”を追加しました。
es10067	変更なし	英語メッセージの先頭の大文字を小文字に変更しました。 旧:“ES: ERROR: es10067: [ttttt] Esmonitorchnl is already running.” 新:“ES: ERROR: es10067: [ttttt] esmonitorchnl is already running.”

2.5.4 メッセージ番号がihsで始まるメッセージ

ここでは、以下について説明します。メッセージの詳細は、“メッセージ集”を参照してください。

- [Interstage Application Server V9.1での変更](#)
- [Interstage Application Server V9.0での変更](#)

■Interstage Application Server V9.1での変更

Interstage Application Server V9.1において、メッセージ番号がihsで始まるメッセージは、Interstage Application Server V9.0から以下のように変更しました。

旧メッセージID	新メッセージID	変更内容
ihs01015	変更なし	メッセージの内容を以下に変更しました。 旧(英語):“Insufficient memory. Detail=(%s1)” 新(英語):“Insufficient memory. Detail=(%s1)”
ihs81320	変更なし	メッセージの内容を以下に変更しました。 旧(英語):“Error creating new Virtual Host. Error Code:%s1, Web server name: %s2” 新(英語):“Error creating new Virtual Host. Detail:%s1, Web server name:%s2”
ihs81321	変更なし	メッセージの内容を以下に変更しました。 旧(英語):“Error deleting new Virtual Host. Error Code:%s1, Web server name: %s2” 新(英語):“Error deleting new Virtual Host. Detail:%s1, Web server name:%s2”
ihs81329	変更なし	メッセージの内容を以下に変更しました。 旧(英語):“The environment definition obtained cannot be used. Server name=%s1 Detail information: %s2” 新(英語):“The environment definition obtained cannot be used. Server name=%s1 Detail:%s2” 旧(日本語):“取得した環境定義は使用できません。サーバ名=%s1 詳細情報: %s2” 新(日本語):“取得した環境定義は使用できません。サーバ名=%s1 詳細コード: %s2”
ihs81330	変更なし	メッセージの内容を以下に変更しました。 旧(英語):“This operation cannot be executed because an error is contained in the environment setting. Detail information: %s1” 新(英語):“This operation cannot be executed because an error is contained in the environment setting. Detail:%s1”

旧メッセージID	新メッセージID	変更内容
		旧(日本語):“環境設定に異常があるため、当操作は行えません。詳細情報: %s1” 新(日本語):“環境設定に異常があるため、当操作は行えません。詳細コード: %s1”
ih81331	変更なし	メッセージの内容を以下に変更しました。 旧(英語):“This operation cannot be executed because another operation contains an error. Detail information: %s1” 新(英語):“This operation cannot be executed because another operation contains an error. Detail: %s1” 旧(日本語):“他の操作で異常があるため、当操作は行えません。詳細情報: %s1” 新(日本語):“他の操作で異常があるため、当操作は行えません。詳細コード: %s1”
ih81338	変更なし	メッセージの内容を以下に変更しました。 旧(英語):“One or more servers failed to operate the virtual host. Web server name=%s1(%s2), Detail information: %s3” 新(英語):“One or more servers failed to operate the virtual host. Web server name=%s1(%s2) Detail: %s3” 旧(日本語):“一部のサーバでバーチャルホストの操作に失敗しました。Webサーバ名: %s1(%s2) 詳細情報: %s3” 新(日本語):“一部のサーバでバーチャルホストの操作に失敗しました。Webサーバ名: %s1(%s2) 詳細コード: %s3”
ih81339	変更なし	メッセージの内容を以下に変更しました。 旧(英語):“Operation of the virtual host has failed. Web server name=%s1(%s2), Detail information: %s3” 新(英語):“Operation of the virtual host has failed. Web server name=%s1(%s2) Detail: %s3” 旧(日本語):“バーチャルホストの操作に失敗しました。Webサーバ名: %s1(%s2) 詳細情報: %s3” 新(日本語):“バーチャルホストの操作に失敗しました。Webサーバ名: %s1(%s2) 詳細コード: %s3”
ih81340	変更なし	メッセージの内容を以下に変更しました。 旧(日本語):“Webサーバの定義情報が一致しない管理対象サーバはサーバグループに追加できません。詳細情報: %s1” 新(日本語):“Webサーバの定義情報が一致しない管理対象サーバはサーバグループに追加できません。詳細コード: %s1”
ih81342	変更なし	メッセージの内容を以下に変更しました。 旧(英語):“Could not add the Managed Server because the Interstage and Web server linkage settings in it are different. Server group=%s1 Detail information: %s2” 新(英語):“Could not add the Managed Server because the Interstage and Web server linkage settings in it are different. Server group=%s1 Detail: %s2” 旧(日本語):“追加する管理対象サーバにおいてInterstageとWebサーバの連動設定が異なるため追加できません。サーバグループ=%s1 詳細情報: %s2” 新(日本語):“追加する管理対象サーバにおいてInterstageとWebサーバの連動設定が異なるため追加できません。サーバグループ=%s1 詳細コード: %s2”
ih81343	変更なし	メッセージの内容を以下に変更しました。 旧(英語):“Could not add the Reserve Server because the Interstage and Web server linkage settings in the Managed Servers that already belong to the Server Group are different. Server group=%s1 Detail

旧メッセージID	新メッセージID	変更内容
		information:%s2” 新(英語):“Could not add the Reserve Server because the Interstage and Web server linkage settings in the Managed Servers that already belong to the Server Group are different. Server group=%s1 Detail:%s2” 旧(日本語):“サーバグループに所属する管理対象サーバにおいて、Interstage とWebサーバとの連動設定が異なるため追加できません。サーバグループ=%s1 詳細情報: %s2” 新(日本語):“サーバグループに所属する管理対象サーバにおいて、InterstageとWebサーバとの連動設定が異なるため追加できません。サーバグループ=%s1 詳細コード: %s2”
ih81373	変更なし	メッセージの内容を以下に変更しました。 旧(英語):“The server cannot be added because the Web server configuration of a Managed Server in the server group is different. Web server name: %s1(%s2) Detailed information=%s3” 新(英語):“The server cannot be added because the Web server configuration of a Managed Server in the server group is different. Web server name=%s1(%s2), Detail=%s3” 旧(日本語):“サーバグループに所属する管理対象サーバにおいて、Webサーバの構成が異なるため追加できません。Webサーバ名: %s1(%s2) 詳細情報=%s3” 新(日本語):“サーバグループに所属する管理対象サーバにおいて、Webサーバの構成が異なるため追加できません。Webサーバ名= %s1(%s2) 詳細コード=%s3”
ih81374	変更なし	メッセージの内容を以下に変更しました。 旧(英語):“Cannot add Managed Server because the configuration of its Web server is different. Web server name: %s1(%s2) Detailed information=%s3” 新(英語):“Cannot add Managed Server because the configuration of its Web server is different. Server group=%s1, Detail=%s2” 旧(日本語):“追加する管理対象サーバにおいて、Webサーバの構成が異なるため追加できません。Webサーバ名: %s1(%s2) 詳細情報= %s3” 新(日本語):“追加する管理対象サーバにおいて、Webサーバの構成が異なるため追加できません。サーバグループ=%s1 詳細コード= %s2”
ih81385	変更なし	メッセージの内容を以下に変更しました。 旧(英語):“The Web server does not exist in the specified server group or server. Server group: %s1” 新(英語):“The Web server does not exist in the specified server group or server. Server name:%s1” 旧(日本語):“指定されたサーバグループまたはサーバにはWebサーバが存在しません。サーバグループ: %s1” 新(日本語):“指定されたサーバグループまたはサーバにはWebサーバが存在しません。サーバ名: %s1”
エラーログ (ログレベル: error)	変更なし	メッセージの内容を以下に変更しました。 旧:“[client %s1] SSL: Error message recived(illegal certificate). (%s2,%s3,%s4)” 新:“[client %s1] SSL: Error message received(illegal certificate). (%s2,%s3,%s4)”
エラーログ (ログレベル: error)	変更なし	メッセージの内容を以下に変更しました。 旧:“[client %s1] SSL: Not enough memory for base64 encoding. (%s2)”

旧メッセージID	新メッセージID	変更内容
		新: “[client %s1] SSL: Not enough memory for base64 encoding. (%s2)”
エラーログ (ログレベル: error)	変更なし	メッセージの内容を以下に変更しました。 旧: “[client %s1] SSL: Version negotiation error. (%s2,%s3,%s4)” 新: “[client %s1] SSL: Version negotiation error. (%s2,%s3,%s4)”

■Interstage Application Server V9.0での変更

Interstage Application Server V9.0以降において、メッセージ番号がihsで始まるメッセージは、Interstage HTTP ServerがApache HTTP Server 1.3ベースからApache HTTP Server 2.0ベースのWebサーバへ変更されたことに伴い、メッセージの出力内容が変更されます。メッセージ番号がihsで始まるメッセージの変更内容については、“[4.1.1 8.0以前のInterstage HTTP Server \(Apache HTTP Server 1.3ベース\)からの移行](#)”の“メッセージ”を参照してください。

2.5.5 メッセージ番号がIJServer1で始まるメッセージ

ここでは、以下について説明します。メッセージの詳細は、“メッセージ集”を参照してください。

- [Interstage Application Server V9.0での変更](#)

■Interstage Application Server V9.0での変更

旧メッセージID	新メッセージID	変更内容
IJServer10203 IJServer10204	変更なし	Webサーバ名が可変情報として追加になりました。
全般	—	日本語のエラーメッセージについて、エラーレベルが日本語から英語に変更になったものがあります。 また、メッセージのラベルやエラー種別の有無・形式が異なるものがあります。

2.5.6 メッセージ番号がIJServer2で始まるメッセージ

ここでは、以下について説明します。メッセージの詳細は、“メッセージ集”を参照してください。

- [Interstage Application Server V9.0での変更](#)
- [Interstage Application Server V7.0での変更](#)

■Interstage Application Server V9.0での変更

旧メッセージID	新メッセージID	変更内容
IJServer21xxx (注1)	—	イベントログファイル、またはシステムログファイルに出力されません。 (注2)

注1)

ただし、以下のメッセージは従来通りシステムログに出力されます。

- IJServer21033: JavaVM上でメモリ不足が発生しました: NAME=%s1
- IJServer21035: CORBAサービスへのアクセス時に異常が発生しました
- IJServer21087: CORBAサービスで異常が発生しました: NAME=%s1 REASON=%s2
- IJServer21410: 異常時用ファイルへのJMSメッセージ退避に失敗しました: NAME=%s1 FILE=%s2 REASON=%s3 CODE=%s4

- IJServer21439: Message-driven Beanで異常が発生したためEJBアプリケーションを停止します: NAME=%s1 CODE=%s2
- IJServer21457: イベントチャネルが停止したためMessage-driven Beanのメッセージ受信を停止します: IJServer=%s1: MDB=%s2: CHANNEL=%s3
- IJServer21999: システムエラーが発生しました: NAME=%s1 CODE=%s2

注2)

バージョンアップ前と同じように、イベントログファイル、またはシステムログファイルにメッセージを出力させたい場合は以下を設定してください。

なお、本設定は、定義変更後に起動されたIJServerに対して有効です。

定義ファイル

Windows

C:\Interstage\J2EE\etc\isj2ee.properties

Solaris

Linux

/opt/FJSVj2ee/etc/isj2ee.properties

追加工定義

```
-----
syslog_output_mode=all
-----
```

■Interstage Application Server V7.0での変更

旧メッセージID	新メッセージID	変更内容
IJServer21020	—	以下の条件の場合、コンテナログファイルに出力されません。(注1) • EJBアプリケーションがSession Beanで、createメソッドでアプリケーション例外が発生した場合 • EJBアプリケーションがSession Beanで、ビジネスメソッドでアプリケーション例外が発生した場合
IJServer21216 IJServer21239	—	以下の条件の場合、コンテナログファイルに出力されません。(注1) • EJBアプリケーションがEntity Beanで、createメソッドでアプリケーション例外が発生した場合
IJServer21235	—	以下の条件の場合、コンテナログファイルに出力されません。(注1) • EJBアプリケーションがEntity Beanで、removeメソッドでアプリケーション例外が発生した場合
IJServer21054	—	以下の条件の場合、コンテナログファイルに出力されません。(注1) • EJBアプリケーションメソッド実行中に、以下のメソッドを使用してトランザクションにロールバックをマークして、メソッドの実行に正常終了した場合 — SessionContextのsetRollbackOnlyメソッド — EntityContextのsetRollbackOnlyメソッド — MessageDrivenContextのsetRollbackOnlyメソッド • アプリケーション例外が発生した場合
IJServer21108 IJServer21113	—	イベントログファイル、またはシステムログファイルに出力されません。(注2)

旧メッセージID	新メッセージID	変更内容
IJServer21114 IJServer21115 IJServer21116 IJServer21118 IJServer21120 IJServer21124 IJServer21125 IJServer21126		
IJServer21421	IJServer21455	プロセス多重エラーで出力されていたメッセージを左記のように変更しました。(注3)

注1)

SystemWalker製品を使用してログメッセージの監視を行っていたユーザがInterstage Application Server V7.0以降にバージョンアップした時に問題となる可能性があります。

バージョンアップ前と同じように、コンテナログファイルにメッセージを出力させたい場合は以下を設定してください。

定義ファイル

Windows

C:\¥Interstage¥EJB¥etc¥FJEJBconfig.properties

Solaris

Linux

/opt/FJSVejb/etc/FJEJBconfig.properties

追加定義

UserExSyslog=0

注2)

バージョンアップ前と同じように、イベントログファイル、またはシステムログファイルにメッセージを出力させたい場合は以下を設定してください。

定義ファイル

Windows

C:\¥Interstage¥J2EE¥etc¥isj2ee.properties

Solaris

Linux

/opt/FJSVj2ee/etc/isj2ee.properties

追加定義

syslog_output_mode=V6

注3)

SystemWalker製品を使用してログメッセージの監視を行っていたユーザがInterstage Application Server V7.0以降にバージョンアップした時に問題となる可能性があります。

バージョンアップ前と同じように、イベントログファイル、またはシステムログファイルにメッセージを出力させたい場合は以下を設定してください。

定義ファイル

Windows

C:\¥Interstage¥EJB¥etc¥FJEJBconfig.properties

Solaris

Linux

/opt/FJSVejb/etc/FJEJBconfig.properties

追加定義

MsgNoCompatible=V6

- FJEJBconfig.propertiesに無効な値を設定した場合は、デフォルト値(8.0)が設定されます。
- FJEJBconfig.propertiesを編集した場合は、変更した値を有効にするために、IJServerを再起動してください。

2.5.7 メッセージ番号がIJServer3で始まるメッセージ

ここでは、以下について説明します。メッセージの詳細は、“メッセージ集”を参照してください。

- [Interstage Application Server V9.0での変更](#)

■Interstage Application Server V9.0での変更

旧メッセージID	新メッセージID	変更内容
全般	—	日本語のエラーメッセージについて、エラーレベルが日本語から英語に変更になったものがあります。 また、メッセージのラベルやエラー種別の有無・形式が異なるものがあります。

2.5.8 メッセージ番号がisで始まるメッセージ

ここでは、以下について説明します。メッセージの詳細は、“メッセージ集”を参照してください。

- [Interstage Application Server V9.1での変更](#)
- [Interstage Application Server V9.0での変更](#)
- [Interstage Application Server 8.0での変更](#)
- [Interstage Application Server V7.0での変更](#)

■Interstage Application Server V9.1での変更

旧メッセージID	新メッセージID	変更内容
is30806	is31702 is31703 is31704 is31705 is31706 is31709 is31710 is31711 is31712	ワークユニットの操作時に発生するメッセージを、エラーの発生原因ごとに分けました。

旧メッセージID	新メッセージID	変更内容
	is31714 is31715 is31716 is31718	
is30807	is31702 is31703 is31704 is31705 is31706 is31709 is31710 is31711 is31712 is31714 is31715 is31716 is31718	ワークユニットの操作時に発生するメッセージを、エラーの発生原因ごとに分けました。

■Interstage Application Server V9.0での変更

旧メッセージID	新メッセージID	変更内容
is20711	is20711 is20786	Interstage JMXサービスの通信エラー発生時のメッセージから、Interstage JMXサービス起動時に発生するエラーのメッセージを分けました。

■Interstage Application Server 8.0での変更

旧メッセージID	新メッセージID	変更内容
is21156 is21166 is21201	変更なし	英語・日本語メッセージの内容を変更しました。

■Interstage Application Server V7.0での変更

旧メッセージID	新メッセージID	変更内容
is20711	is20711 is20725 is20726 is20727	Interstage JMXサービスの通信エラー発生時のメッセージを、通信エラーの原因によって分けました。
is20136 is20409 is20481 is30175	変更なし	英語メッセージ内容のみを変更しました。
is30896	変更なし	英語・日本語メッセージの内容を変更しました。
is40203	is40222	アプリケーションの配備解除処理が正常に終了した場合に出力されるメッセージを左記のように変更し、アプリケーション単位に出力されるように変更しました。

旧メッセージID	新メッセージID	変更内容
is40204	is40223	アプリケーションの配備解除処理に失敗した場合に出力するメッセージを左記のように変更し、アプリケーション単位に出力されるように変更しました。

2.5.9 メッセージ番号がisj2eeで始まるメッセージ

ここでは、以下について説明します。メッセージの詳細は、“メッセージ集”を参照してください。

- [Interstage Application Server V9.1での変更](#)

■Interstage Application Server V9.1での変更

旧メッセージID	新メッセージID	変更内容
isj2ee0215 isj2ee0257	変更なし	和文メッセージ本文内の“SID”を“SID/ネットサービス名”に変更しました。 英文メッセージ本文内の“SID”を“SID/Net Service Name”に変更しました。

2.5.10 メッセージ番号がjmsで始まるメッセージ

ここでは、以下について説明します。メッセージの詳細は、“メッセージ集”を参照してください。

- [Interstage Application Server V9.1での変更](#)
- [Interstage Application Server V9.0での変更](#)
- [Interstage Application Server 8.0での変更](#)

■Interstage Application Server V9.1での変更

旧メッセージID	新メッセージID	変更内容
jms6518	変更なし	メッセージの内容を変更しました。 旧:“Failed to get the enhanced security mode.” 新:“Failed to get the secure mode.”

■Interstage Application Server V9.0での変更

旧メッセージID	新メッセージID	変更内容
jms2050	変更なし	英語・日本語メッセージの内容を変更しました。 旧(英語):“Operation '%s' failed. A null name was passed.” 新(英語):“Operation '%s1' failed. The name ('%s2') is invalid.” 旧(日本語):“Operation '%s' が失敗しました。名前に null が渡されました。” 新(日本語):“Operation '%s1' が失敗しました。名前('%s2') に誤りがあります。”
jms6104	変更なし	可変情報に“Destination名”を追加しました。 旧(英語):“Refer to the local Host Naming Service.” 新(英語):“Refer to the local Host Naming Service. Destination '%s’” 旧(日本語):“ローカルホストのネーミングサービスを参照します。” 新(日本語):“ローカルホストのネーミングサービスを参照します。Destination '%s’”

■Interstage Application Server 8.0での変更

旧メッセージID	新メッセージID	変更内容
jms2563	jms8002	ノーティフィケーションサービスが使用できないイベントチャネルに対してJMSアプリケーションを動作させた場合に出力されるエラーメッセージを変更しました。

2.5.11 メッセージ番号がodで始まるメッセージ

ここでは、以下について説明します。メッセージの詳細は、“メッセージ集”を参照してください。

- [Interstage Application Server V9.0での変更](#)
- [Interstage Application Server V7.0での変更](#)
- [Interstage Application Server V6.0での変更](#)

■Interstage Application Server V9.0での変更

旧メッセージID	新メッセージID	変更内容
od10941	—	デフォルトでは出力されません。(注) サーバへのリクエスト送信がない状態がクライアントの無通信監視時間(configファイルのperiod_client_idle_con_timeout×5秒)を超えた場合、コネクションが切断されます。このとき、左記メッセージが出力されました。しかし、切断されたコネクションは、サーバへのリクエスト送信時に自動再接続されます。このため、クライアントの無通信監視時間超過によるコネクション切断について、ユーザが意識する必要はありません。

注)

メッセージを出力させたい場合はCORBAサービスの動作環境ファイル(config)のmsg_output_compatibleパラメタを設定してください。動作環境ファイル(config)については、“チューニングガイド”の“CORBAサービスの動作環境ファイル”の“config”を参照してください。

■Interstage Application Server V7.0での変更

旧メッセージID	新メッセージID	変更内容
od10719 od10912 od10914 od10922 od10923	変更なし	可変情報に、“OSから通知されたエラー番号”を追加しました。
od10925	変更なし	可変情報に、“タイムアウトの種別”と“リクエスト返信待機時間”を追加しました。

■Interstage Application Server V6.0での変更

旧メッセージID	新メッセージID	変更内容
od10916 od10925	変更なし	メッセージの出力内容を変更しました

旧メッセージID	新メッセージID	変更内容
od10924	—	デフォルトでは出力されません。(注) 以下の理由によりユーザが意識する必要はありません。 - 情報メッセージであり、多くの場合ユーザ対処が不要 - ユーザの対処が必要な場合、左記メッセージの前後に出力されたメッセージより、対処方法が特定可能
od10926	—	デフォルトでは出力されません。(注) クライアントからのリクエスト送信がない状態がサーバの無通信監視時間(configファイルのperiod_idle_con_timeout×5秒)を超えた場合、コネクションが切断されます。このとき、左記メッセージが出力されました。しかし、切断されたコネクションは、クライアントからのリクエスト送信時に自動再接続されます。このため、無通信監視時間超過によるコネクション切断について、ユーザが意識する必要はありません。
od11101	—	デフォルトでは出力されません。(注) queue_policyファイル(キュー制御機能のキューポリシーとして使用されるファイル)を設定しなかった場合、左記メッセージが出力されましたが、ユーザが意識する必要はありません。

注)

メッセージを出力させたい場合はCORBAサービスの動作環境ファイル(config)のmsg_output_compatibleパラメタを設定してください。動作環境ファイル(config)については、“チューニングガイド”の“CORBAサービスの動作環境ファイル”の“config”を参照してください。

2.5.12 メッセージ番号がSvMonで始まるメッセージ

ここでは、以下について説明します。メッセージの詳細は、“メッセージ集”を参照してください。

- [Interstage Application Server V9.0での変更](#)

■Interstage Application Server V9.0での変更

旧メッセージID	新メッセージID	変更内容
SvMon4002 SvMon4004 SvMon4201 SvMon4204	変更なし	可変情報に、Webサーバ名を追加しました。

2.5.13 EJBサービスのメッセージ

ここでは、以下について説明します。メッセージの詳細は、“メッセージ集”を参照してください。

- [Interstage Application Server 8.0での変更](#)
- [Interstage Application Server V6.0での変更](#)
- [Interstage Application Server V5.1での変更](#)
- [Interstage Application Server V5.0での変更](#)
- [旧メッセージを出力させるための設定](#)

■Interstage Application Server 8.0での変更

旧メッセージID	新メッセージID	変更内容
EJB3423	EJB3424	メッセージ改善

■Interstage Application Server V6.0での変更

旧メッセージID	新メッセージID	変更内容
EJB1079	IJServer21131	メッセージ改善
EJB1056	—	英語メッセージのスペルミス修正。 旧:“EJB: ERROR: EJB1056: The transaction attribute in this method is not specificate: NAME=%s1 METHOD =%s2” 新:“EJB: ERROR: EJB1056: The transaction attribute in this method is not specified: NAME=%s1 METHOD =%s2”
EJB1063	—	英語メッセージのスペルミス修正。 旧:“EJB: INFO: EJB1063: Status of the transaction maked rollback : NAME=%s1 METHOD =%s2” 新:“EJB: INFO: EJB1063: Status of the transaction marked rollback : NAME=%s1 METHOD =%s2”
EJB1094	—	英語メッセージのスペルミス修正。 旧:“EJB: ERROR: EJB1094: Session time-out is occured: NAME=%1 CODE=%2” 新:“EJB: ERROR: EJB1094: Session time-out is occurred: NAME=%1 CODE=%2”
EJB1096	—	英語メッセージのスペルミス修正。 旧:“EJB: ERROR: EJB1096: System Exception is occured during business method called: NAME=%1 METHOD=%2 CODE=%3” 新:“EJB: ERROR: EJB1096: System Exception is occurred during business method called: NAME=%1 METHOD=%2 CODE=%3”
EJB1249	—	英語メッセージのスペルミス修正。 旧:“EJB: ERROR: EJB1249: JTS Exception is occured when you get JDBCConnection: METHOD=%1 RESOURCE=%2” 新:“EJB: ERROR: EJB1249: JTS Exception is occurred when you get JDBCConnection: METHOD=%1 RESOURCE=%2”

■Interstage Application Server V5.1での変更

旧メッセージID	新メッセージID	変更内容
EJB1020	EJB1096	メッセージ改善
EJB1024	EJB1092 EJB1093	メッセージ改善
EJB1029	EJB1094	メッセージ改善
EJB1038	EJB1095	メッセージ改善
EJB1204	EJB1247 EJB1248 EJB1249	メッセージ改善

■Interstage Application Server V5.0での変更

旧メッセージID	新メッセージID	変更内容
EJB1023	EJB1081	メッセージ改善

■旧メッセージを出力させるための設定

以前のバージョン・レベルのメッセージを出力させたい場合は、以下を設定してください。

以下のファイルが存在しない場合や指定するキー名、または、値が違っている場合は、改善されたメッセージが出力されます。

定義ファイル

Windows

Interstageインストールディレクトリ¥ejb¥etc¥FJEJBconfig.properties

Solaris

Linux

/opt/FJSVejb/etc/FJEJBconfig.properties

指定するキー

MsgNoCompatible

指定する値

V4 :V4.1以前のメッセージを出力

V5 :V5で改善されたメッセージとV4.1以前のメッセージを出力

V51 :V5、V5.1で改善されたメッセージとV4.1以前のメッセージを出力

2.5.14 SSL環境設定コマンドのメッセージ

ここでは、以下について説明します。メッセージの詳細は、“メッセージ集”の“SSL環境設定コマンドのメッセージ”の“SMEEコマンドから出力されるメッセージ”を参照してください。

- [Interstage Application Server V7.0での変更](#)

■Interstage Application Server V7.0での変更

番号	“メッセージ集”修正内容	メッセージ本文修正内容
0011	—	メッセージの形式を統一(空白、記号、句読点など)
0013	メッセージの形式を統一(空白、記号、句読点など)	メッセージの形式を統一(空白、記号、句読点など) Windows メッセージ本文内の語句を修正
0014	メッセージの形式を統一(空白、記号、句読点など)	メッセージの形式を統一(空白、記号、句読点など)
0016	メッセージのスペルを修正	Solaris Linux メッセージのスペルを修正
0025	メッセージの形式を統一(空白、記号、句読点など)	メッセージの形式を統一(空白、記号、句読点など)
0029	—	Solaris Linux メッセージのスペルを修正
0032	メッセージ本文内の語句を修正	—

番号	“メッセージ集”修正内容	メッセージ本文修正内容
0033	メッセージの形式を統一(空白、記号、句読点など)	メッセージの形式を統一(空白、記号、句読点など)
0034	—	メッセージの形式を統一(空白、記号、句読点など)
0035	メッセージの形式を統一(空白、記号、句読点など)	メッセージの形式を統一(空白、記号、句読点など)
0040	メッセージのスペルを修正	メッセージのスペルを修正
0041	メッセージのスペルを修正	メッセージのスペルを修正
0042	メッセージのスペルを修正	メッセージのスペルを修正
0051	メッセージ本文内の語句を修正	—
0052	メッセージのスペルを修正	Solaris Linux メッセージのスペルを修正
0053	メッセージの形式を統一(空白、記号、句読点など)	—
0054	メッセージの形式を統一(空白、記号、句読点など)	メッセージの形式を統一(空白、記号、句読点など)
0055	メッセージの形式を統一(空白、記号、句読点など)	メッセージの形式を統一(空白、記号、句読点など)
0056	メッセージのスペルを修正	メッセージ本文内の語句を修正 Solaris Linux メッセージのスペルを修正
0065	メッセージの形式を統一(空白、記号、句読点など)	メッセージの形式を統一(空白、記号、句読点など)
0075	メッセージ本文内の語句を修正	—
0077	メッセージの形式を統一(空白、記号、句読点など)	メッセージ本文内の語句を修正
0078	メッセージのスペルを修正	メッセージのスペルを修正
0079	メッセージの形式を統一(空白、記号、句読点など)	メッセージの形式を統一(空白、記号、句読点など)
0080	メッセージのスペルを修正	—
0081	メッセージのスペルを修正	—
0082	メッセージのスペルを修正	—
0083	メッセージ本文内の語句を修正	Windows メッセージ本文内の語句を修正
0085	メッセージの形式を統一(空白、記号、句読点など)	メッセージの形式を統一(空白、記号、句読点など) Solaris Linux メッセージ本文内の語句を修正
0086	メッセージのスペルを修正	メッセージのスペルを修正
0088	メッセージの形式を統一(空白、記号、句読点など)	メッセージの形式を統一(空白、記号、句読点など)
0089	—	メッセージの形式を統一(空白、記号、句読点など)
0090	メッセージ本文内の語句を修正	—

番号	“メッセージ集”修正内容	メッセージ本文修正内容
0096	メッセージのスペルを修正	メッセージのスペルを修正
0098	—	Windows メッセージの形式を統一(空白、記号、句読点など)
0099	メッセージの形式を統一(空白、記号、句読点など)	—
0101	—	メッセージの形式を統一(空白、記号、句読点など)
0105	メッセージの形式を統一(空白、記号、句読点など)	メッセージの形式を統一(空白、記号、句読点など)
0110	—	メッセージの形式を統一(空白、記号、句読点など)
0111	—	メッセージの形式を統一(空白、記号、句読点など) メッセージ本文内の語句を修正
0113	—	メッセージの形式を統一(空白、記号、句読点など)
0115	メッセージの形式を統一(空白、記号、句読点など)	メッセージの形式を統一(空白、記号、句読点など)
0116	—	メッセージの形式を統一(空白、記号、句読点など)
0121	—	メッセージの形式を統一(空白、記号、句読点など)
0200	メッセージのスペルを修正	メッセージのスペルを修正
0201	メッセージのスペルを修正	メッセージのスペルを修正
0206	—	メッセージの形式を統一(空白、記号、句読点など)
0207	メッセージのスペルを修正	メッセージのスペルを修正
0209	メッセージの形式を統一(空白、記号、句読点など)	メッセージの形式を統一(空白、記号、句読点など)
0317	—	メッセージのスペルを修正
0318	メッセージ本文を修正	—
0520	—	メッセージの形式を統一(空白、記号、句読点など)
0521	—	メッセージの形式を統一(空白、記号、句読点など)
0522	メッセージ本文内の語句を修正	メッセージの形式を統一(空白、記号、句読点など)
0523	メッセージ本文内の語句を修正	メッセージの形式を統一(空白、記号、句読点など)
0524	—	メッセージの形式を統一(空白、記号、句読点など)
0525	—	メッセージの形式を統一(空白、記号、句読点など)

2.5.15 CORBAサービスのマイナーコード

ここでは、以下について説明します。メッセージの詳細は、“メッセージ集”を参照してください。

- [Interstage Application Server 8.0での変更](#)
- [Interstage Application Server 8.0/V7.0での変更](#)
- [Interstage Application Server V6.0での変更](#)

■Interstage Application Server 8.0での変更 **Solaris**

Interstage Application Server 8.0以降では、CORBA_ORB_initの引数に指定した拡張システムの情報取得に失敗した場合、以下のマイナーコードが出力されます。

マイナーコード		意味
16進表記	10進表記	
0x464a0125	1179255077	指定された拡張システムの情報取得に失敗しました。
0x464a0925	1179257125	

■Interstage Application Server 8.0/V7.0での変更

[RHEL-AS4\(IPF\)](#)の場合はInterstage Application Server V7.0以降、その他のプラットフォームではInterstage Application Server 8.0以降では、odcntlqueコマンドでキューイングされたリクエストを取り消した場合に、以下のマイナーコードが通知されます。

マイナーコード		意味
16進表記	10進表記	
0x464a0095	1179254933	キュー上に存在するリクエストがキャンセルされました。
0x464a0895	1179256981	

■Interstage Application Server V6.0での変更

configパラメタmax_IIOp_req_per_con廃止に伴い、Interstage Application Server V6.0以降では以下のマイナーコードが出力されなくなりました。

マイナーコード		意味
16進表記	10進表記	
0x464a0103	1179255043	“max_IIOp_req_per_con”を超えました。
0x464a0903	1179257091	

2.5.16 Interstage シングル・サインオンのメッセージ

メッセージ、およびステータスコードのテキストの改善により、旧バージョン・レベルで出力されていたメッセージ、およびステータスコードのテキストと、出力内容が異なる場合があります。ここでは、以下について説明します。

- [旧メッセージを出力させるための設定](#)
- [Webブラウザに表示するメッセージのステータスコードの変更](#)

改善されたメッセージ、およびステータスコードについては、“シングル・サインオン運用ガイド”の“旧バージョンの環境定義と機能について”を参照してください。

■旧メッセージを出力させるための設定

Interstage Application Server V7.0以前のバージョン・レベルで出力されていたメッセージを出力させるには、環境定義ファイルに以下の定義項目を追加します。



ポイント

旧バージョン・レベルから移行した環境においては、移行時に自動的に以下の設定が行われます。新規メッセージを出力させる場合は、以下の設定を削除してください。

リポジトリサーバ

定義ファイル

Windows

C:\Interstage\F3FMssosv\conf\ssoatcsv.conf

Solaris

Linux

/etc/opt/FJSVssosv/conf/ssoatcsv.conf

追加定義

use-old-syslog-sso01050=YES

use-old-syslog-sso01051=YES

認証サーバ

定義ファイル

Windows

C:\Interstage\F3FMssocag\conf\ssoatcag.conf

Solaris

Linux

/etc/opt/FJSVssocag/conf/ssoatcag.conf

追加定義

use-old-syslog-sso02012=YES

use-old-syslog-sso02013=YES

use-old-syslog-sso02015=YES

use-old-syslog-sso02031=YES

use-old-syslog-sso02050=YES

■Webブラウザに表示するメッセージのステータスコードの変更

Interstage Application Server 8.0以前のバージョン・レベルで出力されていた以下のWebブラウザに表示するメッセージがWebブラウザに返却するHTTPステータスコードを403から200に変更し、ファイル名を変更しました。



ポイント

- 旧バージョン・レベルから移行した環境においては、従来どおりステータスコード403で返却します。ステータスコード200で返却するよう変更する場合は、以下の新メッセージファイルをいったん削除し、旧メッセージファイル名を新メッセージファイル名に変更してください。
- 8.0から移行した業務サーバで、WebサーバにMicrosoft(R) Internet Information Services 7.0を使用する場合は、必ず以下に示す業務サーバの新メッセージファイルをいったん削除し、旧メッセージファイル名を新メッセージファイル名に変更してください。

【認証サーバ】

旧メッセージファイル名	新メッセージファイル名	変更内容
403auth_form_en.template	200auth_form_en.template	ステータスコード200で返却します。
403auth_form_ja.template	200auth_form_ja.template	
403passwder_err_form_en.template	200passwder_err_form_en.template	
403passwder_err_form_ja.template	200passwder_err_form_ja.template	
403authexpired_en.template	200authexpired_en.template	
403authexpired_ja.template	200authexpired_ja.template	
403timedout_en.template	200timedout_en.template	
403timedout_ja.template	200timedout_ja.template	
403queryforcedsignon_en.template	200queryforcedsignon_en.template	
403queryforcedsignon_ja.template	200queryforcedsignon_ja.template	
403querysignoff_en.template	200querysignoff_en.template	
403querysignoff_ja.template	200querysignoff_ja.template	

【業務サーバ】

旧メッセージファイル名	新メッセージファイル名	変更内容
403postauth_en.template	200postauth_en.template	ステータスコード200で返却します。
403postauth_ja.template	200postauth_ja.template	
403closeerr_en.template	200closeerr_en.template	
403closeerr_ja.template	200closeerr_ja.template	

2.5.17 Interstage ディレクトリサービスのメッセージ

ここでは、以下について説明します。メッセージの詳細は、“メッセージ集”を参照してください。

- [Interstage Application Server V9.0での変更](#)
- [Interstage Application Server 8.0での変更](#)
- [LDAPコマンドのメッセージ変更](#)

■Interstage Application Server V9.0での変更

旧メッセージID	新メッセージID	変更内容
irep10018 irep10020 irep10023 irep10024 irep10705 irep10900 irep10901 irep10981 irep10982 irep10983 irep10984	変更なし	ディレクトリサービスの名称変更に伴い、名称を含んでいる英語・日本語メッセージの内容を変更しました。

旧メッセージID	新メッセージID	変更内容
irep10985 irep11043 irep11207 irep14100 irep14101 irep14102 irep14103 irep14104 irep70100 irep70502 irep70505 irep78027		

■Interstage Application Server 8.0での変更

旧メッセージID	新メッセージID	変更内容
メッセージ番号が irepで始まるすべてのメッセージ	変更なし	日本語メッセージの最後に、句点(.)をつけるように統一しました。
irep10014 irep10020 irep10023 irep10024 irep10034 irep10171 irep10313 irep10321 irep10709 irep10906 irep10960 irep10981 irep10983 irep10987 irep10990 irep11012 irep11051 irep11208 irep66087 irep78028	変更なし	英語・日本語メッセージの内容を変更しました。

■LDAPコマンドのメッセージ変更

Interstage Application Server 8.0のInterstage ディレクトリサービスのSDKでは、LDAPの最新の規約に正しく準拠するため、メッセージを変更/追加しました。また、InfoDirectory SDKのLDAPコマンドで出力されていたメッセージラベルのない形式(“メッセージ集”の“Interstage ディレクトリサービスが出力するメッセージ”を参照)から、メッセージラベルのある形式に変更しました。メッセージ番号は、irep14000番台です。



メッセージ変更例

例

変更前)

ldap_search : Bad search filter

変更後)

IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。ldap_search_ext: Bad search filter (87)

同様に、irepencupinコマンドから出力されるメッセージも、メッセージラベルのない形式から、メッセージラベルのある形式に変更しました。

具体的な変更内容は、以下に記載しています。

- [LDAPコマンドの変更したメッセージ](#)
- [LDAPコマンドの廃止したメッセージ](#)
- [irepencupinコマンドの変更したメッセージ](#)

◆LDAPコマンドの変更したメッセージ

LDAPコマンドの変更したメッセージを以下に示します。

旧メッセージの先頭文字

[A] [B] [C] [D] [E] [F]
[I] [L] [N] [O] [P] [R]
[S] [T] [U] [V]
[可変情報]

[A]で始まる旧メッセージ

- [Admin limit exceeded](#)
- [Affects multiple DSAs](#)
- [Already exists](#)
- [A secret key doesn't exist %s1 \(%s2 \)](#)

Admin limit exceeded

[新メッセージ]

IREP: ERROR: irep14210: ldapsearch: Search failed. %s1: %s2 (%s3)

IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。 %s1: %s2 (%s3)

Affects multiple DSAs

[新メッセージ]

IREP: ERROR: irep14207: ldapmodify: Failed to modify the entry. %s1
%s2: %s3 (%s4)

%s5

%s6

IREP: エラー: irep14207: ldapmodify: エントリの更新に失敗しました。 %s1

%s2: %s3 (%s4)

%s5

%s6

Already exists

[新メッセージ]

以下のどれかになります。

IREP: ERROR: irep14208: ldapmodify: Failed to delete the entry. %s1
ldap_delete: %s2 (%s3)

IREP: エラー: irep14208: ldapmodify: エントリの削除に失敗しました。 %s1

ldap_delete: %s2 (%s3)

IREP: ERROR: irep14209: ldapmodify: Failed to modify the RDN. %s1

ldap_modrdn: %s2 (%s3)

IREP: エラー: irep14209: ldapmodify: エントリの識別名変更に失敗しました。%s1
ldap_modrdn: %s2 (%s3)

A secret key doesn't exist %s1 (%s2)

[新メッセージ]

IREP: ERROR: irep14182: The private key for the Site certificate is not in the certificate environment. %s1 (%s2)

IREP: エラー: irep14182: サイト証明書に対応する秘密鍵が証明書環境にありません。%s1 (%s2)

[B]で始まる旧メッセージ

- [Bad search filter](#)
- [Because encountered an incomplete character, the iconv stopped converting.\(EBADE\)](#)

Bad search filter

[新メッセージ]

IREP: ERROR: irep14210: ldapsearch: Search failed. %s1: %s2 (%s3)

IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。%s1: %s2 (%s3)

Because encountered an incomplete character, the iconv stopped converting.(EBADE)

[新メッセージ]

IREP: ERROR: irep14305: %s1: The character code in the specified entry information is incorrect. (EBADE)
%s2: %s3 (%s4).

%s5

%s6

IREP: エラー: irep14305: %s1: 指定したエントリ情報内の文字コードに誤りがあります。(EBADE)

%s2: %s3 (%s4).

%s5

%s6

[C]で始まる旧メッセージ

- [Can't connect to the LDAP server](#)
- [Can't contact LDAP server](#)
- [Cannot modify object class](#)
- [Constraint violation](#)
- [Control not found](#)
- [Could not open SSL env file %s1 - absolute path?](#)

Can't connect to the LDAP server

[新メッセージ]

IREP: ERROR: irep14207: ldapmodify: Failed to modify the entry. %s1

%s2: %s3 (%s4)

%s5

%s6

IREP: エラー: irep14207: ldapmodify: エントリの更新に失敗しました。%s1

%s2: %s3 (%s4)

%s5

%s6

Can't contact LDAP server

[新メッセージ]

IREP: ERROR: irep14208: ldapmodify: Failed to delete the entry. %s1

ldap_delete: %s2 (%s3)

IREP: エラー: irep14208: ldapmodify: エントリの削除に失敗しました。%s1

ldap_delete: %s2 (%s3)

Cannot modify object class

[新メッセージ]

IREP: ERROR: irep14207: ldapmodify: Failed to modify the entry. %s1
%s2: %s3 (%s4)
%s5
%s6
IREP: エラー: irep14207: ldapmodify: エントリの更新に失敗しました。 %s1
%s2: %s3 (%s4)
%s5
%s6

Constraint violation

[新メッセージ]

以下のどれかになります。

IREP: ERROR: irep14208: ldapmodify: Failed to delete the entry. %s1
ldap_delete: %s2 (%s3)
IREP: エラー: irep14208: ldapmodify: エントリの削除に失敗しました。 %s1
ldap_delete: %s2 (%s3)

IREP: ERROR: irep14209: ldapmodify: Failed to modify the RDN. %s1
ldap_modrdn: %s2 (%s3)
IREP: エラー: irep14209: ldapmodify: エントリの識別名変更に失敗しました。 %s1
ldap_modrdn: %s2 (%s3)

Control not found

[新メッセージ]

以下のどれかになります。

IREP: ERROR: irep14209: ldapmodify: Failed to modify the RDN. %s1
ldap_modrdn: %s2 (%s3)
IREP: エラー: irep14209: ldapmodify: エントリの識別名変更に失敗しました。 %s1
ldap_modrdn: %s2 (%s3)

IREP: ERROR: irep14210: ldapsearch: Search failed. %s1: %s2 (%s3)
IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。 %s1: %s2 (%s3)

Could not open SSL env file %s1 - absolute path?

[新メッセージ]

IREP: ERROR: irep14150: Failed to open the specified SSL environment definition file. %s1
IREP: エラー: irep14150: 指定したSSL環境定義ファイル%s1をオープンできませんでした。

[D]で始まる旧メッセージ

- [Decoding error](#)
- [DSA is busy](#)
- [DSA is unavailable](#)
- [DSA is unwilling to perform](#)

Decoding error

[新メッセージ]

IREP: ERROR: irep14207: ldapmodify: Failed to modify the entry. %s1
%s2: %s3 (%s4)
%s5
%s6

IREP: エラー: irep14207: ldapmodify: エントリの更新に失敗しました。%s1
%s2: %s3 (%s4)
%s5
%s6

DSA is busy

[新メッセージ]

IREP: ERROR: irep14208: ldapmodify: Failed to delete the entry. %s1
ldap_delete: %s2 (%s3)
IREP: エラー: irep14208: ldapmodify: エントリの削除に失敗しました。%s1
ldap_delete: %s2 (%s3)

DSA is unavailable

[新メッセージ]

IREP: ERROR: irep14209: ldapmodify: Failed to modify the RDN. %s1
ldap_modrdn: %s2 (%s3)
IREP: エラー: irep14209: ldapmodify: エントリの識別名 変更に失敗しました。%s1
ldap_modrdn: %s2 (%s3)

DSA is unwilling to perform

[新メッセージ]

以下のどれかになります。

IREP: ERROR: irep14023: ldapsearch: Specify an essential option. %s1
IREP: エラー: irep14023: ldapsearch: 必須オプション%s1を指定してください。

IREP: ERROR: irep14204: Failed to bind to the repository. %s1 (%s2)
%s3
%s4
IREP: エラー: irep14204: リポジトリへの認証(bind)に失敗しました。%s1 (%s2)
%s3
%s4

IREP: ERROR: irep14210: ldapsearch: Search failed. %s1: %s2 (%s3)
IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。%s1: %s2 (%s3)

IREP: ERROR: irep14211: ldapsearch: Search failed. %s1 (%s2)
%s3
%s4
IREP: エラー: irep14211: ldapsearch: 検索に失敗しました。%s1 (%s2)
%s3
%s4

[E]で始まる旧メッセージ

- [Encoding error](#)
- [Encountered an incomplete character or shift sequence at the end of the input buffer.\(EINVAL\)](#)
- [Encountered an input byte not belonging to the input codeset.\(EILSEQ\)](#)

Encoding error

[新メッセージ]

IREP: ERROR: irep14210: ldapsearch: Search failed. %s1: %s2 (%s3)
IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。%s1: %s2 (%s3)

Encountered an incomplete character or shift sequence at the end of the input buffer.(EINVAL)

[新メッセージ]

IREP: ERROR: irep14300: %s1: The specified entry information is incorrect. (EINVAL)
%s2: %s3 (%s4).
%s5
%s6
IREP: エラー: irep14300: %s1: 指定したエントリ情報に誤りがあります。(EINVAL)
%s2: %s3 (%s4).
%s5
%s6

Encountered an input byte not belonging to the input codeset.(EILSEQ)

[新メッセージ]

IREP: ERROR: irep14302: %s1: The character code in the specified entry information is incorrect. (EILSEQ)
%s2: %s3 (%s4).
%s5
%s6
IREP: エラー: irep14302: %s1: 指定したエントリ情報内の文字コードに誤りがあります。(EILSEQ)
%s2: %s3 (%s4).
%s5
%s6

[F]で始まる旧メッセージ

Failed to convert character code(%s1)

[新メッセージ]

IREP: ERROR: irep14308: %s1: Failed to convert the character code. (%s2)
%s3: %s4 (%s5).
%s6
%s7
IREP: エラー: irep14308: %s1: 文字コード変換に失敗しました。(%s2)
%s3: %s4 (%s5).
%s6
%s7

[I]で始まる旧メッセージ

- [illegal option -- %s1](#)
- [Inappropriate authentication](#)
- [Inappropriate matching](#)
- [incorrect ssl environment value](#)
- [Insufficient access](#)
- [Insufficient space in the buffer.\(ENOMEM\)](#)
- [Insufficient space in the output buffer.\(E2BIG\)](#)
- [Invalid credentials](#)
- [Invalid DN syntax](#)
- [Invalid syntax](#)

illegal option -- %s1

[新メッセージ]

IREP: ERROR: irep14000: %s1: Unknown option. "%s2"
IREP: エラー: irep14000: %s1: 指定したオプション %s2 は無効です。

Inappropriate authentication

[新メッセージ]

IREP: ERROR: irep14207: ldapmodify: Failed to modify the entry. %s1
%s2: %s3 (%s4)
%s5
%s6
IREP: エラー: irep14207: ldapmodify: エントリの更新に失敗しました。 %s1
%s2: %s3 (%s4)
%s5
%s6

Inappropriate matching

[新メッセージ]

以下のどれかになります。

IREP: ERROR: irep14208: ldapmodify: Failed to delete the entry. %s1
ldap_delete: %s2 (%s3)
IREP: エラー: irep14208: ldapmodify: エントリの削除に失敗しました。 %s1
ldap_delete: %s2 (%s3)

IREP: ERROR: irep14209: ldapmodify: Failed to modify the RDN. %s1
ldap_modrdn: %s2 (%s3)
IREP: エラー: irep14209: ldapmodify: エントリの識別名変更に失敗しました。 %s1
ldap_modrdn: %s2 (%s3)

IREP: ERROR: irep14210: ldapsearch: Search failed. %s1: %s2 (%s3)
IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。 %s1: %s2 (%s3)

incorrect ssl environment value

[新メッセージ]

以下のどれかになります。

IREP: ERROR: irep14151: %s1: line %s2: The description format of SSL environment definition file is incorrect. The contents will be ignored.
IREP: エラー: irep14151: SSL環境定義ファイル%s1の%s2行目の記述形式に誤りがあります。無視しました。

IREP: ERROR: irep14152: %s1: line %s2: The value for the configuration option in SSL environment definition file is incorrect.
IREP: エラー: irep14152: SSL環境定義ファイル%s1の%s2行目の定義項目の値に誤りがあります。

IREP: ERROR: irep14153: %s1: line %s2: The value for configuration option %s3 in SSL environment definition file is incorrect.
IREP: エラー: irep14153: SSL環境定義ファイル%s1の%s2行目の定義項目 %s3の値に誤りがあります。

IREP: ERROR: irep14154: %s1: line %s2: The directory path specified for configuration option %s3 in SSL environment definition file is incorrect.
IREP: エラー: irep14154: SSL環境定義ファイル%s1の%s2行目の定義項目 %s3に指定したディレクトリのパスに誤りがあります。

IREP: ERROR: irep14155: %s1: line %s2: The user PIN specified for configuration option tkn_pwd in SSL environment definition file is incorrect.
IREP: エラー: irep14155: SSL環境定義ファイル%s1の%s2行目の定義項目 tkn_pwd に指定したユーザPINに誤りがあります。

IREP: ERROR: irep14156: %s1: line %s2: Configuration option "%s3" in SSL environment definition file is invalid.
IREP: エラー: irep14156: SSL環境定義ファイル%s1の%s2行目の定義項目 "%s3"は無効です。

IREP: ERROR: irep14157: The description format of the SSL environment definition file is incorrect.
IREP: エラー: irep14157: SSL環境定義ファイルの記述形式に誤りがあります。

IREP: ERROR: irep14158: The user PIN specified in the SSL environment definition file has been changed by mistake.
IREP: エラー: irep14158: SSL環境定義ファイルに指定したユーザPINが誤って変更されています。

IREP: ERROR: irep14159: Specify an essential configuration option for SSL environment definition file. %s1
IREP: エラー: irep14159: SSL環境定義ファイル%s1に、必須の定義項目を指定してください。

Insufficient access

[新メッセージ]

以下のどれかになります。

IREP: ERROR: irep14207: ldapmodify: Failed to modify the entry. %s1
%s2: %s3 (%s4)
%s5
%s6
IREP: エラー: irep14207: ldapmodify: エントリの更新に失敗しました。 %s1
%s2: %s3 (%s4)
%s5
%s6

IREP: ERROR: irep14208: ldapmodify: Failed to delete the entry. %s1
ldap_delete: %s2 (%s3)
IREP: エラー: irep14208: ldapmodify: エントリの削除に失敗しました。 %s1
ldap_delete: %s2 (%s3)

IREP: ERROR: irep14209: ldapmodify: Failed to modify the RDN. %s1
ldap_modrdn: %s2 (%s3)
IREP: エラー: irep14209: ldapmodify: エントリの識別名変更に失敗しました。 %s1
ldap_modrdn: %s2 (%s3)

IREP: ERROR: irep14210: ldapsearch: Search failed. %s1: %s2 (%s3)
IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。 %s1: %s2 (%s3)

Insufficient space in the buffer.(ENOMEM)

[新メッセージ]

IREP: ERROR: irep14301: %s1: There is not enough memory available for character code conversion. (ENOMEM)
%s2: %s3 (%s4).
%s5
%s6
IREP: エラー: irep14301: %s1: 文字コードの変換でメモリ不足が発生しました。(ENOMEM)
%s2: %s3 (%s4).
%s5
%s6

Insufficient space in the output buffer.(E2BIG)

[新メッセージ]

IREP: ERROR: irep14303: %s1: There is not enough memory available for character code conversion. (E2BIG)
%s2: %s3 (%s4).
%s5
%s6
IREP: エラー: irep14303: %s1: 文字コードの変換でメモリ不足が発生しました。(E2BIG)
%s2: %s3 (%s4).
%s5
%s6

Invalid credentials

[新メッセージ]

以下のどちらかになります。

```
IREP: ERROR: irep14204: Failed to bind to the repository. %s1 (%s2)
%s3
%s4
IREP: エラー: irep14204: リポジトリへの認証(bind)に失敗しました。 %s1 (%s2)
%s3
%s4
```

```
IREP: ERROR: irep14207: ldapmodify: Failed to modify the entry. %s1
%s2: %s3 (%s4)
%s5
%s6
IREP: エラー: irep14207: ldapmodify: エントリの更新に失敗しました。 %s1
%s2: %s3 (%s4)
%s5
%s6
```

Invalid DN syntax

[新メッセージ]

```
IREP: ERROR: irep14208: ldapmodify: Failed to delete the entry. %s1
ldap_delete: %s2 (%s3)
IREP: エラー: irep14208: ldapmodify: エントリの削除に失敗しました。 %s1
ldap_delete: %s2 (%s3)
```

Invalid syntax

[新メッセージ]

以下のどちらかになります。

```
IREP: ERROR: irep14209: ldapmodify: Failed to modify the RDN. %s1
ldap_modrdn: %s2 (%s3)
IREP: エラー: irep14209: ldapmodify: エントリの識別名変更に失敗しました。 %s1
ldap_modrdn: %s2 (%s3)

IREP: ERROR: irep14210: ldapsearch: Search failed. %s1: %s2 (%s3)
IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。 %s1: %s2 (%s3)
```

[L]で始まる旧メッセージ

- [ldap_init failed](#)
- [ldap_parse_result: Bad parameter to an ldap routine](#)
- [ldap_parse_result failed %s1 : %s2](#)
- [ldap_search_ext_s returned : %s1](#)
- [ldap_search timed out](#)
- [ldapsl_init failed](#)
- [Local error](#)
- [ldmv3: expecting "%s1" but saw "%s2" \(line %s3 of entry %s4\)](#)
- [ldmv3: extra lines at end \(line %s1 of entry %s2\)](#)
- [ldmv3: invalid ldif format \(line %s1 of entry: %s2\)](#)
- [ldmv3: no attributes to change or add \(entry %s1\)](#)

- [ldmv3: unknown changetype "%s1" \(line %s2 of entry %s3\)](#)

ldap_init failed

[新メッセージ]

IREP: ERROR: irep14200: %s1: Failed to open the session.

IREP: エラー: irep14200: %s1: セッションのオープンに失敗しました。

ldap_parse_result: Bad parameter to an ldap routine

[新メッセージ]

以下のどれかになります。

IREP: ERROR: irep14205: ldapdelete: Failed to delete the entry. %s1: %s2 (%s3)

%s4

%s5

IREP: エラー: irep14205: ldapdelete: エントリの削除に失敗しました。 %s1: %s2 (%s3)

%s4

%s5

IREP: ERROR: irep14206: ldapdelete: Failed to delete the entry. %s1 (%s2)

%s3

%s4

IREP: エラー: irep14206: ldapdelete: エントリの削除に失敗しました。 %s1: (%s2)

%s3

%s4

IREP: ERROR: irep14207: ldapmodify: Failed to modify the entry. %s1

%s2: %s3 (%s4)

%s5

%s6

IREP: エラー: irep14207: ldapmodify: エントリの更新に失敗しました。 %s1

%s2: %s3 (%s4)

%s5

%s6

IREP: ERROR: irep14208: ldapmodify: Failed to delete the entry. %s1

ldap_delete: %s2 (%s3)

IREP: エラー: irep14208: ldapmodify: エントリの削除に失敗しました。 %s1

ldap_delete: %s2 (%s3)

IREP: ERROR: irep14209: ldapmodify: Failed to modify the RDN. %s1

ldap_modrdn: %s2 (%s3)

IREP: エラー: irep14209: ldapmodify: エントリの識別名変更に失敗しました。 %s1

ldap_modrdn: %s2 (%s3)

IREP: ERROR: irep14210: ldapsearch: Search failed. %s1: %s2 (%s3)

IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。 %s1: %s2 (%s3)

ldap_parse_result failed %s1 : %s2

[新メッセージ]

IREP: ERROR: irep14210: ldapsearch: Search failed. %s1: %s2 (%s3)

IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。 %s1: %s2 (%s3)

ldap_search_ext_s returned : %s1

[新メッセージ]

IREP: ERROR: irep14210: ldapsearch: Search failed. %s1: %s2 (%s3)

IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。 %s1: %s2 (%s3)

ldap_search timed out

[新メッセージ]

IREP: ERROR: irep14210: ldapsearch: Search failed. %s1: %s2 (%s3)

IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。 %s1: %s2 (%s3)

ldapsl_init failed

[新メッセージ]

以下のどちらかになります。

IREP: ERROR: irep14159: Specify an essential configuration option for SSL environment definition file. %s1

IREP: エラー: irep14159: SSL環境定義ファイル%s1に、必須の定義項目を指定してください。

IREP: ERROR: irep14201: %s1: Failed to open the SSL session.

IREP: エラー: irep14201: %s1: SSLを使用したセッションのオープンに失敗しました。

Local error

[新メッセージ]

以下のどれかになります。

IREP: ERROR: irep14207: ldapmodify: Failed to modify the entry. %s1

%s2: %s3 (%s4)

%s5

%s6

IREP: エラー: irep14207: ldapmodify: エントリの更新に失敗しました。 %s1

%s2: %s3 (%s4)

%s5

%s6

IREP: ERROR: irep14208: ldapmodify: Failed to delete the entry. %s1

ldap_delete: %s2 (%s3)

IREP: エラー: irep14208: ldapmodify: エントリの削除に失敗しました。 %s1

ldap_delete: %s2 (%s3)

IREP: ERROR: irep14209: ldapmodify: Failed to modify the RDN. %s1

ldap_modrdn: %s2 (%s3)

IREP: エラー: irep14209: ldapmodify: エントリの識別名変更に失敗しました。 %s1

ldap_modrdn: %s2 (%s3)

IREP: ERROR: irep14210: ldapsearch: Search failed. %s1: %s2 (%s3)

IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。 %s1: %s2 (%s3)

ldmv3: expecting "%s1" but saw "%s2" (line %s3 of entry %s4)

[新メッセージ]

IREP: ERROR: irep14015: ldapmodify: Expecting "%s1:" for modrdn, but saw "%s2:" . (line %s3 entry:"%s4")

IREP: エラー: irep14015: ldapmodify: 識別名変更の指定項目 "%s1:"がなく、"%s2:"になっています。(%s3行目 エントリ:"%s4")

ldmv3: extra lines at end (line %s1 of entry %s2)

[新メッセージ]

IREP: ERROR: irep14016: ldapmodify: The modrdn specification is incorrect. (line %s1 entry:"%s2").

IREP: エラー: irep14016: ldapmodify: 識別名変更の指定に誤りがあります。(%s1行目 エントリ:"%s2")

ldmv3: invalid ldif format (line %s1 of entry: %s2)

[新メッセージ]

IREP: ERROR: irep14010: ldapmodify: The LDIF description format is incorrect. (line %s1 entry: "%s2")

Error: %s3 (%s4).

IREP: エラー: irep14010: ldapmodify: LDIFの記述形式に誤りがあります。(%s1行目 エントリ:"%s2")
Error: %s3 (%s4).

ldmv3: no attributes to change or add (entry %s1)

[新メッセージ]

IREP: ERROR: irep14017: ldapmodify: Specify the attribute to be changed or added. (entry= "%s1").
IREP: エラー: irep14017: ldapmodify: 変更、または追加する属性を指定してください。(エントリ="%s1")

ldmv3: unknown changetype "%s1" (line %s2 of entry %s3)

[新メッセージ]

IREP: ERROR: irep14013: ldapmodify: changetype "%s1" is incorrect. (line %s2 entry:"%s3")
IREP: エラー: irep14013: ldapmodify: changetype "%s1" の指定に誤りがあります。(%s2行目 エントリ:"%s3")

[N]で始まる旧メッセージ

- [Naming violation](#)
- [No such attribute](#)
- [No such object](#)
- [Not compliated exchange](#)
- [Not enough memory](#)
- [Not enough memory by SSL %s1 \(%s2 \)](#)
- [Not enough stream by SSL %s1 \(%s2 \)](#)
- [Not support code](#)

Naming violation

[新メッセージ]

IREP: ERROR: irep14207: ldapmodify: Failed to modify the entry. %s1
%s2: %s3 (%s4)
%s5
%s6
IREP: エラー: irep14207: ldapmodify: エントリの更新に失敗しました。 %s1
%s2: %s3 (%s4)
%s5
%s6

No such attribute

[新メッセージ]

IREP: ERROR: irep14207: ldapmodify: Failed to modify the entry. %s1
%s2: %s3 (%s4)
%s5
%s6
IREP: エラー: irep14207: ldapmodify: エントリの更新に失敗しました。 %s1
%s2: %s3 (%s4)
%s5
%s6

No such object

[新メッセージ]

以下のどれかになります。

IREP: ERROR: irep14206: ldapdelete: Failed to delete the entry. %s1 (%s2)
%s3
%s4

IREP: エラー: irep14206: ldapdelete: エントリの削除に失敗しました。%s1: (%s2)
%s3
%s4

IREP: ERROR: irep14208: ldapmodify: Failed to delete the entry. %s1
ldap_delete: %s2 (%s3)
IREP: エラー: irep14208: ldapmodify: エントリの削除に失敗しました。%s1
ldap_delete: %s2 (%s3)

IREP: ERROR: irep14209: ldapmodify: Failed to modify the RDN. %s1
ldap_modrdn: %s2 (%s3)
IREP: エラー: irep14209: ldapmodify: エントリの識別名変更に失敗しました。%s1
ldap_modrdn: %s2 (%s3)

IREP: ERROR: irep14210: ldapsearch: Search failed. %s1: %s2 (%s3)
IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。%s1: %s2 (%s3)

IREP: ERROR: irep14211: ldapsearch: Search failed. %s1 (%s2)
%s3
%s4
IREP: エラー: irep14211: ldapsearch: 検索に失敗しました。%s1 (%s2)
%s3
%s4

Not compliated exchange

[新メッセージ]

IREP: ERROR: irep14308: %s1: Failed to convert the character code. (%s2)
%s3: %s4 (%s5).
%s6
%s7
IREP: エラー: irep14308: %s1: 文字コード変換に失敗しました。 (%s2)
%s3: %s4 (%s5).
%s6
%s7

Not enough memory

[新メッセージ]

IREP: ERROR: irep14105: %s1: %s2: There is not enough memory available.
%s3: %s4 (%s5).
%s6
%s7
IREP: エラー: irep14105: %s1: %s2: メモリ不足が発生しました。
%s3: %s4 (%s5).
%s6
%s7

Not enough memory by SSL %s1 (%s2)

[新メッセージ]

IREP: ERROR: irep14171: There is not enough memory available for SSL. %s1 (%s2)
IREP: エラー: irep14171: SSL通信でメモリ不足が発生しました。%s1 (%s2)

Not enough stream by SSL %s1 (%s2)

[新メッセージ]

IREP: ERROR: irep14172: There is not enough memory available for SSL. %s1 (%s2)
IREP: エラー: irep14172: SSL通信でメモリ不足が発生しました。%s1 (%s2)

Not support code

[新メッセージ]

IREP: ERROR: irep14004: %s1: The character code specified in the -G option is not supported.
IREP: エラー: irep14004: %s1: -Gオプションで指定した文字コードはサポートしていません。

[O]で始まる旧メッセージ

- [Object class violation](#)
- [Object is a leaf](#)
- [Operation not allowed on nonleaf](#)
- [Operation not allowed on RDN](#)
- [Operations error](#)
- [Out of memory](#)

Object class violation

[新メッセージ]

IREP: ERROR: irep14208: ldapmodify: Failed to delete the entry. %s1
ldap_delete: %s2 (%s3)
IREP: エラー: irep14208: ldapmodify: エントリの削除に失敗しました。 %s1
ldap_delete: %s2 (%s3)

Object is a leaf

[新メッセージ]

IREP: ERROR: irep14207: ldapmodify: Failed to modify the entry. %s1
%s2: %s3 (%s4)
%s5
%s6
IREP: エラー: irep14207: ldapmodify: エントリの更新に失敗しました。 %s1
%s2: %s3 (%s4)
%s5
%s6

Operation not allowed on nonleaf

[新メッセージ]

IREP: ERROR: irep14208: ldapmodify: Failed to delete the entry. %s1
ldap_delete: %s2 (%s3)
IREP: エラー: irep14208: ldapmodify: エントリの削除に失敗しました。 %s1
ldap_delete: %s2 (%s3)

Operation not allowed on RDN

[新メッセージ]

IREP: ERROR: irep14209: ldapmodify: Failed to modify the RDN. %s1
ldap_modrdn: %s2 (%s3)
IREP: エラー: irep14209: ldapmodify: エントリの識別名変更に失敗しました。 %s1
ldap_modrdn: %s2 (%s3)

Operations error

[新メッセージ]

IREP: ERROR: irep14210: ldapsearch: Search failed. %s1: %s2 (%s3)
IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。 %s1: %s2 (%s3)

Out of memory

[新メッセージ]

IREP: ERROR: irep14105: %s1: %s2: There is not enough memory available.
%s3: %s4 (%s5).
%s6
%s7
IREP: エラー: irep14105: %s1: %s2: メモリ不足が発生しました。
%s3: %s4 (%s5).
%s6
%s7

[P]で始まる旧メッセージ

Protocol error

[新メッセージ]

以下のどれかになります。

IREP: ERROR: irep14207: ldapmodify: Failed to modify the entry. %s1
%s2: %s3 (%s4)
%s5
%s6
IREP: エラー: irep14207: ldapmodify: エントリの更新に失敗しました。 %s1
%s2: %s3 (%s4)
%s5
%s6

IREP: ERROR: irep14208: ldapmodify: Failed to delete the entry. %s1
ldap_delete: %s2 (%s3)
IREP: エラー: irep14208: ldapmodify: エントリの削除に失敗しました。 %s1
ldap_delete: %s2 (%s3)

IREP: ERROR: irep14209: ldapmodify: Failed to modify the RDN. %s1
ldap_modrdn: %s2 (%s3)
IREP: エラー: irep14209: ldapmodify: エントリの識別名変更に失敗しました。 %s1
ldap_modrdn: %s2 (%s3)

IREP: ERROR: irep14210: ldapsearch: Search failed. %s1: %s2 (%s3)
IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。 %s1: %s2 (%s3)

[R]で始まる旧メッセージ

- [realloc of %s1 bytes failed](#)
- [Results too large](#)

realloc of %s1 bytes failed

[新メッセージ]

IREP: ERROR: irep14105: %s1: %s2: There is not enough memory available.
%s3: %s4 (%s5).
%s6
%s7
IREP: エラー: irep14105: %s1: %s2: メモリ不足が発生しました。
%s3: %s4 (%s5).
%s6
%s7

Results too large

[新メッセージ]

IREP: ERROR: irep14210: ldapsearch: Search failed. %s1: %s2 (%s3)

IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。 %s1: %s2 (%s3)

[S]で始まる旧メッセージ

- [scope should be base, one, or sub](#)
- [Sizelimit exceeded](#)
- [SSL connect error %s1 \(%s2 \)](#)
- [Strong authentication not supported](#)
- [Strong authentication required](#)

scope should be base, one, or sub

[新メッセージ]

IREP: ERROR: irep14021: ldapsearch: Specify one of base, one, or sub for the search scope.

IREP: エラー: irep14021: ldapsearch: 検索の範囲には、base、one、subのどれか1つを指定してください。

Sizelimit exceeded

[新メッセージ]

IREP: ERROR: irep14210: ldapsearch: Search failed. %s1: %s2 (%s3)

IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。 %s1: %s2 (%s3)

SSL connect error %s1 (%s2)

[新メッセージ]

IREP: ERROR: irep14183: SSL cannot be used for the repository. %s1 (%s2)

IREP: エラー: irep14183: リポジトリとSSL通信できません。 %s1 (%s2)

Strong authentication not supported

[新メッセージ]

IREP: ERROR: irep14207: ldapmodify: Failed to modify the entry. %s1

%s2: %s3 (%s4)

%s5

%s6

IREP: エラー: irep14207: ldapmodify: エントリの更新に失敗しました。 %s1

%s2: %s3 (%s4)

%s5

%s6

Strong authentication required

[新メッセージ]

以下のどれかになります。

IREP: ERROR: irep14207: ldapmodify: Failed to modify the entry. %s1

%s2: %s3 (%s4)

%s5

%s6

IREP: エラー: irep14207: ldapmodify: エントリの更新に失敗しました。 %s1

%s2: %s3 (%s4)

%s5

%s6

IREP: ERROR: irep14208: ldapmodify: Failed to delete the entry. %s1

ldap_delete: %s2 (%s3)

IREP: エラー: irep14208: ldapmodify: エントリの削除に失敗しました。%s1
ldap_delete: %s2 (%s3)

IREP: ERROR: irep14209: ldapmodify: Failed to modify the RDN. %s1
ldap_modrdn: %s2 (%s3)
IREP: エラー: irep14209: ldapmodify: エントリの識別名変更に失敗しました。%s1
ldap_modrdn: %s2 (%s3)

IREP: ERROR: irep14210: ldapsearch: Search failed. %s1: %s2 (%s3)
IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。%s1: %s2 (%s3)

[T]で始まる旧メッセージ

- The certificate of the client isn't correct %s1 (%s2)
- The certificate of the client or the server is not correct %s1 (%s2)
- The certificate of the server isn't correct %s1 (%s2)
- The error concerning SSL occurred %s1 (%s2)
- The file converting the user definition code could not open.(ENOENT)
- The format of table converting the user definition code which was not supporting was specified.(ENOEXEC)
- The SSL environment isn't correct %s1 (%s2)
- The table address converting the user definition code was not specified.(ENODATA)
- The time limit of the CA bureau certificate is set %s1 (%s2)
- The time limit of the certificate of the client is set %s1 (%s2)
- The time limit of the certificate of the client or the server is set %s1 (%s2)
- The time limit of the certificate of the server is set %s1 (%s2)
- Timed out
- timeout by SSL %s1 (%s2)
- Timelimit exceeded
- Too many tokens (max %s1)
- Type or value exists

The certificate of the client isn't correct %s1 (%s2)

[新メッセージ]

IREP: ERROR: irep14181: Failed to verify the Site certificate that was received. %s1 (%s2)
IREP: エラー: irep14181: 受信したサイト証明書の検証に失敗しました。%s1 (%s2)

The certificate of the client or the server is not correct %s1 (%s2)

[新メッセージ]

IREP: ERROR: irep14176: Failed to verify the CA certificate or Site certificate. %s1 (%s2)
IREP: エラー: irep14176: 認証局証明書、またはサイト証明書の検証に失敗しました。%s1 (%s2)

The certificate of the server isn't correct %s1 (%s2)

[新メッセージ]

IREP: ERROR: irep14179: Failed to verify the Site certificate of the server you want to access. %s1 (%s2)
IREP: エラー: irep14179: 接続先のサイト証明書の検証に失敗しました。%s1 (%s2)

The error concerning SSL occurred %s1 (%s2)

[新メッセージ]

IREP: ERROR: irep14184: An error occurred using SSL. %s1 (%s2)

IREP: エラー: irep14184: SSL通信でエラーが発生しました。 %s1 (%s2)

The file converting the user definition code could not open.(ENOENT)

[新メッセージ]

IREP: ERROR: irep14304: %s1: Failed to open the character code conversion library. (ENOENT)

%s2: %s3 (%s4).

%s5

%s6

IREP: エラー: irep14304: %s1: 文字コード変換ライブラリのオープンに失敗しました。 (ENOENT)

%s2: %s3 (%s4).

%s5

%s6

The format of table converting the user definition code which was not supporting was specified.(ENOEXEC)

[新メッセージ]

IREP: ERROR: irep14306: %s1: Failed to convert the character code. (ENOEXEC)

%s2: %s3 (%s4).

%s5

%s6

IREP: エラー: irep14306: %s1: 文字コード変換に失敗しました。 (ENOEXEC)

%s2: %s3 (%s4).

%s5

%s6

The SSL environment isn't correct %s1 (%s2)

[新メッセージ]

IREP: ERROR: irep14170: The certificate environment is incorrect. %s1 (%s2)

IREP: エラー: irep14170: 証明書環境に誤りがあります。 %s1 (%s2)

The table address converting the user definition code was not specified.(ENODATA)

[新メッセージ]

IREP: ERROR: irep14307: %s1: Failed to convert the character code. (ENODATA)

%s2: %s3 (%s4).

%s5

%s6

IREP: エラー: irep14307: %s1: 文字コード変換に失敗しました。 (ENODATA)

%s2: %s3 (%s4).

%s5

%s6

The time limit of the CA bureau certificate is set %s1 (%s2)

[新メッセージ]

IREP: ERROR: irep14175: The CA certificate has expired. %s1 (%s2)

IREP: エラー: irep14175: 認証局証明書の有効期限が切れています。 %s1 (%s2)

The time limit of the certificate of the client is set %s1 (%s2)

[新メッセージ]

IREP: ERROR: irep14180: The Site certificate that was received has expired. %s1 (%s2)

IREP: エラー: irep14180: 受信したサイト証明書の有効期限が切れています。 %s1 (%s2)

The time limit of the certificate of the client or the server is set %s1 (%s2)

[新メッセージ]

IREP: ERROR: irep14174: The Site certificate has expired. %s1 (%s2).

IREP: エラー: irep14174: サイト証明書の有効期限が切れています。 %s1 (%s2)

The time limit of the certificate of the server is set %s1 (%s2)

[新メッセージ]

IREP: ERROR: irep14178: The Site certificate of the server to you want to access has expired. %s1 (%s2)

IREP: エラー: irep14178: 接続先のサイト証明書の有効期限が切れています。 %s1 (%s2)

Timed out

[新メッセージ]

IREP: ERROR: irep14210: ldapsearch: Search failed. %s1: %s2 (%s3)

IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。 %s1: %s2 (%s3)

timeout by SSL %s1 (%s2)

[新メッセージ]

IREP: ERROR: irep14173: There was a timeout during the SSL connection. %s1 (%s2)

IREP: エラー: irep14173: SSL接続中にタイムアウトになりました。 %s1 (%s2)

Timelimit exceeded

[新メッセージ]

以下のどちらかになります。

IREP: ERROR: irep14210: ldapsearch: Search failed. %s1: %s2 (%s3)

IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。 %s1: %s2 (%s3)

IREP: ERROR: irep14211: ldapsearch: Search failed. %s1 (%s2)

%s3

%s4

IREP: エラー: irep14211: ldapsearch: 検索に失敗しました。 %s1 (%s2)

%s3

%s4

Too many tokens (max %s1)

[新メッセージ]

IREP: ERROR: irep14157: The description format of the SSL environment definition file is incorrect.

IREP: エラー: irep14157: SSL環境定義ファイルの記述形式に誤りがあります。

Type or value exists

[新メッセージ]

IREP: ERROR: irep14209: ldapmodify: Failed to modify the RDN. %s1

ldap_modrdn: %s2 (%s3)

IREP: エラー: irep14209: ldapmodify: エントリの識別名変更に失敗しました。 %s1

ldap_modrdn: %s2 (%s3)

[U]で始まる旧メッセージ

- [Unavailable critical extension](#)
- [Undefined attribute type](#)
- [Unknown authentication method](#)

Unavailable critical extension

[新メッセージ]

IREP: ERROR: irep14207: ldapmodify: Failed to modify the entry. %s1
%s2: %s3 (%s4)
%s5
%s6
IREP: エラー: irep14207: ldapmodify: エントリの更新に失敗しました。 %s1
%s2: %s3 (%s4)
%s5
%s6

Undefined attribute type

[新メッセージ]

IREP: ERROR: irep14209: ldapmodify: Failed to modify the RDN. %s1
ldap_modrdn: %s2 (%s3)
IREP: エラー: irep14209: ldapmodify: エントリの識別名変更に失敗しました。 %s1
ldap_modrdn: %s2 (%s3)

Unknown authentication method

[新メッセージ]

以下のどれかになります。

IREP: ERROR: irep14208: ldapmodify: Failed to delete the entry. %s1
ldap_delete: %s2 (%s3)
IREP: エラー: irep14208: ldapmodify: エントリの削除に失敗しました。 %s1
ldap_delete: %s2 (%s3)

IREP: ERROR: irep14209: ldapmodify: Failed to modify the RDN. %s1
ldap_modrdn: %s2 (%s3)
IREP: エラー: irep14209: ldapmodify: エントリの識別名変更に失敗しました。 %s1
ldap_modrdn: %s2 (%s3)

IREP: ERROR: irep14210: ldapsearch: Search failed. %s1: %s2 (%s3)
IREP: エラー: irep14210: ldapsearch: 検索に失敗しました。 %s1: %s2 (%s3)

[V]で始まる旧メッセージ

version error

[新メッセージ]

IREP: ERROR: irep14011: ldapmodify: The contents specified in LDIF version line %s1 are incorrect. These contents will be ignored. (line %s2)
IREP: エラー: irep14011: ldapmodify: LDIFのversion行の指定%s1に誤りがあります。無視しました。(%s2行目)

[可変情報]で始まる旧メッセージ

- %s1 : Can't open file(line %s2 of entry : %s3)
(line %s2 of entry : %s3)
- %s1: line %s2: bad config line (ignored)
- %s1: line %s2: missing path-name in %s3 <path-name> line
- %s1: line %s2: missing string in %s3 <string> line
- %s1: line %s2: missing token-passwd in %s3 <token-passwd> line
- %s1: No such file or directory

**%s1 : Can't open file
(line %s2 of entry : %s3)**

[新メッセージ]

IREP: ERROR: irep14030: %s1: Failed to open the specified file. %s2.

IREP: エラー: irep14030: %s1: 指定したファイル%s2のオープンに失敗しました。

%s1: line %s2: bad config line (ignored)

[新メッセージ]

IREP: ERROR: irep14151: %s1: line %s2: The description format of SSL environment definition file is incorrect. The contents will be ignored.

IREP: エラー: irep14151: SSL環境定義ファイル%s1の%s2行目の記述形式に誤りがあります。無視しました。

%s1: line %s2: missing path-name in %s3 <path-name> line

[新メッセージ]

IREP: ERROR: irep14154: %s1: line %s2: The directory path specified for configuration option %s3 in SSL environment definition file is incorrect.

IREP: エラー: irep14154: SSL環境定義ファイル%s1の%s2行目の定義項目%s3に指定したディレクトリのパスに誤りがあります。

%s1: line %s2: missing string in %s3 <string> line

[新メッセージ]

IREP: ERROR: irep14153: %s1: line %s2: The value for configuration option %s3 in SSL environment definition file is incorrect.

IREP: エラー: irep14153: SSL環境定義ファイル%s1の%s2行目の定義項目%s3の値に誤りがあります。

%s1: line %s2: missing token-passwd in %s3 <token-passwd> line

[新メッセージ]

IREP: ERROR: irep14155: %s1: line %s2: The user PIN specified for configuration option tkn_pwd in SSL environment definition file is incorrect.

IREP: エラー: irep14155: SSL環境定義ファイル%s1の%s2行目の定義項目 tkn_pwd に指定したユーザPINに誤りがあります。

%s1: No such file or directory

[新メッセージ]

以下のどちらかになります。

IREP: ERROR: irep14030: %s1: Failed to open the specified file. %s2.

IREP: エラー: irep14030: %s1: 指定したファイル%s2のオープンに失敗しました。

IREP: ERROR: irep14150: Failed to open the specified SSL environment definition file. %s1

IREP: エラー: irep14150: 指定したSSL環境定義ファイル%s1をオープンできませんでした。

◆LDAPコマンドの廃止したメッセージ

LDAPコマンドの廃止したメッセージを以下に示します。

- alias deref should be never, search, find, or always
- Alias dereferencing problem
- Alias problem
- base64 decode err. data length is %s1
(line %s2 of entry : %s3)
- ber_alloc(sort keys) failed
- ber_flatten(sortKeyList) failed
- ber_printf(sortKeyList %s1) failed
- Failed base64 encode

- Failed to convert from %s1 to %s2 : %s3
- Failed to sort entries. Not enough memory
- invalid base64 code
- ldmv3: invalid format (line %s1 of entry: %s2)
- ldmv3: invalid Michigan format. missing value on line %s1 (attr is %s2)
- Loop detected
- Not supported
- Not support version
- The attribute value of the %s1 character code is not correct.(EBADRQC)
- Unknown error
- You should specify the sort key.

◆irepencupinコマンドの変更したメッセージ

irepencupinコマンドの変更したメッセージを以下に示します。

旧メッセージの先頭文字

[E] [I]

[E]で始まる旧メッセージ

- Error : No authority to execute the program
- Error : SSL environmental file does not exist (%s)
- Error : The specified file is a directory (%s)
- Error : Unexpected (%s)
- Error : User PIN is not defined.

Error : No authority to execute the program

[新メッセージ]

IREP: ERROR: irep10403: Option "%s1" is duplicated.

IREP: エラー: irep10403: オプション "%s1" が複数指定されています。

Error : SSL environmental file does not exist (%s)

[新メッセージ]

IREP: ERROR: irep10411: There is no authority to execute the program. Please execute it by the manager authority.

IREP: エラー: irep10411: プログラムを実行する権限がありません。管理者権限で実行してください。

Error : The specified file is a directory (%s)

[新メッセージ]

IREP: ERROR: irep10406: Specified file in option "%s1" does not exist.

IREP: エラー: irep10406: オプション "%s1" で指定されたファイルが存在しません。

Error : Unexpected (%s)

[新メッセージ]

IREP: ERROR: irep10408: No access permission to specified file in option "%s1".

IREP: エラー: irep10408: オプション "%s1" で指定されたファイルへのアクセス権に誤りがあります。

Error : User PIN is not defined.

[新メッセージ]

IREP: ERROR: irep10413: An error was detected in the function during processing. func=%s1, errno=%s2

IREP: エラー: irep10413: 処理中に関数でエラーが検出されました。func=%s1, errno=%s2

[I]で始まる旧メッセージ

- [irepencupin : illegal option -- %s](#)
- [irepencupin is complete.](#)

irepencupin : illegal option -- %s

[新メッセージ]

IREP: ERROR: irep10414: Configure the user PIN.

IREP: エラー: irep10414: ユーザPINを定義してください。

irepencupin is complete.

[新メッセージ]

IREP: INFO: irep10415: irepencupin was completed.

IREP: 情報: irep10415: irepencupinコマンドが正常に完了しました。

2.5.18 J2EE使用時に出力される例外情報

ここでは、以下について説明します。メッセージの詳細は、“メッセージ集”を参照してください。

- [J2EEアプリケーションクライアントが返す例外メッセージについて](#)

■J2EEアプリケーションクライアントが返す例外メッセージについて

Interstage Application Server V6.0から、Interstage ディレクトリサービスがディレクトリサービスに加わりました。そのため、以下のメッセージを変更しました。

- Interstage V5.1以前:"javax.naming.NamingException: Access to InfoDirectory failed :"
- Interstage V6.0以降:"javax.naming.NamingException: Access to directory service failed :"

Interstage Application Server V5.1以前のメッセージを出力させるためには、以下のVMオプションを設定してください。

VMオプション

com.fujitsu.interstage.j2ee.jndi.directoryservice.msg.directoryname

設定値

InfoDirectory	:Interstage V5.1以前のメッセージを出力
directoryservice	:Interstage V6.0以降のメッセージを出力

第3章 旧バージョン・レベルからの資源の移行方法

3.1 移行方法

Interstage Application Server/Interstage Web Serverのバックアップおよびリストア機能を使用すれば、旧バージョン・レベルの運用環境を移行できます。

以下に、バックアップおよびリストア機能を使用した移行の手順について説明します。

3.1.1 移行手順

旧バージョン・レベルからの移行の手順について説明します。

1. 旧バージョン・レベルでサポートしているInterstageの資源のバックアップ機能を使用し、Interstage運用環境の資源をバックアップします。
2. 旧バージョン・レベルのInterstage Application Serverをアンインストールします。
3. 本バージョン・レベルのInterstage Application Serverをインストールします。
詳細な手順は、“インストールガイド”を参照してください。
4. 本バージョン・レベルのリストア機能を使用し、バックアップ資源をリストアします。
詳細な手順は、“運用ガイド(基本編)”の“メンテナンス(資源のバックアップ)”を参照してください。

5. **Solaris** **Linux**
issetsecuritymodeコマンドを使用して、セキュリティ権限の設定を行います。コマンドの詳細については、“リファレンスマニュアル(コマンド編)”を参照してください。

なお、上記手順で移行できない機能もあります。それらの機能については、本マニュアルにおいて移行方法を説明していますので、本マニュアルを一通り確認してください。

また、移行作業は、移行の必要な資源に対する移行方法を確認した上で、作業を開始してください。



注意

- ・ 旧バージョン・レベルのInterstage Application Server資源のバックアップ方法およびアンインストール方法については、該当するバージョン・レベルのInterstageのマニュアルを参照してください。
- ・ 旧バージョン・レベルでバックアップしたInterstage Application Server資源は、本バージョン・レベルのコマンドを使用してリストアしてください。

3.2 旧バージョン・レベルのユーザ資源の互換

旧バージョン・レベルで作成・使用したユーザ資源(環境定義ファイルやユーザデータなど)を本バージョン・レベルで使用する上での互換の有無を示します。



注意

“旧バージョン・レベル”として記載しているバージョン・レベル表記は、以下を使用しています。読み替えて利用してください。

バージョン・レベル 表記	対象製品名/バージョン・レベル
V9.0	Interstage Application Server V9.0 Interstage Web Server V9.0
8.0	Interstage Application Server 8.0
V7.0、V7.x、V7	Interstage Application Server 7.0/7.0.1 (Solarisの場合) Interstage Application Server V7.0L10/V7.0L11 (Windows(R)/Linuxの場合)
V6.0、V6.x、V6	Interstage Application Server V6.0L10/V6.0L10A/V6.0L10B/V6.0L10C (Windows(R)の場合) Interstage Application Server 6.0/6.0.1/6.0.2 (Solarisの場合) Interstage Application Server V6.0L10/V6.0L11/V6.0L10A/V6.0L11A (Linuxの場合)
V5.1	Interstage Application Server 5.1 (Solarisの場合) Interstage Application Server V5.0L20 (Windows(R)/Linuxの場合)
V5.0	Interstage Application Server 5.0 (Solarisの場合) Interstage Application Server V5.0L10 (Windows(R)/Linuxの場合)
V5.x、V5	Interstage Application Server 5.0/5.1 (Solarisの場合) Interstage Application Server V5.0L10/V5.0L20 (Windows(R)/Linuxの場合)
V4.1 (注)	INTERSTAGE Application Server 4.1 (Solaris/Linuxの場合) INTERSTAGE Application Server V4.0L20 (Windows(R)の場合)
V4.0 (注)	INTERSTAGE Application Server 4.0 (Solarisの場合) INTERSTAGE Application Server V4.0L10 (Windows(R)の場合)
V4.x (注)	INTERSTAGE Application Server 4.0/4.1 (Solaris/Linuxの場合) INTERSTAGE Application Server V4.0L10/V4.0L20 (Windows(R)の場合)

注) V3以前も同様。

3.2.1 Interstageの動作環境定義ファイル

■Interstageの動作環境定義ファイル

旧バージョン・レベル	本バージョン・レベルでの使用可否
V1.x	—
V2.x	△ (注1) (注2)
V3.x	△ (注2)
V4.x ~ V6.0	○
V7.0 以降	○ (注3)

[○:互換あり。 △:一部互換なし。 ×:互換なし。 —:定義が存在しない。]

注1)

isinitコマンド実行時に運用形態にtype3を指定し、かつ、EJBオプションを指定する場合は、Interstage動作環境定義に“IR USE = local”を追加後、コマンドを実行してください。

注2)

isinitコマンド実行時に運用形態にtype2を指定する場合は、Interstage動作環境定義に“OTS Setup mode = sys”を追加後、コマンドを実行してください。

注3)

Interstage動作環境定義の“NS Jp”および“NS Locale”は、互換用の定義項目です。本定義を使用する場合は、Interstage Application Server V6.0以前のマニュアルを参照してください。

3.2.2 Interstage HTTP Serverの資源

■環境定義ファイル

旧バージョン・レベル	本バージョン・レベルでの使用可否
V5.x ~ 8.0	× (注1)
V9.0 以降	○

[○:互換あり。△:一部互換なし。×:互換なし。ー:定義が存在しない。]

注1)

Interstage HTTP Serverは、Apache HTTP Server 1.3ベースからApache HTTP Server 2.0ベースのWebサーバへ変更されました。8.0以前のInterstage HTTP Server (Apache HTTP Server 1.3ベース)の資源との互換はありません。8.0以前のInterstage HTTP Server (Apache HTTP Server 1.3ベース)からInterstage HTTP Server (Apache HTTP Server 2.0ベース)へ移行する場合は、“[4.1.1 8.0以前のInterstage HTTP Server \(Apache HTTP Server 1.3ベース\)からの移行](#)”を参照して移行してください。

3.2.3 J2EEの資源

■セキュリティ管理環境定義ファイル

旧バージョン・レベル	本バージョン・レベルでの使用可否
V4.0	△
V4.1 以降	○

[○:互換あり。△:一部互換なし。×:互換なし。ー:定義が存在しない。]

■isj2eeadminコマンドの定義ファイル

旧バージョン・レベル	本バージョン・レベルでの使用可否
8.0	○ (注1)

[○:互換あり。△:一部互換なし。×:互換なし。ー:定義が存在しない。]

注1)

以前のバージョン・レベルで抽出した定義ファイルを利用して、定義を新規追加する場合、isj2eeadminコマンドの互換オプションを使用することができます。

互換オプションについての詳細は、“リファレンスマニュアル(コマンド編)”の“isj2eeadmin”の“互換オプションについて”を参照してください。

3.2.4 Servletサービスの資源

■環境定義ファイル

旧バージョン・レベル	本バージョン・レベルでの使用可否
V2.x ~ V5.x	×
V6.0 以降	△ (注1)

[○:互換あり。△:一部互換なし。×:互換なし。ー:定義が存在しない。]

注1)

本バージョン・レベルではTomcat5.5ベースのServletサービスを提供しています。また、互換用として以前のTomcat4.1ベースのServletサービスも提供しています。Tomcat5.5ベースのServletサービスとTomcat4.1ベースのServletサービスの資源は互換はありません。Tomcat5.5ベースのServletサービスへ移行する場合は、“[4.2 Servletサービス\(Tomcat 5.5ベースのサーブレット実行環境\)への移行](#)”を参照して移行してください。

なお、[RHEL5\(x86\)](#)、[RHEL5\(Intel64\)](#)、[RHEL5\(IPF\)](#)、[RHEL6\(x86\)](#)、[RHEL6\(Intel64\)](#)の場合にはTomcat4.1ベースのServletサービスは提供していないため、Tomcat5.5ベースのServletサービスへ移行してください。



注意

本バージョン・レベルでは、Interstage Application Server V5.1以前のサーブレット実行環境であるTomcat3.1ベースのサーブレット実行環境(V5.1以前のServletサービス)を提供していません。V5.1以前のServletサービスの環境定義ファイルは使用できません。

3.2.5 Interstage JMSの資源

Interstage JMSは、以下の製品で利用可能です。

- Interstage Application Server Enterprise Edition
- Interstage Application Server Standard-J Edition

■環境定義ファイル

旧バージョン・レベル	本バージョン・レベルでの使用可否
V4.x ~ V5.x	○ (注1)
V6.0 以降	○

[○:互換あり。△:一部互換なし。×:互換なし。ー:定義が存在しない。]

注1)

Interstage Application Server V4.x/V5.xで作成した以下の定義のJNDI名に「コロンの(:)」が含まれている場合、本バージョン・レベルにおいて、Interstage管理コンソールを使用して以下の定義の一覧を表示すると、エラーメッセージ“jms9997”が出力され操作することができません。

- ー ConnectionFactory定義
- ー Destination定義

「コロンの(:)」が設定された上記の定義を管理する場合は、JMS運用コマンドを使用するか、またはInterstage管理コンソールを使用する前にJMS運用コマンドを使用して、該当する上記の定義を削除してください。



注意

イベントサービス運用コマンド使用時に利用するイベントサービスの資源について注意する必要があります。“[3.2.14 イベントサービスの資源](#)”を参照して確認してください。

3.2.6 SOAPサービスの資源

SOAPサービスは、以下の製品で利用可能です。

- Interstage Application Server Enterprise Edition
- Interstage Application Server Standard-J Edition

本節では、旧バージョン・レベルで開発したSOAPサービスの資源を本製品のSOAPサービスに移行する場合について説明します。新機能J2EEのWebサービス機能への移行については、“[8.3 SOAPサービスの移行](#)”の“[8.3.1 J2EEのWebサービス機能への移行](#)”を参照してください。

■環境定義ファイル

Interstage V9.1.0の“SOAPサービス ユーザーズガイド”の“サービスシステムの環境構築”および“Webアプリケーションファイルの編集”を参照して、旧バージョン・レベルと同様な環境を構築してください。

Webサービス情報管理ファイルをInterstage Application Server V5.x/V4.xから移行する場合は、“[8.3 SOAPサービスの移行](#)”の“[8.3.3 Interstage Application Server V6.0以前からの移行方法](#)”を参照してください。

■Webサービスのセキュリティ環境

旧バージョン・レベルのサービスシステム上に構築したWebサービスで使用するセキュリティ環境はInterstage Application Server V6.0で使用することはできません、移行することはできません。soapSetSecurity、soapMngSecurityコマンドで操作してください。

Interstage管理コンソールからWebサービスのセキュリティ環境を構築および操作するためには、Interstage証明書環境を構築する必要があります。詳細は“セキュリティシステム運用ガイド”の“Interstage証明書環境の構築と利用”を参照してください。

なおクライアントシステム上に構築したWebサービスのセキュリティ環境はsoapSetSecurity、soapMngSecurityコマンドで操作してください。

■ユーザデータ(Webサービス情報管理ファイル)

旧バージョン・レベル	本バージョン・レベルでの使用可否
V4.x ~ V5.0	× (注1)
V5.1 以降	○

[○:互換あり。△:一部互換なし。×:互換なし。ー:定義が存在しない。]

注1)

soapdd2wsddコマンドを使用して旧バージョン・レベルのWebサービス情報管理ファイルを本バージョン・レベルの形式に変換できます。

■ユーザデータ(CORBA/SOAPゲートウェイ)

旧バージョン・レベル	本バージョン・レベルでの使用可否
V4.x	△
V5.x 以降	○

[○:互換あり。△:一部互換なし。×:互換なし。ー:定義が存在しない。]

■Webサービス情報編集ツール

Webサービス情報編集ツールは、Interstage管理コンソールから起動します。ディレクトリサービスで管理していた認証情報はシステム認証に移行する必要があります。

■旧バージョン・レベルのJavaクラスライブラリのAPIを使用したRPCアプリケーション

旧バージョン・レベル	本バージョン・レベルでの使用可否
V4.x ~ V5.x	△
V6.0 以降	○

[○:互換あり。△:一部互換なし。×:互換なし。ー:定義が存在しない。]

RPCアプリケーションのパラメタおよび返り値に含まれる配列型は、単一参照(変数の値ごとにインスタンスが生成されます)になります。

3.2.7 Interstage証明書環境の資源

■環境定義ファイル

旧バージョン・レベル	本バージョン・レベルでの使用可否
V6.0 ～ V7.0	○ (注1)
8.0 ～ V9.0.1	○ (注2)
V9.1 以降	○

[○:互換あり。△:一部互換なし。×:互換なし。－:定義が存在しない。]

注1)

Interstage証明書環境はそのまま利用することができます。ただし、Interstage管理コンソール([システム]>[セキュリティ]>[SSL])でSSL定義を変更する場合には、旧バージョン・レベルで設定したときとは下記が異なります。

- 強い暗号アルゴリズムが選択可能になる

[暗号化方法]において、AES暗号アルゴリズムとSC2000暗号アルゴリズムを含む暗号化方法が表示されるようになり、また、選択できるようになります。ただし、今日ではまだAES暗号アルゴリズムをサポートしている製品は限られていますし、SC2000暗号アルゴリズムは富士通研究所で開発された独自の暗号アルゴリズムです。そのため、他製品とSSL通信する場合は、接続性を保証するために、SC2000暗号アルゴリズムは選択せず、また必要に応じて他製品がサポートしている暗号化方式も選択するようにしてください。

- 安全性が低いSSL定義の設定に警告が表示される

[プロトコルバージョン]に「SSL2.0」を選択し、かつ、[クライアント認証]で「する(クライアント証明書を必ず認証する)」を選択している場合、[適用]ボタンを押すと警告メッセージが表示されるようになります。この設定では、ブラウザなどのクライアント側からクライアント自身の証明書が提示されなくてもサーバに接続ができるため、安全性を損なう可能性があるためです。SSL通信を行う製品やアプリケーションの仕様や設定を考慮したうえで、警告を無視して適用するか、または、SSL定義を変更するか、判断してください。

なお、本バージョン・レベルでは、上記の機能に加え、[暗号化方法]の初期設定で選択されているものから、DES暗号アルゴリズムを含む暗号化方法を外しています。これは、今日ではDES暗号アルゴリズムは安全とは言えなくなっており、より強い暗号アルゴリズムを利用することが求められてきていることによるものです。Interstage Application Server 7.0以前で作成した環境においても、SSL通信を行う製品やアプリケーションの仕様や設定を考慮したうえで、可能であれば、DES暗号アルゴリズムを外すようにしてください。



注意

SSL通信を行う製品やアプリケーションが128bit暗号対応されていない場合(128bit暗号対応以前の古いWebブラウザを使用している場合など)、本バージョン・レベルで初期設定で選択されている暗号化方法では接続できません。これは、それらの製品やアプリケーションが弱い暗号アルゴリズムしかサポートしていないことによります。安全な暗号通信のためには、128bit暗号に対応した製品やアプリケーションを利用することを推奨します。なお、128bit暗号対応以前の製品やアプリケーションとSSL通信を行う必要がある場合には、弱い暗号アルゴリズムを利用することを認識したうえで、DES暗号アルゴリズムなどを含んだ暗号化方法を追加で選択するようにしてください。

また、同様に、本バージョン・レベルでは、[プロトコルバージョン]の初期設定で選択されているものから「SSL2.0」を外しています。これは、SSL通信のプロトコルとしてはSSL3.0やTLS1.0を使用するのが主流となっており、SSL2.0は下位互換のために使用されていることによるものです。Interstage Application Server 7.0以前で作成した環境においても、SSL通信を行う製品やアプリケーションの仕様や設定を考慮したうえで、可能であれば、「SSL2.0」を外すようにしてください。

注2)

Interstage証明書環境はそのまま利用することができます。ただし、Interstage管理コンソール([システム]>[セキュリティ]>[SSL])でSSL定義を新規作成した場合には、旧バージョン・レベルで新規作成したときとは下記が異なります。

- SC2000暗号アルゴリズムは選択されない

[暗号化方法]の初期設定で選択されているものから、SC2000暗号アルゴリズムを含む暗号化方法を外しています。SC2000暗号アルゴリズムは富士通研究所で開発された独自の暗号アルゴリズムであるため、SSL通信を行う製品やアプリケーションが

SC2000暗号アルゴリズムをサポートしている場合に、選択してください。他製品とSSL通信する場合は、接続性を保証するために、SC2000暗号アルゴリズムを選択しないことを推奨します。

3.2.8 Interstage シングル・サインオンの資源

■SSL環境

Interstage Application Server V5.1のシステムに構築したSSL環境を本バージョン・レベルにおいて使用できますが、Interstage管理コンソールからは操作できません。Interstage管理コンソールから操作する場合は、SSL環境を再構築する必要があります。SSLの環境設定については、“セキュリティシステム運用ガイド”の“Interstage証明書環境の構築と利用”を参照してください。

■環境定義ファイル

Interstage シングル・サインオンでは、旧バージョン・レベルで使った環境定義ファイルを本バージョン・レベルで使用できます。

旧バージョン・レベル	本バージョン・レベルでの使用可否
V5.1	△ (注1) (注2)
V6.0	△ (注1) (注3)
V7.0	△ (注1)
8.0以降	○

[○:互換あり。 △:一部互換なし。 ×:互換なし。 -:定義が存在しない。]

注1)

業務サーバが257台以上定義されている場合は、1台のマシンに業務サーバが256台以下に配置されるようにサーバ構成を見直し、移行してください。

注2)

移行時に環境定義ファイルの編集が必要です。移行手順については、“[5.11 Interstage シングル・サインオンの移行](#)”を参照してください。

リポジトリサーバと認証サーバで使用する環境定義ファイルのバージョン・レベルが異なる場合には、“シングル・サインオン運用ガイド”の“旧バージョン・レベルの環境を使用する場合の注意事項”を参照してください。

注3)

以下の製品を使用している場合、移行時に環境定義ファイルの編集が必要です。移行手順については、“[5.11 Interstage シングル・サインオンの移行](#)”を参照してください。

- Interstage Application Server Plus
- Interstage Application Server Web-J Edition

3.2.9 Interstage ディレクトリサービスの資源

Interstage ディレクトリサービスは、以下の製品で利用可能です。

- Windows Server(R) for Itanium-based Systems/[RHEL-AS4\(IPF\)](#)/[RHEL5\(IPF\)](#)の場合
 - Interstage Application Server Enterprise Edition
- 上記以外のオペレーティングシステムの場合
 - Interstage Application Server Enterprise Edition
 - Interstage Application Server Standard-J Edition

■リポジトリ環境

旧バージョン・レベル	本バージョン・レベルでの使用可否
V6.0 以降	△ (注1) (注2) (注3)

[○:互換あり。 △:一部互換なし。 ×:互換なし。 -:定義が存在しない。]

注1)

リポジトリのデータベースに標準データベースを使用していた場合、リポジトリが使用するポート番号として、2000～5999の範囲は使用できません。旧バージョン・レベルにおいて使用していたポート番号は、移行時に6000～65535の範囲のポート番号に自動的に変更されます。

標準データベース(Fujitsu Enabler)のデータストアが使用しているポート番号は、以下の方法で、確認してください。

Windows

1. %IS_HOME%\Enabler\server\bin\omslist.exe -l

Solaris

1. LANG環境変数に「C」を設定します。
2. /opt/FJSVena/server/bin/omslist -l

Linux

1. LANG環境変数に「C」を設定します。
2. /opt/FJSVena/Enabler/server/bin/omslist -l

【実行結果例】

```
-----
rep001: server=host01 port=6000 XF
rep002: server=host01 port=6001 XF
-----
```

「port」の値が、Fujitsu Enablerのデータストアが使用しているポート番号です。

ポート番号が変更された結果、他アプリケーションで使用しているポート番号と衝突する可能性があります。他アプリケーションで使用しているポート番号に変更された場合、以下の方法で他のアプリケーションや他のFujitsu Enablerのデータストアが使用していないものに変更してください。

Windows

1. Fujitsu Enablerのデータストアが使用しているポート番号を変更します。
%IS_HOME%\Enabler\server\bin\omschangeport.exe "リポジトリ名" -pn "新しいポート番号"

Solaris

1. LANG環境変数に「C」を設定します。
2. Fujitsu Enablerのデータストアが使用しているポート番号を変更します。
/opt/FJSVena/server/bin/omschangeport "リポジトリ名" -pn "新しいポート番号"

Linux

1. LANG環境変数に「C」を設定します。
2. Fujitsu Enablerのデータストアが使用しているポート番号を変更します。
/opt/FJSVena/Enabler/server/bin/omschangeport "リポジトリ名" -pn "新しいポート番号"

注2)

リポジトリのデータベースに標準データベースを使用していた場合、標準データベース(Fujitsu Enabler)サービスのポート番号を変更するには、以下の方法で他のアプリケーションや他のFujitsu Enablerのデータストアが使用していないものに変更してください。

Windows

1. Fujitsu Enablerサービスが使用しているポート番号を確認します。

以下のファイルの、「OMS_SERVICE」に定義されている値が、Fujitsu Enablerサービスが使用しているポート番号です。

```
%IS_HOME%\Enabler\server\param\enabler.conf
```

2. 以下の状態であることを確認します。

- Fujitsu Enablerサービスが起動している
- リポジトリが停止している

3. 以下のコマンドを使って、ポート番号を変更します。

```
%IS_HOME%\Enabler\server\bin\omschangeport.exe -u "新しいポート番号"
```

Solaris

1. Fujitsu Enablerサービスが使用しているポート番号を確認します。

以下のファイルの、「OMS_SERVICE」に定義されている値が、Fujitsu Enablerサービスが使用しているポート番号です。

```
/opt/FJSVena/server/param/enabler.conf
```

2. 以下の状態であることを確認します。

- リポジトリが停止している

3. LANG環境変数に「C」を設定します。

4. Fujitsu Enablerサービスを停止します。

```
/opt/FJSVena/server/bin/enablerstop
```

5. 以下のコマンドを使って、ポート番号を変更します。

```
/opt/FJSVena/server/bin/omschangeport -u "新しいポート番号"
```

6. Fujitsu Enablerサービスを起動します。

```
/opt/FJSVena/server/bin/enablerstart
```

Linux

1. Fujitsu Enablerサービスが使用しているポート番号を確認します。

以下のファイルの、「OMS_SERVICE」に定義されている値が、Fujitsu Enablerサービスが使用しているポート番号です。

```
/opt/FJSVena/Enabler/server/param/enabler.conf
```

2. 以下の状態であることを確認します。

- リポジトリが停止している

3. LANG環境変数に「C」を設定します。

4. Fujitsu Enablerサービスを停止します。

```
/opt/FJSVena/Enabler/server/bin/enablerstop
```

5. 以下のコマンドを使って、ポート番号を変更します。

```
/opt/FJSVena/Enabler/server/bin/omschangeport -u "新しいポート番号"
```

6. Fujitsu Enablerサービスを起動します。

```
/opt/FJSVena/Enabler/server/bin/enablerstart
```

注3)

リポジトリのデータベースに標準データベースを使用して、リポジトリのキャッシュ値に100～999の範囲の値を指定していた場合、移行後のリポジトリのキャッシュ値は、管理コンソール上は100～999の範囲の値が表示されますが、リポジトリへのアクセス性能を維持するために、内部的には1000が指定されたものとみなします。

■リポジトリデータ

旧バージョン・レベル	本バージョン・レベルでの使用可否
V6.0 ～ V7.0	△ (注1)
8.0	△ (注2)

[○:互換あり。 △:一部互換なし。 ×:互換なし。 ー:定義が存在しない。]

注1)

ユーザパスワード暗号化方式を初期値から変更していた場合、リポジトリデータだけを移行することはできません。リポジトリ環境を含めて移行してください。

注2)

リポジトリのデータベースにリレーショナルデータベース(RDB)を使用していた場合、RDBのバックアップ機能を使用してデータを移行することはできません。リポジトリのデータをLDAPコマンド等を使用してLDIFファイルに取り出して移行してください。

Windows

LDIFファイルにリポジトリデータを取り出す場合は、最新の緊急修正が適用されていることを確認してから行ってください。

なお、リポジトリのデータベースにRDBを使用していた場合は、リポジトリデータを移行しなくてもアクセスできます。ただし、リポジトリデータを移行せずに、Interstage Application Server 8.0で作成したデータベースを使用する場合は、Interstage Application Server V9.0で追加された以下の機能が使用できません。

【使用できない機能】

- ・アクセス制御
- ・スキーマ拡張

なお、Interstage Application Server 8.0で作成したデータベースを、移行せずに使用する場合の各種操作等については、Interstage Application Server 8.0のマニュアルを参照してください。

■SSL環境

旧バージョン・レベルで作成したSSL定義をそのまま本バージョン・レベルで使用できます。

SSLの環境設定については、“ディレクトリサービス運用ガイド”の“SSL通信環境の構築”を参照してください。

3.2.10 CORBAサービスの資源

■動作環境ファイル(configファイルなど)

旧バージョン・レベル	本バージョン・レベルでの使用可否
V1.x ～ V5.x	○ (注1)
V6.0 以降	○

[○:互換あり。 △:一部互換なし。 ×:互換なし。 ー:定義が存在しない。]

注1)

configファイルのパラメタが一部変更されています。

— 廃止パラメタ

- max_IIOp_init_con
- max_IIOp_req_per_con

— 新規パラメタ

- max_IIOp_resp_requests
- max_IIOp_local_init_con
- max_IIOp_local_init_requests

旧バージョン・レベルのconfigを使用した場合でも、内部ロジックで適切値に読み替えられますので問題はありませんが、“チューニングガイド”を参照してconfigを見直すことを推奨します。

新規パラメタと廃止パラメタの両方が設定されている場合は新規パラメタの設定が優先されます。

また、configのパラメタ変更に伴いodprtcuparamコマンドの出力が変更されています。odprtcuparamコマンドの出力については、“リファレンスマニュアル(コマンド編)”を参照してください。

■ユーザデータ(インタフェースリポジトリ・ネーミングサービスなどのDB)

旧バージョン・レベル	本バージョン・レベルでの使用可否
V1.x ~ V2.x	○ (注1) (注2)
V3.x ~ V6.0	○ (注2)
V7.0 以降	○

[○:互換あり。 △:一部互換なし。 ×:互換なし。 —:定義が存在しない。]

注1)

CORBA標準対応のため、オブジェクトリファレンスのエンコード方式が一部変更されました。このため、以下の条件の時、下位バージョンを使用しているPortable-ORB以外のクライアントアプリケーションでFREE_MEM例外が発生する場合があります。

1. クライアントマシンにV2.0/V2.0L10以前のINTERSTAGEがインストールされている、かつ
2. サーバアプリケーションのオブジェクトリファレンスにコード系を指定している場合。

また、下位バージョンのPortable-ORBを使用している場合は以下の条件の時Portable-ORBクライアントでMARSHAL例外が発生します。

1. クライアントマシンでV2.1/V2.0L20以前のPortable-ORBを使用している、かつ
2. サーバアプリケーションのオブジェクトリファレンスにコード系を指定している場合。

本件の現象が発生した場合は技術員に連絡してください。

注2)

ネーミングサービスにおける日本語バインディング名の使用は、旧バージョン互換用の機能です。日本語バインディング名を使用する場合は、Interstage Application Server V6.0以前のマニュアルを参照してください。

■ユーザアプリケーション

CORBAアプリケーションは、一部の例外(注1)を除いて、異なるバージョン間でのバイナリ互換は保証しません。そのため、ユーザアプリケーションを移行する場合は、ユーザアプリケーションを再ビルドする必要があります。

注1)

Interstage Application Server V7.0以降で作成したCOBOLアプリケーションについては、NetCOBOLで保証する範囲内でバイナリ互換があります。なお、OOCOBOLアプリケーションについては、バイナリ互換は保証しません。

ソースレベルの互換性については以下の表のとおりとなります。

ただし、IDLコンパイラに対する機能追加、修正を実施している場合がありますので、再ビルドの際にはIDLコンパイルを再実施することを推奨します。

旧バージョン・レベル	本バージョン・レベルでの使用可否
V1.x ~ V4.x	△ (注1) (注4)
V5.x ~ V6.0	△ (注1) (注2) (注3) (注4)
V7.0 ~ V9.0	○ (注4)
V9.1以降	○

[○:互換あり。 △:一部互換なし。 ×:互換なし。]

注1)

Windows(R)版Interstage Application Server V7.0以降では、CORBAサービスが提供しているライブラリのコンパイルオプションが変更されました。そのため、C/C++ユーザアプリケーションを移行する場合にはユーザアプリケーションのコンパイルオプションも変更してアプリケーションを再ビルドする必要があります。詳細は、“[Windows\(R\)のアプリケーションのコンパイルオプション変更について](#)”を参照してください。

注2)

Windows(R)用のC++ CORBAワークユニットアプリケーションにおいて、古いiostreamライブラリを使用して標準出力、標準エラー出力にデータを出力している場合、Interstage Application Server V7.0以降ではstdout,stderrファイルにデータが出力されません。iostreamライブラリを標準C++ iostreamライブラリに変更して、ユーザアプリケーションの再ビルドを行ってください。詳細は、“OLTPサーバ運用ガイド”の“カレントディレクトリ”を参照してください。

注3)

Interstage Application Server V6.0以前では、COBOL言語で作成したCORBAワークユニットアプリケーションにおいて、翻訳時に翻訳オプション“SSOUT(環境変数情報名)”を指定していなくても、DISPLAY文の出力がstdoutファイルに出力される場合がありましたが、Interstage Application Server V7.0以降では、“SSOUT(環境変数情報名)”を指定しないと、ワークユニット停止時にアプリケーションがハングアップする場合があります。翻訳時に翻訳オプション“SSOUT(環境変数情報名)”を指定して再ビルドを行ってください。詳細は、“OLTPサーバ運用ガイド”の“カレントディレクトリ”を参照してください。

注4)

Interstage Application Server V9.1以降では、C++アプリケーションに対するIDL生成物内でメモリ不足が発生した場合の処理が修正されました。詳細は、“[C++アプリケーションの動作について](#)”を参照してください。

■SSL環境

SMEEコマンドで作成したSSLの証明書/鍵管理環境をInterstage証明書環境に移行することはできません。Interstage証明書環境を使用する場合は、Interstage管理コンソールを使用して再構築する必要があります。SSLの環境設定については“セキュリティシステム運用ガイド”の“Interstage証明書環境の構築と利用”を参照してください。

3.2.11 コンポーネントランザクションサービスの資源

コンポーネントランザクションサービスは、以下の製品で利用可能です。

- Interstage Application Server Enterprise Edition
- Interstage Application Server Standard-J Edition
注) Standard-J Editionでは、以下のファイルだけを使用することができます。
 - ー ワークユニット定義ファイル

■環境定義ファイル

環境定義ファイルは移行の必要はありません。
isgendefコマンド、isregistdefコマンド、isinitコマンドで環境設定してください。

旧バージョン・レベル	本バージョン・レベルでの使用可否
V1.x 以降	○

[○:互換あり。△:一部互換なし。×:互換なし。ー:定義が存在しない。]

■ワークユニット定義ファイル

ワークユニットの定義ファイルを新システムに移動してください。
その後、isaddwudefコマンド、またはtdadddefコマンドにより、ワークユニットを登録してください。

旧バージョン・レベル	本バージョン・レベルでの使用可否
V1.x ~ V3.x	△ (注1)
V4.x 以降	○

[○:互換あり。△:一部互換なし。×:互換なし。ー:定義が存在しない。]

注1)

データベース連携サービスを利用する場合、INTERSTAGE Application Server V3.x以前のバージョン・レベルで使用していたワークユニット定義はそのままでは使用することができません。移行方法については、“[7.7 データベース連携サービスの移行](#)”で説明しています。

■ユーザデータ(スケルトンおよびユーザアプリケーションなど)

旧バージョン・レベル	本バージョン・レベルでの使用可否
V1.x	△ (注1) (注2)
V2.x ~ V6.0	△ (注1)
V7.0 以降	○

[○:互換あり。△:一部互換なし。×:互換なし。ー:定義が存在しない。]

注1)

Windows(R)用のC/C++ユーザアプリケーションは、Interstage Application Server V7.0以降ではバイナリ互換がありません。これは、CORBAサービスが提供しているライブラリのコンパイルオプションが変更されたためです。そのため、ユーザアプリケーションを移行する場合にはユーザアプリケーションのコンパイルオプションも変更してアプリケーションを再ビルドする必要があります。詳細は、“[Windows\(R\)のアプリケーションのコンパイルオプション変更について](#)”を参照してください。

注2)

スケルトン論理の変更によりINTERSTAGE Application Server V1.0L10で作成したスケルトンはINTERSTAGE Application Server V1.0L20以降では使用できません。

3.2.12 データベース連携サービスの資源

データベース連携サービスは、以下の製品で利用可能です。

- ・ Interstage Application Server Enterprise Edition
- ・ Interstage Application Server Standard-J Edition

■運用環境ファイル

C:\¥Interstage¥ots¥etc¥config

旧バージョン・レベル	本バージョン・レベルでの使用可否
V1.x 以降	○

[○:互換あり。△:一部互換なし。×:互換なし。ー:定義が存在しない。]

■ユーザデータ

データベース連携サービスのコマンドで採取した、C:\Interstage¥var配下のダンプファイルなど。

旧バージョン・レベル	本バージョン・レベルでの使用可否
V1.x ~ V6.0	○ (注1)
V7.0 以降	○

[○:互換あり。△:一部互換なし。×:互換なし。ー:定義が存在しない。]

注1)

Windows(R)用のC/C++ユーザアプリケーションは、Interstage Application Server V7.0以降ではバイナリ互換がありません。これは、CORBAサービスが提供しているライブラリのコンパイルオプションが変更されたためです。そのため、ユーザアプリケーションを移行する場合にはユーザアプリケーションのコンパイルオプションも変更してアプリケーションを再ビルドする必要があります。詳細は、“[Windows\(R\)のアプリケーションのコンパイルオプション変更について](#)”を参照してください。

3.2.13 Portable-ORBの資源

Portable-ORBは、以下の製品で利用可能です。

- Interstage Application Server Enterprise Edition
- Interstage Application Server Standard-J Edition

■環境定義ファイル

旧バージョン・レベル	本バージョン・レベルでの使用可否
V2.x ~ V5.x	○ (注1)
V6.0 以降	○

[○:互換あり。△:一部互換なし。×:互換なし。ー:定義が存在しない。]

注1)

Portable-ORB環境設定コマンド(porbeditenv)の定義項目の初期値が一部変更されています。

旧バージョン・レベルでのPortable-ORBの環境設定において、「最大コネクション数」を“257”以上と設定した場合、または「最大リクエスト数」を“4097”以上と設定した場合は、Portable-ORB環境設定コマンド(porbeditenv)を使用して再設定してください。なお、「最大コネクション数」を“256”以下、および「最大リクエスト数」を“4096”以下の場合は問題ありません。

Portable-ORB環境設定コマンドの詳細は、“リファレンスマニュアル(コマンド編)”の“porbeditenv”を参照してください。

3.2.14 イベントサービスの資源

イベントサービスは、以下の製品で利用可能です。

- Interstage Application Server Enterprise Edition
- Interstage Application Server Standard-J Edition

■環境定義ファイル

旧バージョン・レベル	本バージョン・レベルでの使用可否
V2.x ~ V3.x	○ (注1)
V4.x ~ V5.x	○ (注1) (注2) (注3)
V6.0 ~ V7.0	○ (注3)

旧バージョン・レベル	本バージョン・レベルでの使用可否
8.0以降	○ (注4)

[○:互換あり。△:一部互換なし。×:互換なし。―:定義が存在しない。]

注1)

Interstage Application Server V5.x以前において作成したイベントチャネルのイベントチャネルグループ名またはイベントチャネル名に以下の文字以外が含まれている場合、本バージョン・レベルにおいてInterstage管理コンソールを使用してイベントチャネルの一覧を表示すると、エラーメッセージ“es39996”または“es39999”が出力され操作できません。

	Interstage管理コンソールで使用可能な文字
イベントチャネルグループ名	Windows 英数字、イクスクラメーションマーク(!)、ナンバー(#)、ドルマーク(\$)、アポストロフィー(')、左丸括弧((), 右丸括弧()), プラス(+), ハイフン(-), ピリオド(.), スラッシュ(/), セミコロン(;), アットマーク(@), 左角括弧([), 右角括弧()], アンダースコア(_), 逆クォート(`), 左中括弧({), 右中括弧()}、波線(~)
	Solaris 英数字、イクスクラメーションマーク(!), プラス(+), ハイフン(-), ピリオド(.), スラッシュ(/), アットマーク(@), 左角括弧([), 右角括弧()], アンダースコア(_), 左中括弧({), 右中括弧()}, 波線(~)
	Linux 英数字、プラス(+), ハイフン(-), ピリオド(.), スラッシュ(/), アットマーク(@), 左角括弧([), 右角括弧()], キャロット(^), アンダースコア(_), 左中括弧({), 右中括弧()}
イベントチャネル名	Windows 英数字、イクスクラメーションマーク(!)、ナンバー(#)、ドルマーク(\$)、アポストロフィー(')、左丸括弧((), 右丸括弧()), アスタリスク(*), プラス(+), カンマ(,), ハイフン(-), ピリオド(.), スラッシュ(/), コロン(:), セミコロン(;), イコール(=), クエスチョンマーク(?), アットマーク(@), 左角括弧([), 右角括弧()], アンダースコア(_), 逆クォート(`), 左中括弧({), 右中括弧()}, 波線(~)
	Solaris 英数字、イクスクラメーションマーク(!), アスタリスク(*), プラス(+), カンマ(,), ハイフン(-), ピリオド(.), スラッシュ(/), コロン(:), イコール(=), クエスチョンマーク(?), アットマーク(@), 左角括弧([), 右角括弧()], アンダースコア(_), 左中括弧({), 右中括弧()}, 波線(~)
	Linux 英数字、パーセント(%), アスタリスク(*), プラス(+), カンマ(,), ハイフン(-), ピリオド(.), スラッシュ(/), コロン(:), イコール(=), クエスチョンマーク(?), アットマーク(@), 左角括弧([), 右角括弧()], キャロット(^), アンダースコア(_), 左中括弧({), 右中括弧()}

上記の文字以外が設定されたイベントチャネルを運用する場合は、イベントサービス運用コマンドを使用するか、またはInterstage管理コンソールを使用する前にイベントサービス運用コマンドを使用して、該当するイベントチャネルを削除してください。

注2)

Interstage Application Server V4.x/V5.xで作成したユニットの格納ディレクトリに以下の文字が含まれている場合、本バージョン・レベルにおいてInterstage管理コンソールを使用して保存先の一覧を表示すると、エラーメッセージ“es39996”が出力され操作できません。

― アンパサンド(&)

ー 小なり(<)

上記の使用禁止文字が設定されたユニットを運用する場合は、イベントサービス運用コマンドを使用するか、またはInterstage管理コンソールを使用する前にイベントサービス運用コマンドを使用して、該当するユニットを削除してください。

注3)

本バージョン・レベルでは、Point-To-Pointモデルでも“コンシューマ未接続時のエラー復帰モード”(esetcnfコマンドおよびesetcnfchnlコマンドの-s -chkcon yes)が有効となります。

Interstage Application Server V7.0以前の環境において、MultiCastモデルとPoint-To-Pointモデルを混在して使用し、かつ“コンシューマ未接続時のエラー復帰モード”をシステム単位で有効としている(esetcnfコマンドで設定している)場合、Interstage Application Server V7.0以前に開発したサプライヤのアプリケーションでは、当該機能使用時の例外をハンドリングしていないため、エラーが発生する可能性があります。

上記の環境で“コンシューマ未接続時のエラー復帰モード”を使用するイベントチャネルを追加する場合は、esetcnfchnlコマンドを使用して当該イベントチャネルにだけ有効としてください。

注4)

Interstage Application Server V6.0以降では、イベントサービス運用コマンド使用時の省略値が以下のように変更になりました。本バージョン・レベルにおいて新規にイベントサービスのセットアップを行い、Interstage Application Server V5.x以前と同様の運用を行う場合は、省略値に注意して設定してください。イベントサービス運用コマンドの設定値の詳細は、“リファレンスマニュアル(コマンド編)”を参照してください。

[変更内容]

	V5.x以前の場合	V6.0以降の場合
esetcnfコマンドの-s -ltime	300	0
esetcnfコマンドの-s -sthmax	8	100
esetcnfコマンドの-s -conenum	50	100
esetcnfコマンドの-s -supenum	50	100
esetcnfコマンドの-s -ltrntime	604800	300
esetcnfコマンドの-s -pltime	604800	0
esmkunitコマンドのユニット定義ファイルの項目shmmax	8	80

■ユーザアプリケーション

イベントサービスのユーザアプリケーションは、CORBAサービスのユーザアプリケーションに依存している箇所があります。詳細については、CORBAサービスの“[ユーザアプリケーション](#)”を参照してください。なお、イベントサービスでは、OOCOBOLアプリケーションはサポートしていません。

第4章 旧機能から新機能への移行方法

4.1 Interstage HTTP Server(Apache HTTP Server 2.0ベース)への移行

本製品では、V9.0以降、Interstage HTTP Server(Apache HTTP Server 2.0ベース)を提供します。

8.0以前のInterstage HTTP Server(Apache HTTP Server 1.3ベース)、およびV7.0以前のInfoProvider Proは提供されていないため、以下を参照し、Interstage HTTP Server(Apache HTTP Server 2.0ベース)へ移行してください。

4.1.1 8.0以前のInterstage HTTP Server(Apache HTTP Server 1.3ベース)からの移行

8.0以前のInterstage HTTP Server(Apache HTTP Server 1.3ベース)からの移行方法について説明します。

- [8.0/V7.0からの移行手順](#)
- [V6.0以前からの移行手順](#)
- [機能差異](#)
- [メッセージ](#)

■8.0/V7.0からの移行手順

ihrestoreコマンド、またはメンテナンスの一括実行のバッチファイル(Windows(R))/シェルスクリプト(Solaris/Linux)を使用して移行します。詳細については、“運用ガイド(基本編)”の“メンテナンス(資源のバックアップ)”を参照し、移行してください。

■V6.0以前からの移行手順

メンテナンスの一括実行のバッチファイル(Windows(R))/シェルスクリプト(Solaris/Linux)を使用して移行できますが、ihrestoreコマンドは使用できません。メンテナンスの一括実行の詳細については、“運用ガイド(基本編)”の“メンテナンス(資源のバックアップ)”を参照し、移行してください。メンテナンスの一括実行以外の方法で移行する場合は、“[機能差異](#)”を参照し、本製品のインストール後、環境定義ファイル(httpd.conf)を直接編集してください。

■機能差異

◆追加機能

Interstage HTTP Server(Apache HTTP Server 2.0ベース)では、8.0以前のInterstage HTTP Server(Apache HTTP Server 1.3ベース)の機能に加え、以下の機能を提供しています。機能内容の詳細については、“Interstage HTTP Server 運用ガイド”を参照してください。

- 複数Webサーバ運用
- コンテンツの圧縮機能
- IPv6による通信
- ihsstartコマンドによる起動
- ihsstopコマンドによる停止
- トレースログ

◆環境定義ファイルの格納先

環境定義ファイル(httpd.conf)の格納先について、以下の変更があります。

Windows (インストールパスはデフォルト)

【8.0以前の格納先】 C:¥Interstage¥F3FMihs¥conf¥httpd.conf 【V9.0以降の格納先】 C:¥Interstage¥F3FMihs¥servers¥FJapache¥conf¥httpd.conf
--

Solaris (インストールパスはデフォルト)

【8.0以前の格納先】 /etc/opt/FJSVihs/conf/httpd.conf 【V9.0以降の格納先】 /var/opt/FJSVihs/servers/FJapache/conf/httpd.conf
--

Linux

【8.0以前の格納先】 /etc/opt/FJSVihs/conf/httpd.conf 【V9.0以降の格納先】 /var/opt/FJSVihs/servers/FJapache/conf/httpd.conf
--

◆アクセスログ／エラーログの格納先

アクセスログ／エラーログの格納先について、以下の変更があります。

Windows (インストールパスはデフォルト)

【8.0以前の格納先】 C:¥Interstage¥F3FMihs¥logs 【V9.0以降の格納先】 C:¥Interstage¥F3FMihs¥servers¥FJapache¥logs
--

Solaris (インストールパスはデフォルト)

【8.0以前の格納先】 /var/opt/FJSVihs/logs 【V9.0以降の格納先】 /var/opt/FJSVihs/servers/FJapache/logs
--

Linux

【8.0以前の格納先】 /var/opt/FJSVihs/logs 【V9.0以降の格納先】 /var/opt/FJSVihs/servers/FJapache/logs
--

◆環境定義ファイル(httpd.conf)の設定内容

環境定義ファイル(httpd.conf)の定義について、以下の変更があります。変更内容の詳細については、“Interstage HTTP Server運用ガイド”の“ディレクトリ一覧”を参照してください。

定義項目	8.0以前のディレクティブ名	V9.0以降のディレクティブ名	変更内容
モジュールのロード	AddModule ClearModuleList LoadModule	LoadModule	AddModuleディレクティブおよびClearModuleディレクティブは、削除されました。LoadModuleディレクティブだけで設定してください。 また、LoadModuleディレクティブは、ServerRootディレクティブからの相対パスは指定不可となりました。絶対パスで指定してください。
キャッシュドキュメントの有効期限	CacheMaxExpire	CacheMaxExpire	ディレクティブ名の変更はありません。ただし、有効期限の指定が、時間単位から秒単位に変更になります。
キャッシュドキュメントの最大サイズ	CacheSize	-	CacheSizeディレクティブは、削除されました。キャッシュドキュメントの最大サイズは、設定できません。
アクセスログファイル	CustomLog	CustomLog	ディレクティブ名の変更はありません。ただし、初期値が変更され、V8以前のログ情報に加え、リクエスト処理時間およびリクエスト受信情報を記録します。 Windows ihsrlogコマンド実行文の指定方法に変更があります。 アクセスログファイル 、 エラーログファイル を参照してください。
Windows アクセスログファイル、エラーログファイル	CustomLog	CustomLog	ディレクティブ名の変更はありません。ただし、ihsrlogコマンド実行文の指定方法に変更があります。ihsrlogコマンド実行文を指定している場合は、以下のように拡張子“.exe”を付けて指定してください。 「CustomLog " ihsrlog.exe -s logs/accesslog 1 5" common」
	ErrorLog	ErrorLog	
	TransferLog	TransferLog	
ステータスコードに指定するエラードキュメント	ErrorDocument	ErrorDocument	ディレクティブ名の変更はありません。ただし、出力するテキストメッセージを指定する場合、テキストメッセージの先頭にダブルクォーテーション(")を付加して指定する形式から、ダブルクォーテーション(")でテキストメッセージを囲んで指定する形式に変更になります。
Solaris Linux グループID	Group	Group	ディレクティブ名の変更はありません。ただし、省略値が“#-1”から“nobody”に変更になります。
HTTPレスポンスヘッダ	Header	Header	ディレクティブ名の変更はありません。ただし、形式が以下に変更され、echoオプションが追加になります。 「Header set append add echo ヘッダ名 値」
Windows 接続待ちキューの最大数	IHSAcceptQueueSize	-	IHSAcceptQueueSizeディレクティブは、削除されました。接続待ちキューの最大数は、設定できません。
Serverヘッダフィールドの送信	IHSServerHeader	IHSServerHeader	ディレクティブ名の変更はありません。ただし、初期値が未設定から“Off”に変更になり、Serverヘッダフィールドがクライアントに送信されないように変更になります。
HTTPリクエストヘッダの1フィールドの最大サイズ	LimitRequestField size	LimitRequestField size	ディレクティブ名の変更はありません。ただし、HTTPリクエストヘッダの最大サイズの上限値が“8190”から“2147483647”に変更になります。
HTTPメソッドが含まれる行の最大サイズ	LimitRequestLine	LimitRequestLine	ディレクティブ名の変更はありません。ただし、HTTPメソッドが含まれる行の最大サイズの上限値が“8190”から“2147483647”に変更になります。
ログフォーマット	LogFormat	LogFormat	ディレクティブ名の変更はありません。ただし、“レスポンス送信終了時の接続ステータス”を指定する場合のパラメタが“%c”から“%X”に変更になります。

定義項目	8.0以前のディレクティブ名	V9.0以降のディレクティブ名	変更内容
Solaris Linux クライアントの同時接続数	MaxClients	MaxClients	ディレクティブ名の変更はありません。ただし、クライアントの同時接続数の上限値が“4096”からServerLimitディレクティブの設定値までに変更になります。また、省略値が“4096”から“256”に変更になります。
Solaris Linux 1つの通信プロセスが処理できるHTTPリクエスト数	MaxRequestsPerChild	MaxRequestsPerChild	ディレクティブ名の変更はありません。ただし、Solaris/Linuxの場合は、1つの通信プロセスが処理できるHTTPリクエスト数の省略値が“0”から“10000”に変更になります。なお、Windows(R)の場合、省略値の変更はありません。
Solaris Linux 待機状態の通信プロセスの最大数	MaxSpareServers	MaxSpareServers	ディレクティブ名の変更はありません。ただし、通信プロセスの最大数の上限値が“4096”からServerLimitディレクティブの設定値までに変更になります。
Solaris Linux 待機状態の通信プロセスの最小数	MinSpareServers	MinSpareServers	ディレクティブ名の変更はありません。ただし、通信プロセスの最小数の上限値が“4096”からServerLimitディレクティブの設定値までに変更になります。
ポート番号	Port	Listen	Portディレクティブは、削除されました。Listenディレクティブでポート番号を設定してください。なお、Listenディレクティブは複数設定できます。
TRACEメソッドの抑止	RewriteCond RewriteRule RewriteEngine	TraceEnable	TRACEメソッドの抑止は、TraceEnableディレクティブで設定します。TraceEnableディレクティブの初期値は“Off”であり、TRACEメソッドを抑止する設定に変更になります。
動作状態を記録するファイル	ScoreBoardFile	ScoreBoardFile	ディレクティブ名の変更はありません。ただし、動作状態をメモリ上に記録(メモリの獲得の失敗時のみファイルに記録)していましたが、動作状態をファイルに記録するように変更になります。メモリ上に記録したい場合は、本ディレクティブを設定しないでください。
Webサーバの起動タイプ	ServerType	-	ServerTypeディレクティブは、削除されました。Webサーバの起動タイプは、設定できません。
SSLプロトコルバージョン	SSLVersion	SSLVersion	ディレクティブ名の変更はありません。ただし、SSLプロトコルバージョン“SSL3.1”(TLS 1.0)の指定が可能となります。
Solaris Linux 待機状態の通信プロセス数	StartServers	StartServers	ディレクティブ名の変更はありません。ただし、通信プロセス数の上限値が“4096”からServerLimitディレクティブの設定値までに変更になります。
Windows クライアント数	ThreadsPerChild	ThreadsPerChild	ディレクティブ名の変更はありません。ただし、クライアントの同時接続数の上限値がThreadLimitディレクティブの設定値までに変更になります。また、省略値が“40”から“64”に変更になります。
クライアント送受信タイムアウト時間	Timeout	Timeout	ディレクティブ名の変更はありません。ただし、0を指定した場合、タイムアウト時間は無制限でしたが、0秒に変更になります。

定義項目	8.0以前のディレクティブ名	V9.0以降のディレクティブ名	変更内容
Solaris Linux ユーザID	User	User	ディレクティブ名の変更はありません。ただし、省略値が“#-1”から“nobody”に変更になります。



注意

- Interstage HTTP Server (Apache HTTP Server 2.0ベース)では、8.0以前のInterstage HTTP Server (Apache HTTP Server 1.3ベース)上で運用していたプラグインモジュールは使用できません。Interstage HTTP Server (Apache HTTP Server 2.0ベース)に対応したプラグインモジュールを使用して運用してください。
- Interstage HTTP Server (Apache HTTP Server 2.0ベース)では、以下の環境定義ファイルは提供されません。環境定義ファイル (httpd.conf) を直接編集してください。

Windows

C:\Interstage\F3FMihs\conf\srm.conf
C:\Interstage\F3FMihs\conf\access.conf

Solaris

Linux

/etc/opt/FJSVihs/conf/srm.conf
/etc/opt/FJSVihs/conf/access.conf

◆アクセスログの出力形式

アクセスログの出力形式について、以下の変更があります。出力形式に対応する各出力項目の詳細については、“Interstage HTTP Server運用ガイド”の“運用・保守”－“ログ”－“アクセスログ”を参照してください。

【8.0以前のアクセスログの出力形式】

ホスト名 ユーザ名識別 ユーザ名 日時 "リクエスト" ステータスコード データ転送量

【V9.0以降のアクセスログの出力形式】

ホスト名 ユーザ名識別 ユーザ名 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

◆CGIプログラムの実行方法

8.0以前は、初期設定のままでCGIプログラムを実行できましたが、V9.0以降では、CGIプログラム実行を許可するように、環境定義ファイル (httpd.conf) の定義を編集する必要があります。

CGIプログラム実行を許可する場合は、環境定義ファイル (httpd.conf) において、コメント行として定義されている以下のディレクティブのハッシュマーク(#)を取り除いてください。なお、CGIプログラム実行の許可設定の詳細については、“Interstage HTTP Server運用ガイド”の“環境設定”－“環境定義ファイル”－“CGIプログラム実行の許可”を参照してください。

Windows

【修正前】

```
#LoadModule cgi_module "C:/Interstage/F3FMihs/modules/mod_cgi.so"
```

【修正後】

```
LoadModule cgi_module "C:/Interstage/F3FMihs/modules/mod_cgi.so"
```

Solaris

Linux

【修正前】

```
#LoadModule cgi_module "/opt/FJSVihs/modules/mod_cgi.so"
#ScriptAlias /cgi-bin/ "/opt/FJSVihs/servers/FJapache/cgi-bin/"
```

【修正後】

```
LoadModule cgi_module "/opt/FJSVihs/modules/mod_cgi.so"
ScriptAlias /cgi-bin/ "/opt/FJSVihs/servers/FJapache/cgi-bin/"
```

■メッセージ

メッセージ番号がihsで始まるメッセージは、Interstage HTTP ServerがApache HTTP Server 1.3ベースからApache HTTP Server 2.0ベースのWebサーバへ変更されたことに伴い、メッセージの出力内容が変更されます。

メッセージ番号がihsで始まるメッセージの変更内容について以下に示します。メッセージの詳細については、“メッセージ集”を参照してください。

旧メッセージID	新メッセージID	変更内容
ihs00001 ihs00002 ihs81001 ihs81002 ihs81101 ihs81102 ihs81206～ihs81209 ihs81211 ihs81213～ihs81218 ihs81223 ihs81224 ihs81301 ihs81302 ihs81304 ihs81305 ihs81308 ihs81313～ihs81315 ihs81320 ihs81321 ihs81323～ihs81328 ihs81332～ihs81339 ihs81501～ihs81505 ihs81517	変更なし	可変情報に、Webサーバ名を追加しました。
ihs10000 ihs10001 ihs10011～ihs10014 ihs10017 ihs20414 ihs20618 ihs20619 ihs20621 ihs20657 ihs20658 ihs20663 ihs20693 ihs20708 ihs20806 ihs21202 ihs65000 ihs65001 ihs66022 ihs66027	—	本メッセージは、出力されません。

旧メッセージID	新メッセージID	変更内容
ihs81307 ihs81434		
ihs10002	ihs00007	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“Out of memory in ap_setup_prelinked_modules()!” 新:“Ouch! Out of memory in ap_setup_prelinked_modules()!”
ihs10003	ihs00010	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“Syntax error in -C/-c directive: %s1” 新:“Syntax error in -C/-c directive: %s1 [%s2]”
ihs10005	ihs00011	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“fopen: %s1: %s2: Could not open document config file %s3” 新:“%s1: could not open document config file %s2 [%s3]”
ihs10006	ihs00012	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“Syntax error on line %s1 of %s2: %s3” 新:“Syntax error on line %s1 of %s2: %s3 [%s4]”
ihs10007	ihs00021	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“ap_spawn_child: %s1: Couldn't fork child for ErrorLog process” 新:“(%s1)%s2: Couldn't start ErrorLog process [%s3]”
ihs10008	ihs00023	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“fopen: %s1: %s2: Could not open error log file %s3.” 新:“(%s1)%s2: %s3: could not open error log file %s4. [%s5]”
ihs10009	ihs00028	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“fopen: %s1: %s2: Could not log pid to file %s3” 新:“(%s1)%s2: %s3: could not log pid to file %s4 [%s5]”
ihs10010	ihs00207	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“fork: %s1: %s2: Unable to fork new process” 新:“(%s1)%s2: apr_proc_detach failed [%s3]”
ihs10015	ihs00223	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“%s1: Bad user name %s2” 新:“%s1: bad user name %s2 [%s3]”
ihs10016	ihs00224	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“%s1: Bad group name %s2” 新:“%s1: bad group name %s2 [%s3]”
ihs20303	エラーログ (ログレベル: alert)	出力先をエラーログに変更しました。メッセージの内容は、変更ありません。
ihs20304	エラーログ (ログレベル: crit)	出力先をエラーログに変更しました。メッセージの内容は、変更ありません。
ihs20501	エラーログ (ログレベル: crit)	出力先をエラーログに変更しました。メッセージの内容は、変更ありません。
ihs20502	エラーログ (ログレベル: crit)	出力先をエラーログに変更しました。メッセージの内容は、変更ありません。
ihs20602	ihs00225	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“(%s1)%s2: fcntl: F_SETLKW: Error getting accept lock, exiting! Perhaps you need to use the LockFile directive to place your lock file on a local disk!” 新:“(%s1)%s2: couldn't grab the accept mutex [%s3]”
ihs20603	ihs00226	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“(%s1)%s2: fcntl: F_SETLKW: Error freeing accept lock, exiting! Perhaps you need to use the LockFile directive to place your lock file on a

旧メッセージID	新メッセージID	変更内容
		local disk!” 新:“(s1)s2: couldn't release the accept mutex [%s3]”
ihs20631	エラーログ (ログレベル: emerg)	出力先をエラーログに変更しました。メッセージの内容は、変更ありません。
ihs20632	エラーログ (ログレベル: emerg)	出力先をエラーログに変更しました。メッセージの内容は、変更ありません。
ihs20645	ihs00215	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“(s1)s2: getpwuid: Couldn't determine user name from uid %s3, you probably need to modify the User directive” 新:“(s1)s2: getpwuid: couldn't determine user name from uid %s3, you probably need to modify the User directive [%s4]”
ihs20646	ihs00216	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“(s1)s2: setgid: Unable to set group id to Group %s3” 新:“(s1)s2: setgid: unable to set group id to Group %s3 [%s4]”
ihs20647	ihs00217	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“(s1)s2: initgroups: Unable to set groups for User %s3 and Group %s4” 新:“(s1)s2: initgroups: unable to set groups for User %s3 and Group %s4 [%s5]”
ihs20650	ihs00015	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“(s1)s2: make_sock: Failed to get a socket for %s3” 新:“(s1)s2: alloc_listener: failed to get a socket for %s3 [%s4]”
ihs20651	ihs00016	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“(s1)s2: make_sock: for %s3, setsockopt: (SO_REUSEADDR)” 新:“(s1)s2: make_sock: for address %s3, apr_socket_opt_set: (%s4) [%s5]”
ihs20652	ihs00016	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“(s1)s2: make_sock: for %s3, setsockopt: (SO_KEEPALIVE)” 新:“(s1)s2: make_sock: for address %s3, apr_socket_opt_set: (%s4) [%s5]”
ihs20654	ihs00018	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“(s1)s2: make_sock: Could not bind to %s3” 新:“(s1)s2: make_sock: could not bind to address %s3 [%s4]”
ihs20655	ihs00019	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“(s1)s2: make_sock: Unable to listen for connections on %s3” 新:“(s1)s2: make_sock: unable to listen for connections on address %s3 [%s4]”
ihs20656	ihs00227	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧:“(s1)s2: A listening socket could not be made non-blocking.” 新:“(s1)s2: ap_listen_open: unable to make socket non-blocking [%s3]”
ihs20661	ihs00214	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“(s1)s2: setuid: Unable to change to uid: %s3” 新:“(s1)s2: setuid: unable to change to uid: %s3 [%s4]”
ihs20662	エラーログ (ログレベル:error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧:“(s1)s2: select: (listen)” 新:“(s1)s2: apr_poll: (listen)”

旧メッセージID	新メッセージID	変更内容
ihs20676	ihs00206	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧: “Child %s1 returned a Fatal error... HTTP Server is exiting!” 新: “Child %s1 returned a Fatal error... Apache is exiting! [%s2]”
ihs20696	エラーログ (ログレベル: crit)	出力先をエラーログに変更しました。メッセージの内容は、変更ありません。
ihs20698	エラーログ (ログレベル: crit)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧: “(%s1)%s2: setup_inherited_listeners: WSASocket failed to open the inherited socket.” 新: “(%s1)%s2: Child %s3: setup_inherited_listeners(), WSASocket failed to open the inherited socket.”
ihs20704	ihs00020	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧: “No sockets were created for listening” 新: “no listening sockets available, shutting down [%s1]”
ihs20709	ihs00112	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧: “(%s1)%s2: Parent: GetModuleFileName() returned NULL for current process.” 新: “(%s1)%s2: GetModuleFileName failed [%s3]”
ihs20710	ihs00125	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧: “(%s1)%s2: Parent: Could not create exit event for child process” 新: “(%s1)%s2: Parent: Could not create %s3 event for child process [%s4]”
ihs20711	ihs00124	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧: “(%s1)%s2: Parent: Unable to create pipe to child process.” 新: “(%s1)%s2: Parent: Unable to %s3 [%s4]”
ihs20712	ihs00124	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧: “(%s1)%s2: Parent: Unable to create null output pipe for child process.” 新: “(%s1)%s2: Parent: Unable to %s3 [%s4]”
ihs20713	ihs00124	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧: “(%s1)%s2: Parent: Unable to create null error pipe for child process.” 新: “(%s1)%s2: Parent: Unable to %s3 [%s4]”
ihs20714	ihs00126	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧: “(%s1)%s2: Parent: Not able to create the child process.” 新: “(%s1)%s2: Parent: Failed to create the child process. [%s3]”
ihs20717	ihs00127	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧: “(%s1)%s2: Parent: WSADuplicateSocket failed for socket %s3.” 新: “(%s1)%s2: Parent: WSADuplicateSocket failed for socket %s3. Check the FAQ. [%s4]”
ihs20718	ihs00124	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧: “(%s1)%s2: Parent: Unable to write duplicated socket %s3 to the child.” 新: “(%s1)%s2: Parent: Unable to %s3 [%s4]”
ihs20719	ihs00104	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧: “(%s1)%s2: master_main: Cannot create shutdown event %s3” 新: “(%s1)%s2: Parent: Cannot create shutdown event %s3 [%s4]”
ihs20720	ihs00105	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧: “(%s1)%s2: master_main: Cannot create restart event %s3” 新: “(%s1)%s2: Parent: Cannot create restart event %s3 [%s4]”
ihs20722	ihs00108	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧: “(%s1)%s2: master_main: : WaitForMultipleObjects on process handles and HTTP Server-signal -- doing shutdown”

旧メッセージID	新メッセージID	変更内容
		新:“(%)%s2: master_main: WaitForMultipleObjects WAIT_FAILED --doing server shutdown [%s3]”
ih20809	エラーログ (ログレベル: crit)	出力先をエラーログに変更しました。メッセージの内容は、変更ありません。
ih21001	エラーログ (ログレベル: crit)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧:“(%)%s2: bind: rfc1413: Error binding to local port” 新:“(%)%s2: rfc1413: Error binding query socket to local port”
ih21002	エラーログ (ログレベル: crit)	出力先をエラーログに変更しました。メッセージの内容は、変更ありません。
ih21003	エラーログ (ログレベル: crit)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧:“(%)%s2: read: rfc1413: Error reading response” 新:“(%)%s2: read: rfc1413: error reading response”
ih21004	エラーログ (ログレベル: crit)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧:“(%)%s2: socket: rfc1413: Error creating socket” 新:“(%)%s2: rfc1413: error creating query socket”
ih21111	エラーログ (ログレベル: alert)	出力先をエラーログに変更しました。メッセージの内容は、変更ありません。
ih21201	ih00028	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“open: %s1: Could not open pid file %s2: %s3” 新:“(%)%s2: %s3: could not log pid to file %s4 [%s5]”
ih33401	ih00041	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“gethostname: mod_unique_id requires the hostname of the server” 新:“(%)%s2: mod_unique_id: unable to find hostname of the server [%s3]”
ih33402	エラーログ (ログレベル: alert)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧:“mod_unique_id: Unable to gethostbyname(“%s1”)” 新:“(%)%s2: mod_unique_id: unable to find hostname of the server”
ih33404	エラーログ (ログレベル: crit)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧:“Pids are greater than 32-bits! I'm broken!” 新:“oh no! pids are greater than 32-bits! I'm broken!”
ih61000	ih00500	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“(%)%s2: %s3. httpd-ssl: dlopen(SMEE) error:%s4” 新:“SSL: Library initialization failed. (%s1,%s2) [%s3]”
ih62004	ih00524 ih00525 ih00526 ih00527 ih00528 ih00529	メッセージIDを変更し、メッセージの内容を以下に変更しました。また、エラー発生条件に応じてメッセージが分かれました。 旧:“(%)%s1,%s2,%s3” 新(ih00524):“SSL: SSLSlotDir is not defined. [%s1]” 新(ih00525):“SSL: SSLTokenLabel is not defined. [%s1]” 新(ih00526):“SSL: SSLUserPINFile is not defined. [%s1]” 新(ih00527):“SSL: SSLVerifyClient is not defined. [%s1]” 新(ih00528):“SSL: SSLEnvDir is not defined. [%s1]” 新(ih00529):“SSL: SSLCertName is not defined. [%s1]”
ih65100	エラーログ (ログレベル: error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧:“SSL: SCMI: Not enough memory(%s1)” 新:“[client %s1]SSL: SCMI Insufficient memory”
ih65101	エラーログ (ログレベル: error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧:“SSL: SCMI_Login failed. Detail(%s1)(%s2)” 新:“[client %s1]SSL: SCMI_Login failed.(%s3,...%n)”

旧メッセージID	新メッセージID	変更内容
ihs65102	エラーログ (ログレベル:error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧:“SSL: SCMI_Logout failed. Detail(%s1)” 新:“[client %s1]SSL: SCMI_Logout failed.(%s3,...%n)”
ihs65103	エラーログ (ログレベル:error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧:“SSL: SCMI_Expand failed. Detail(%s1)(%s2)” 新:“[client %s1]SSL: SCMI_Expand failed.(%s3,...%n)”
ihs66000	ihs00502	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“SSL: slot information directory(%s1) error (%s2,%s3,%s4)” 新:“SSL: Slot information directory error.(%s1,%s2,%s3) [%s4]”
ihs66001	ihs00503	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“SSL: token label(%s1) error (%s2,%s3,%s4)” 新:“SSL: Invalid Token label.(%s1,%s2,%s3) [%s4]”
ihs66002	ihs00501	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“SSL: not enough memory (%s1,%s2,%s3)(%s4)” 新:“SSL: Insufficient memory.(%s1,%s2,%s3,%s4) [%s5]”
ihs66003	ihs00504	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“SSL: site certificate(%s1) expired (%s2,%s3,%s4)(%s5)” 新:“SSL: The site certificate has expired.(%s1,%s2,%s3,%s4) [%s5]”
ihs66004	ihs00505	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“SSL: CA certificate expired (%s1,%s2,%s3)” 新:“SSL: The CA certificate has expired.(%s1,%s2,%s3,%s4) [%s5]”
ihs66005	ihs00506 ihs00507	メッセージIDを変更し、メッセージの内容を以下に変更しました。また、エラーがサイト証明書またはCA証明書のどちらで発生したかに応じてメッセージが分かれました。 旧:“SSL: couldn't verify certificate(%s1) (%s2,%s3,%s4)” 新(ihs00506):“SSL: Couldn't verify the site certificate. (%s1,%s2,%s3,%s4) [%s5]” 新(ihs00507):“SSL: Couldn't verify the CA certificate. (%s1,%s2,%s3,%s4) [%s5]”
ihs66006	ihs00510	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“SSL: revoked site certificate(%s1) (%s2,%s3,%s4)” 新:“SSL: The site certificate has been revoked.(%s1,%s2,%s3,%s4) [%s5]”
ihs66007	ihs00518	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“SSL: private key not exist (%s1) (%s2,%s3,%s4)” 新:“SSL: Private key does not exist.(%s1,%s2,%s3,%s4) [%s5]”
ihs66008	ihs00512	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“SSL: site certificate nickname error (%s1,%s2,%s3)” 新:“SSL: Invalid Site certificate nickname.(%s1,%s2,%s3,%s4) [%s5]”
ihs66009	ihs00513	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“SSL: client CA certificate nickname error (%s1,%s2,%s3)” 新:“SSL: Invalid Client CA certificate nickname.(%s1,%s2,%s3,%s4) [%s5]”
ihs66010	ihs00514	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“SSL: site certificate(%s1) has no path list (%s2,%s3,%s4)” 新:“SSL: The site certificate has no path list.(%s1,%s2,%s3,%s4) [%s5]”
ihs66011	ihs00515	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“SSL: CA certificate has no path list (%s1,%s2,%s3)” 新:“SSL: The CA certificate has no path list.(%s1,%s2,%s3,%s4) [%s5]”

旧メッセージID	新メッセージID	変更内容
ihs66012	ihs00516	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“SSL: site certificate(%s1) path error (%s2,%s3,%s4)” 新:“SSL: The site certificate path is invalid.(%s1,%s2,%s3,%s4) [%s5]”
ihs66013	ihs00517	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“SSL: CA certificate path error (%s1,%s2,%s3)” 新:“SSL: The CA certificate is invalid.(%s1,%s2,%s3,%s4) [%s5]”
ihs66014	ihs00511	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“SSL: revoked CA certificate (%s1,%s2,%s3)” 新:“SSL: The CA certificate is invalid.(%s1,%s2,%s3,%s4) [%s5]”
ihs66015	ihs00519	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“SSL: user pin(%s1) error (%s2,%s3,%s4)” 新:“SSL: Invalid User pin.(%s1,%s2,%s3,%s4) [%s5]”
ihs66016	ihs00520	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“SSL: certificate environment directory(%s1) error (%s2,%s3,%s4)” 新:“SSL: Certificate environment directory error.(%s1,%s2,%s3,%s4) [%s5]”
ihs66017	ihs00505	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“SSL: client CA certificate expired (%s1,%s2,%s3)(%s4)” 新:“SSL: The CA certificate has expired.(%s1,%s2,%s3,%s4) [%s5]”
ihs66018	エラーログ (ログレベル:error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧:“SSL: client CA certificate has no path list (%s1,%s2,%s3)(%s4)” 新:“[client %s1] SSL: The client CA certificate has no path list. (%s2,%s3,%s4)”
ihs66019	エラーログ (ログレベル:error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧:“SSL: client CA certificate path error (%s1,%s2,%s3)(%s4)” 新:“[client %s1] SSL: The client CA certificate path is invalid. (%s2,%s3,%s4)”
ihs66020	ihs00507	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“SSL: couldn't verify client CA certificate (%s1,%s2,%s3)(%s4)” 新:“SSL: Couldn't verify the CA certificate.(%s1,%s2,%s3,%s4) [%s5]”
ihs66021	エラーログ (ログレベル:error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧:“SSL: revoked client CA certificate (%s1,%s2,%s3)(%s4)” 新:“[client %s1] SSL: The client CA certificate has been revoked. (%s2,%s3,%s4)”
ihs66023	エラーログ (ログレベル:error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧:“SSL: socket access error (socket,%s1) (%s2,%s3,%s4)” 新:“[client %s1] SSL: Socket access error(%s2).(%s3,%s4,%s5)”
ihs66028	ihs00508	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“SSL: couldn't analyze site certificate(%s1) (%s2,%s3,%s4)” 新:“SSL: Couldn't analyze the site certificate.(%s1,%s2,%s3,%s4) [%s5]”
ihs66029	エラーログ (ログレベル:error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧:“SSL: couldn't analyze client CA certificate (%s1,%s2,%s3)(%s4)” 新:“[client %s1] SSL: Couldn't analyze the client CA certificate. (%s2,%s3,%s4)”
ihs66030	ihs00509	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧:“SSL: couldn't analyze CA certificate (%s1,%s2,%s3)” 新:“SSL: Couldn't analyze the CA certificate.(%s1,%s2,%s3,%s4) [%s5]”
ihs66031	エラーログ (ログレベル:error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧:“SSL: revoked client certificate (%s1,%s2,%s3)(%s4)”

旧メッセージID	新メッセージID	変更内容
		新: “[client %s1] SSL: The client CA certificate has been revoked. (%s2,%s3,%s4)”
ihs66032	エラーログ (ログレベル: error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧: “SSL: client certificate expired (%s1,%s2,%s3)(%s4)” 新: “[client %s1] SSL: The client certificate has expired.(%s2,%s3,%s4)”
ihs66033	エラーログ (ログレベル: error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧: “SSL: couldn't verify client certificate (%s1,%s2,%s3)(%s4)” 新: “[client %s1] SSL: Couldn't verify the client certificate. (%s2,%s3,%s4)”
ihs66034	エラーログ (ログレベル: error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧: “SSL: client certificate has no path list (%s1,%s2,%s3)(%s4)” 新: “[client %s1] SSL: The client certificate has no path list. (%s2,%s3,%s4)”
ihs66035	エラーログ (ログレベル: error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧: “SSL: client certificate path error (%s1,%s2,%s3)(%s4)” 新: “[client %s1] SSL: The client certificate path is invalid. (%s2,%s3,%s4)”
ihs66036	エラーログ (ログレベル: error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧: “SSL: couldn't analyze client certificate (%s1,%s2,%s3)(%s4)” 新: “[client %s1] SSL: Couldn't analyze the client certificate. (%s2,%s3,%s4)”
ihs66037	エラーログ (ログレベル: error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧: “SSL: version error (client %s1.%s2) (%s3,%s4,%s5)(%s6)” 新: “[client %s1] SSL: Version negotiation error.(%s2,%s3,%s4)”
ihs66038	エラーログ (ログレベル: error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧: “SSL: chipher handshake error (%s1,%s2,%s3)(%s4)” 新: “[client %s1] SSL: Cipher handshake error.(%s2,%s3,%s4)”
ihs66039	エラーログ (ログレベル: error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧: “SSL: unacceptable client certificate (%s1,%s2,%s3)(%s4)” 新: “[client %s1] SSL: Unacceptable client certificate.(%s2,%s3,%s4)”
ihs66040	エラーログ (ログレベル: error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧: “SSL: unsupported client certificate (%s1,%s2,%s3)(%s4)” 新: “[client %s1] SSL: Certificate is invalid.(%s2,%s3,%s4)”
ihs66043	エラーログ (ログレベル: error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧: “SSL: no client certificate (%s1,%s2,%s3)(%s4)” 新: “[client %s1] SSL: No client certificate.(%s2,%s3,%s4)”
ihs66044	エラーログ (ログレベル: error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧: “SSL: authority error (%s1,%s2,%s3)(%s4)” 新: “[client %s1] SSL: Authority error.(%s2,%s3,%s4)”
ihs66045	ihs00521	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧: “Errors are in description of SSLCipherSuite in a Configuration file.” 新: “SSL: Invalid cipher suite.(%s1,%s2,%s3,%s4) [%s5]”
ihs66200	エラーログ (ログレベル: error)	メッセージの内容を以下に変更し、出力先をエラーログに変更しました。 旧: “Failed to encode (obj=%s1, func=%s2, detail=%s3)(%s4)” 新: “[client %s1]SSL: apr_base64_encode error.(%s2,%s3)” “[client %s1]SSL: apr_base64_encode_len error.(%s2)” “[client %s1]SSL: Not enough memory for base64 encoding.(%s2)”
ihs66999	ihs00599	メッセージIDを変更し、メッセージの内容を以下に変更しました。 旧: “SSL: unexpected error (%s1,%s2,%s3)(%s4)” 新: “SSL: Unexpected error.(%s1,%s2,%s3,%s4) [%s5]”

旧メッセージID	新メッセージID	変更内容
ihs81513	変更なし	メッセージの内容を以下に変更しました。 旧: “The "ihdisp" command option was specified incorrectly. Usage: ihdisp [-a]” 新: “The "ihdisp" command option was specified incorrectly. Usage: ihdisp [-n WebServerName[,...]] [-a]”

4.1.2 V7.0以前のInfoProvider Proからの移行

Windows Solaris

V7.0以前のInfoProvider Proからの移行方法について説明します。

なお、Webサーバとしての基本的な動作は移行できますが、InfoProvider Proの機能の一部は、Interstage HTTP Serverでは使用できません。Interstage HTTP Serverにおいて該当する機能を使用せずに運用してください。

Interstage HTTP Serverで使用できない機能には、主な項目として以下のような機能があります。

- ・ コード変換
- ・ メモリキャッシュ
- ・ 拡張CGI
- ・ 特定アプリケーション向けインタフェース
- ・ WebGateway連携
- ・ NETSTAGE高速連携

InfoProvider Proの動作環境を移行する場合、インストール後にInterstage管理コンソールなどの環境設定を行い、InfoProvider Proの環境定義に応じて、Interstage HTTP Serverの環境定義ファイル(httpd.conf)を設定します。以下の対処を行ってください。

1. [InfoProvider Pro環境定義ファイルの対処](#)
2. [セキュリティ管理ファイルの対処](#)
3. [SSL環境の対処](#)
4. [クライアント証明書条件ファイルの対処](#)
5. [ディレクトリサーバ情報管理ファイルの対処](#)



注意

以降に説明されている対処においてInterstage管理コンソールでのWebサーバの環境定義を有効とする場合は、Interstage HTTP Serverの環境定義ファイル(httpd.conf)を編集する必要はありません。

1) InfoProvider Pro環境定義ファイルの対処

InfoProvider Proの動作環境が定義されているInfoProvider Pro環境定義ファイルのそれぞれの定義項目を、Interstage HTTP Serverの環境定義ファイル(httpd.conf)の各ディレクティブに設定します。一部の定義項目については、Interstage管理コンソールで設定することもできます。以下に対応表を示します。

なお、InfoProvider Proの環境定義ファイルおよび定義項目の詳細についてはInterstage Application Server V7.0以前のマニュアルの“Webサーバ運用ガイド(InfoProvider Pro編)”を、Interstage HTTP Serverの環境定義ファイルおよびディレクティブの詳細については“Interstage HTTP Server 運用ガイド”を参照してください。

InfoProvider Proの定義項目	InfoProvider Proの定義名	Interstage HTTP Server		備考
		ディレクティブ名	Interstage管理コンソールの設定可否	
説明	comment	なし	—	各行の先頭にハッシュマーク(#)で指定します。
インストールディレクトリ	installdir	なし	—	設定する必要はありません(動作上問題なし)。
動作ディレクトリ	exedir	ServerRoot	設定不可	Interstage HTTP Serverの動作ディレクトリをServerRootに設定します。
ホスト名	hostname	ServerName	設定可 (注1)	
IPアドレス	IP-address	Listen	設定可 (注1)	複数Webサーバ機能を使用します。複数Webサーバの運用については、“Interstage HTTP Server 運用ガイド”の“運用・保守”―“複数Webサーバの運用”を参照してください。
IPバージョン	Solaris IP-version	Allow Deny Listen NameVirtualHost <VirtualHost>	設定可 (注) IPアドレスの指定に、IPv6アドレスを設定します。	Allow、Deny、Listen、NameVirtualHost、<VirtualHost>の各ディレクティブでIPアドレスの指定に、IPv6アドレスを設定します。
ポート番号	port	Listen	設定可 (注1)	
公開する最上位のディレクトリ名	acstop	DocumentRoot	設定可 (注1)	
ホームページ	topHTML	DirectoryIndex	設定不可	代表するHTMLファイル文書と同一です。
代表するHTMLファイル文書	typical-HTML	DirectoryIndex	設定不可	
データタイプとファイルの拡張子の関連を指定するファイル名	content-type	TypesConfig	設定不可	デフォルトのデータタイプ(MIMEタイプ)は、“text/plain”に変更となります。 “Interstage HTTP Server 運用ガイド”の“環境定義ファイル”―“デフォルトMIMEタイプの設定”を参照して設定してください。
ステータスコードファイル名	statusfile	ErrorDocument	設定不可	エラードキュメントは先頭にスラッシュ(/)を付加し、ServerRootからの相対パスで指定します。
仮想ディレクトリ	link	Alias	設定不可	
DNSの逆引き動作	dns-resolve	HostnameLookups	設定不可	CGIに対する要求を行った場合だけDNSの逆引きを行うという設定はできません。
InfoProvider Proから起動したアプリケーションに渡すコード系	input-code	なし	—	コード変換はできません。

InfoProvider Proの定義項目	InfoProvider Proの定義名	Interstage HTTP Server		備考
		ディレクティブ名	Interstage管理コンソールの設定可否	
Webブラウザへ送信するコード系	output-code	なし	—	コード変換はできません。
サーバ内にあるHTML文書のコード系	server-file-code	なし	—	コード変換はできません。
HTTPプロトコルバージョン	http-ver	なし	—	HTTP/1.1です。
同時アクセス最大数	concurrency	Windows ThreadsPerChild	設定可 (注1)	上限値はThreadLimitディレクティブの設定値です。
		Solaris MaxClients	設定可 (注1)	上限値はServerLimitディレクティブの設定値です。
最大プーリング数	pool-max	Windows なし	設定不可	Windows(R)では、最大プーリング数は設定できません。
		Solaris ListenBacklog	設定不可	
最大接続時間	connection-limit	KeepAliveTime out	設定可 (注1)	
ブラウザ送受信タイムアウト時間	browser-timeout	Timeout	設定可 (注1)	
受け入れ可能リクエストサイズ	acceptance-request-size	LimitRequestBody	設定可 (注1)	
キャッシュヘッダ制御	cache-control	Header	設定不可	HTTPヘッダの値を設定します。 “Pragma”ヘッダを“no-cache”として返却する場合、以下のよう指定します。 Header set Pragma no-cache
キャッシュファイル数	cache-file-no	なし	—	メモリキャッシュはできません。
キャッシュファイルサイズ	cache-file-size	なし	—	メモリキャッシュはできません。
アクセスログ採取方法	acslogまたは logautoまたは acslog-rotation	CustomLog	設定可 (注1)	指定方法が異なります。 “Interstage HTTP Server 運用ガイド”の“環境定義ファイル” — “アクセスログの設定”を参照して設定してください。
全アクセスのログ出力	acslog-all	なし	—	中断系のログは採取できません。
アクセスログの形式	logformat	CustomLogおよびLogFormat	設定不可	指定方法が異なります。 拡張モードを指定する場合は、CustomLogのニックネームに“combined”を指定してください。
処理時間の出力	acslog-proctime	CustomLogおよびLogFormat	設定不可	指定方法が異なります。 処理時間を出力する場合は、LogFormatに“%S”を指定して、CustomLogに該当するニックネームを指定してください。

InfoProvider Proの定義項目	InfoProvider Proの定義名	Interstage HTTP Server		備考
		ディレクティブ名	Interstage管理コンソールの設定可否	
Cookie情報の出力	acslog-cookie	CustomLogおよびLogFormat	設定不可	指定方法が異なります。 “Interstage HTTP Server 運用ガイド”の“環境定義ファイル”－“クッキーログの設定”を参照して設定してください。
エラーログのファイル名	errlog	ErrorLog	設定可 (注1)	指定方法が異なります。 “Interstage HTTP Server 運用ガイド”の“環境定義ファイル”－“エラーログの設定”を参照して設定してください。
エラーログの形式	errlog-format	なし	—	エラーログの形式は設定できません。
ログ出力異常時の動作	log-erraction	なし	—	ログ出力異常時は継続運用されます。
受信データ改ざん通知ログ出力しきい値	falsification-threshold	なし	—	改ざん発生時はログが出力されません。
改ざん検知によるInfoProvider Pro停止	falsification-term	なし	—	改ざん検知による停止は行いません。
受信可能最大サイズ制限オーバー発生回数ログ出力しきい値	reqsize-threshold	なし	—	受信可能最大サイズの制限オーバー発生時には毎回ログに出力されます。
受信可能最大サイズ制限オーバー発生回数によるInfoProvider Pro停止	reqsize-term	なし	—	受信可能最大サイズの制限オーバー発生による停止は行われません。
Servletサービスのアクセスログ出力形式	acslog-jsformat	なし	—	該当する機能はありません。常にURI部分がURLで要求してきた文字列(common)となります。
異常終了時サービス同期停止	Windows sync	なし	—	異常終了時、サービスは停止されません。
CGIアプリケーションの格納場所と識別名	cgi-path-idnt	ScriptAlias (注2)	設定不可	指定する順番が異なるので、注意してください。
アプリケーション送受信タイムアウト時間	timeout	Timeout	設定可 (注1)	ブラウザ送受信タイムアウト時間と同じ時間になります。
タイムアウトしたCGIアプリケーションへの対応	timeout-kill	なし	—	該当するCGIアプリケーションは強制終了されます。
Location返却フラグ	location-redirect	なし	—	Locationヘッダがそのまま返却されます。
拡張CGI定義	gai-gw	なし	—	拡張CGIは使用できません。
セッション管理型拡張CGIのログアウト監視時間間隔	alarmtime	なし	—	拡張CGIは使用できません。
フィルタアプリケーションのパス	filter-file	LoadModule	設定不可	(注3)

InfoProvider Proの定義項目	InfoProvider Proの定義名	Interstage HTTP Server		備考
		ディレクトティブ名	Interstage管理コンソールの設定可否	
InfoProvider Proの実行時のユーザ名	Solaris uid	Solaris User	—	
InfoProvider Proの実行時のグループ名	Solaris gid	Solaris Group	—	
セキュリティ管理ファイル名	authfile	なし	—	セキュリティ管理ファイルを別途作成する必要はありません。 セキュリティ管理ファイルの対処 を参照してください。
オンライン照合機能ログ	dirservlog	なし	—	オンライン照合機能の動作ログはアクセスログに出力されます。
SSL環境定義ファイル名	sslfile	なし	—	SSL環境定義ファイルを別途作成する必要はありません。 SSL環境の対処 を参照してください。
使用SSLライブラリ	ssllib	なし	—	SSLライブラリはSMEE3です。

注1)

Interstage管理コンソールの操作画面を以下に示します。Interstage管理コンソールの起動については“運用ガイド(基本編)”を、Interstage管理コンソールの定義詳細についてはInterstage管理コンソールのヘルプを参照してください。

InfoProvider Proの定義項目	Interstage HTTP ServerのInterstage管理コンソール
ホスト名	[システム] > [サービス] > [Webサーバ] > [Webサーバ名] > [環境設定]タブ > [詳細設定[表示]] > [詳細設定]-[サーバのホスト名またはIPアドレス]
IPアドレス	[システム] > [サービス] > [Webサーバ] > [新規作成]タブ > [Webサーバ: 新規作成]
ポート番号	[システム] > [サービス] > [Webサーバ] > [Webサーバ名] > [環境設定]タブ > [簡易設定]-[ポート番号]
公開する最上位のディレクトリ名	[システム] > [サービス] > [Webサーバ] > [Webサーバ名] > [環境設定]タブ > [簡易設定]-[クライアントに公開するサーバ資源の最上位ディレクトリ]
同時アクセス最大数	[システム] > [サービス] > [Webサーバ] > [Webサーバ名] > [環境設定]タブ > [詳細設定[表示]] > [詳細設定]-[クライアントの同時接続数]
最大接続時間	[システム] > [サービス] > [Webサーバ] > [Webサーバ名] > [環境設定]タブ > [詳細設定[表示]] > [詳細設定]-[Keep-Alive接続において、次のリクエストまでのタイムアウト時間]
ブラウザ送受信タイムアウト時間	[システム] > [サービス] > [Webサーバ] > [Webサーバ名] > [環境設定]タブ > [詳細設定[表示]] > [詳細設定]-[タイムアウト]
アプリケーション送受信タイムアウト時間	
受け入れ可能リクエストサイズ	[システム] > [サービス] > [Webサーバ] > [Webサーバ名] > [環境設定]タブ > [詳細設定[表示]] > [詳細設定]-[リクエストメッセージ本体の最大サイズ制限]
アクセスログ採取方法	[システム] > [サービス] > [Webサーバ] > [Webサーバ名] > [環境設定]タブ > [詳細設定[表示]] > [アクセスログ]
エラーログのファイル名	[システム] > [サービス] > [Webサーバ] > [Webサーバ名] > [環境設定]タブ > [詳細設定[表示]] > [エラーログ]

注2)

CGIプログラム実行を許可する場合は、環境定義ファイル(httpd.conf)において、コメント行として定義されている以下のディレクティブのハッシュマーク(#)を取り除いてください。なお、CGIプログラム実行の許可設定の詳細については、“Interstage HTTP Server 運用ガイド”の“環境設定”―“環境定義ファイル”―“CGIプログラム実行の許可”を参照してください。

Windows

```
【修正前】
#LoadModule cgi_module "C:/Interstage/F3FMihs/modules/mod_cgi.so"
【修正後】
LoadModule cgi_module "C:/Interstage/F3FMihs/modules/mod_cgi.so"
```

Solaris

Linux

```
【修正前】
#LoadModule cgi_module "/opt/FJSVihs/modules/mod_cgi.so"
#ScriptAlias /cgi-bin/ "/opt/FJSVihs/servers/FJapache/cgi-bin/"
【修正後】
LoadModule cgi_module "/opt/FJSVihs/modules/mod_cgi.so"
ScriptAlias /cgi-bin/ "/opt/FJSVihs/servers/FJapache/cgi-bin/"
```

注3)

InfoProvider Pro向けのフィルタアプリケーションは、Interstage HTTP Serverでは使用できません。
Interstage HTTP Server用に同等機能のモジュールが提供されている場合があります。
LoadModuleには、Interstage HTTP Server用のモジュールを以下のように指定してください。

Windows

```
LoadModule example_module "C:/Interstage/F3FMihs/modules/mod_example.so"
```

Solaris

```
LoadModule example_module "/opt/FJSVihs/modules/mod_example.so"
```

2) セキュリティ管理ファイルの対処

InfoProvider Proの認証とアクセス制御が定義されているセキュリティ管理ファイルのそれぞれの定義項目を、Interstage HTTP Serverの環境定義ファイル(httpd.conf)の各ディレクティブに設定します。以下に対応表を示します。Interstage管理コンソールで設定することはできません。

なお、InfoProvider Proのセキュリティ管理ファイルおよび定義項目の詳細についてはInterstage Application Server V7.0以前のマニュアルの“セキュリティシステム運用ガイド”の“InfoProvider Proの認証とアクセス制御の設定”を、Interstage HTTP Serverの環境定義ファイルおよびディレクティブの詳細については“Interstage HTTP Server 運用ガイド”を参照してください。

InfoProvider Proの定義項目	InfoProvider Proの定義名	Interstage HTTP Serverのディレクティブ名	備考
パスワード管理ファイル名	PasswdFile	AuthUserFile	AuthUserFileにhtpasswdコマンドを使用して作成したパスワードファイルを設定します。
グループ管理ファイル名	GroupFile	AuthGroupFile	
IPグループ管理ファイル名	IPGroupFile	なし (Allow)	Allowにアクセスを許可するホストを設定します。“Interstage HTTP Server運用ガイド”の“環境定義ファイル”―“IPアクセスコントロールの設定”を参照して設定してください。

InfoProvider Proの定義項目		InfoProvider Proの定義名	Interstage HTTP Serverのディレクティブ名	備考
プロテクトパス	URL	—	<Location>	URLを<Location>に指定し、<Location>内にプロテクトパス設定部の情報を記述します。
	ユーザ管理ファイル名	UserFile	なし	ユーザ管理ファイルは使用できません。
	アクセスを許可するユーザ名	User	Require	ユーザを指定する場合は、以下のよう に設定します。 Require user ユーザ名 グループを指定する場合は、以下のよう に設定します。 Require group グループ名
	IPアドレス/ホスト名	IPAddress	Allow	Allowにアクセスを許可するホストを設定 します。“Interstage HTTP Server 運用ガイ ド”の“環境定義ファイル”—“IPアクセ スコントロールの設定”を参照して設定し てください。
	認証名	AuthName	AuthName	
	認証方法	AuthType	AuthType	
	クライアント証明書 条件ファイル名	ClientCert	なし	クライアント証明書条件ファイルを別途作 成する必要はありません。 クライアント証 明書条件ファイルの対処 を参照してくだ さい。
	ディレクトリサーバ のベースDN	SearchBase	AuthLDAPbasedn	
ディレクトリサーバ情報管理ファイル 名		DirServFile	なし	ディレクトリサーバ情報管理ファイル名を 別途作成する必要はありません。 ディレ クトリサーバ情報管理ファイルの対処 を 参照してください。

3) SSL環境の対処

InfoProvider Proにおいて構築したSSL環境は、Interstage HTTP ServerのSSL環境として移行することはできません。SSL環境を再構築する必要があります。SSLの環境設定については“セキュリティシステム運用ガイド”の“Interstage証明書環境の構築と利用”または“Interstage HTTP ServerでSSLを利用する方法”を参照してください。

4) クライアント証明書条件ファイルの対処

SSL環境を再構築したあと、InfoProvider Proのクライアント証明書条件ファイルのそれぞれの定義項目を、Interstage HTTP Serverの環境定義ファイル(httpd.conf)のディレクティブに設定します。対応表を以下に示します。Interstage管理コンソールでは、設定できません。

InfoProvider Proのクライアント証明書条件ファイルおよび各定義項目の詳細についてはInterstage Application Server V7.0以前のマニュアルの“セキュリティシステム運用ガイド”の“InfoProvider Proの認証とアクセス制御の設定”を、Interstage HTTP Serverの環境定義ファイルおよびディレクティブの詳細については“Interstage HTTP Server運用ガイド”を参照してください。

なお、Interstage HTTP Serverの証明書の認証条件によるクライアント認証の詳細については、“セキュリティシステム運用ガイド”の“クライアント認証”—“クライアント認証(証明書の条件設定による認証)”を参照してください。

InfoProvider Proの定義項目		InfoProvider Proの定義内容	Interstage HTTP Serverのディレクティブ定義内容(注1)
条件項目	氏名	n = 氏名	RewriteCond %{ENV:SSL_CLIENT_CN} 氏名
	組織名	o = 組織名	RewriteCond %{ENV:SSL_CLIENT_O} 組織名
	組織単位名	u = 組織単位名	RewriteCond %{ENV:SSL_CLIENT_OU} 組織単位名
	肩書	t = 肩書	RewriteCond %{ENV:SSL_CLIENT_T} 肩書
	国名	c = 国名	RewriteCond %{ENV:SSL_CLIENT_C} 国名
	電話番号	te = 電話番号	RewriteCond %{ENV:SSL_CLIENT_PHONE} 電話番号
	メールアドレス	m = メールアドレス	RewriteCond %{ENV:SSL_CLIENT_EMAIL} メールアドレス
	州 都道府県	なし	RewriteCond %{ENV:SSL_CLIENT_ST} 州 都道府県
	市町村名	なし	RewriteCond %{ENV:SSL_CLIENT_L} 市町村名
クライアント認証識別子		ID = 識別子	RewriteRule .* - [E=REMOTE_USER:識別子,E=GAI_REMOTE_USER:識別子] (注2)

注1) RewriteCondディレクティブ(複数指定可能)のあとに、RewriteRuleディレクティブ(Lフラグ付)を指定する必要があります。また、最後のRewriteRuleディレクティブ(Lフラグ付)のあとに、RewriteRuleディレクティブ(Fフラグ付)を指定する必要があります。

注2) RewriteRuleディレクティブは、以下のようにLフラグとEフラグを同時に指定することもできます。

```
RewriteRule .* - [L, E=REMOTE_USER:識別子,E=GAI_REMOTE_USER:識別子]
```

InfoProvider Proのクライアント証明書条件ファイルの定義例に対応するInterstage HTTP Serverの環境定義ファイル(httpd.conf)の定義例を以下に示します。

InfoProvider Proのクライアント証明書条件ファイルの定義例

```
{
ID = FJ-D1
o = "Fujitsu Ltd"
u = development1*
t =
c = jp
}
{
ID = FJ-D2
o = "Fujitsu Ltd"
t =
c = jp
m = *dev2.fujitsu.co.jp
}
```

Interstage HTTP Serverの環境定義ファイル(httpd.conf)の定義例

```
RewriteEngine On

RewriteCond %{ENV:SSL_CLIENT_O} "^Fujitsu Ltd$"
RewriteCond %{ENV:SSL_CLIENT_OU} "^development1.*"
RewriteCond %{ENV:SSL_CLIENT_C} ^JP$
RewriteRule .* - [L, E=REMOTE_USER:FJ-D1, E=GAI_REMOTE_USER:FJ-D1]

RewriteCond %{ENV:SSL_CLIENT_O} "^Fujitsu Ltd$"
RewriteCond %{ENV:SSL_CLIENT_C} ^JP$
RewriteCond %{ENV:SSL_CLIENT_EMAIL} .*dev2.fujitsu.co.jp$
```

```
RewriteRule .* - [L,E=REMOTE_USER:FJ-D2,E=GAI_REMOTE_USER:FJ-D2]
```

```
RewriteRule .* - [F]
```

5) ディレクトリサーバ情報管理ファイルの対処

InfoProvider Proがディレクトリサーバと通信するための情報が定義されているディレクトリサーバ情報管理ファイルのそれぞれの定義項目を、Interstage HTTP Serverの環境定義ファイル(httpd.conf)の各ディレクティブに設定します。以下に対応表を示します。Interstage 管理コンソールで設定することはできません。

ただし、Interstage HTTP Serverとディレクトリサーバ間でSSLを使用したセキュアな通信を行う場合は、SSLの環境設定を再構築する必要があります。

InfoProvider Proのディレクトリサーバ情報管理ファイルおよび定義項目の詳細についてはInterstage Application Server V7.0以前のマニュアルの“セキュリティシステム運用ガイド”の“InfoProvider Proの認証とアクセス制御の設定”を、Interstage HTTP Serverの環境定義ファイルおよびディレクティブの詳細については“Interstage HTTP Server運用ガイド”を、Interstage HTTP ServerのSSLの環境設定については“セキュリティシステム運用ガイド”の“Interstage HTTP Serverの認証とアクセス制御の設定”－“オンライン照合”を参照してください。

InfoProvider Proの 定義項目	InfoProvider Proの定義名	Interstage HTTP Serverの ディレクティブ名	備考
ホスト名	Hostname	AuthLDAPHost	
SSL	SSL	AuthLDAPSecure	
ポート番号	Port	AuthLDAPPort	
ベースDN	BaseDN	AuthLDAPbasedn	
BindDN名	BindDN	AuthLDAPBindDN	
BindPassword	BindPassword	AuthLDAPBindPassword	
匿名の許可	AnonymousPermission	なし	匿名のアクセスを許可することはできません。

4.2 Servletサービス(Tomcat 5.5ベースのサーブレット実行環境)への移行

Interstage Application Server/Interstage Web Server V9.0以降、Tomcat 5.5ベースのサーブレット実行環境(以降、Tomcat 5.5ベースのServletサービス)を提供しています。

4.2.1 Tomcat3.1ベースのServletサービスからの移行について

「Tomcat3.1ベースのServletサービス」と「Tomcat5.5ベースのServletサービス」(V9以降のIJSERVER)では運用方法が異なります。

「Tomcat5.5ベースのServletサービス」の操作は、Interstage管理コンソールまたはisj2eeadminコマンドを使用して行います。Interstage 管理コンソールについては“運用ガイド(基本編)”を、isj2eeadminコマンドについては“リファレンスマニュアル(コマンド編)”を参照してください。

【配備方法】

Interstage管理コンソールでIJSERVERワークユニットを作成した後、Webアプリケーションを配備しなおします。

【環境定義ファイル】

Tomcat3.1ベースのServletサービス固有の環境ファイルの移行については、“付録A V5.1以前のServletサービス環境定義の移行”を参照してください。

その他の環境定義ファイルについては以下を参照してください。

Tomcat3.1ベースの Servletサービス	Tomcat5.5ベースのServletサービス(V9以降のIJSERVER)
名前変換定義ファイル (注1)	・ [ワークユニット] > “ワークユニット名” > “モジュール名” > [名前変換]タブ

Tomcat3.1ベースのServletサービス	Tomcat5.5ベースのServletサービス(V9以降のIIServer)
配備時の名前変換定義 (interstage.xml)の <app-name>タグ	「Tomcat3.1ベースのServletサービス」では指定しても意味がありませんでしたが、「Tomcat5.5ベースのServletサービス」(V9以降のIIServer)では名前変換定義情報の設定対象であるWebアプリケーションの名前を正しく指定してください。app-nameタグを指定しない場合や、タグに間違ったWebアプリケーション名を指定すると名前変換は動作しません。 app-nameタグに指定するWebアプリケーション名は以下のようになしてください。 - 以下の文字以外の文字が使用されていないこと 英数字、'+'、'-'、'!'、'_','\$','/' '!'を使用する場合は、他の文字と一緒に使用されていること

注1)

以下のファイルです。

Windows

C:\¥INTERSTAGE¥j2ee¥etc¥FJWebebeProperties.xml

Solaris

Linux

/opt/FJSVj2ee/etc/FJWebebeProperties.xml

4.2.2 Servletサービス移行時の注意

Tomcat3.1ベースのServletサービス、またはTomcat4.1ベースのServletサービスから、Tomcat5.5ベースのServletサービス(V9のIIServer)へ移行する場合の注意事項について説明します。

特に表中記載のないものについては、設定や動作上の差分はありません。

- Webアプリケーション環境定義ファイル(deployment descriptor)の記述形式
- サーブレット・マッピング
- JSPのimport
- 配備時のWebアプリケーション名
- HTTPトネリングの移行
- マルチコンテナ機能の移行
- JSPのコンパイル
- Systemwalker Service Quality Coordinatorと連携したトランザクション内訳分析の対象
- Interstage管理コンソールのWebアプリケーションのモニタの対象
- JSPのリロード
- セッション管理用CookieにSecure属性を常に付加する
- リクエストのエンコーディング
- HTTP TRACEメソッド
- 接続先コネクタの制限
- 同時処理数
- JSP Document(XMLベースのJSP)の属性の順番
- リクエストのURIに¥を含む場合

■Webアプリケーション環境定義ファイル(deployment descriptor)の記述形式

Tomcat5.5ベースのServletサービスでは、Webアプリケーション環境定義ファイル(deployment descriptor)のタグは以下の方法でチェックします。

- Servlet2.2/Servlet2.3のdeployment descriptor
DTD(文書型定義)ベース
- Servlet2.4のdeployment descriptor
XMLスキーマベース
- **Tomcat3.1ベースのServletサービスから移行時の注意**
Tomcat3.1ベースのServletサービスでは独自の方式でチェックを行っていたため、これまでチェックされていなかった内容がチェックされる場合があります。
- **Tomcat4.1ベースのServletサービスから移行時の注意**
deployment descriptorをServlet2.4のものに修正した場合、チェック方法(チェック対象)が異なるため、これまで配備時にはチェックされていなかった内容がチェックされる場合があります。

タグの指定方法が誤っている場合には、Interstage管理コンソールで配備を行った際に、画面上のメッセージ領域にエラーメッセージを出力して配備を中断します。

また、IJServerの起動時にタグの指定方法の誤りが検出された場合には、IJServerのコンテナログに以下のエラーメッセージが出力されますので、“J2EEユーザーズガイド”の“Webアプリケーション環境定義ファイル(deployment descriptor)”を参照して修正してください。たとえば、servlet-mappingタグをservletタグよりも前に定義した場合にエラーとなります。

[ERROR] Digester - -Parse Error at line [行数] column [列数]: [原因]

Webアプリケーション環境定義ファイル(deployment descriptor)修正後、IDE(統合開発環境ツール)等に付属しているdeployment descriptorの検証機能を使用して、修正内容の妥当性を検証することを推奨します。

配備済みのアプリケーションのdeployment descriptorの記述形式を、Servlet2.2または2.3からServlet2.4の記述形式に変更する場合は、アプリケーションを配備しなおしてください。

■サーブレット・マッピング

- **Tomcat3.1ベースのServletサービスから移行時の注意**
デフォルトでは、サーブレット・マッピングの定義を行わない、以下のURLの形式でのサーブレットの呼び出しはできません。

http://サーバホスト名:ポート番号/Webアプリケーション名/servlet/サーブレット名

IJServerワークユニットの環境設定で、サーブレット・マッピングの定義を行わずに、上記URLの形式で呼び出すことができる設定に切り替えることができますが、以下の理由から推奨しません。

- クラス名が分かればすべてのサーブレットが動作できてしまう(セキュリティ)。
- Filterアプリケーションの呼び出しがサポートされない。
- Servletの仕様外の機能である(移植性)。

- **Tomcat4.1ベースのServletサービスから移行時の注意**
特にありません。

「V6.0以降のServletサービス」では、Webアプリケーション環境定義ファイル(deployment descriptor)のサーブレット・マッピングで定義されたものしか動作しません。

「V5.1以前のServletサービス」で、サーブレット・マッピングの定義を行わず、以下のURLの形式でサーブレットの呼び出しを行っていた場合は、Webアプリケーション環境定義ファイルにサーブレット・マッピングの定義を行う必要があります。

■JSPのimport

Tomcat5.5ベースのServletサービスでは、JSP1.2/2.0の仕様の以下のJavaパッケージのクラス以外は、importなしにクラス名のみで直接スクリプトレット使用はできません。

javax.servlet

- javax.servlet

- javax.servlet.http
- javax.servlet.jsp
- **Tomcat3.1ベースのServletサービスから移行時の注意**
仕様外のクラスを使用している場合は、下記の対処を行ってください。

- **Tomcat4.1ベースのServletサービスから移行時の注意**
JavaVMオプションの以下のプロパティは使用できません。

-Dcom.fujitsu.interstage.j2ee.ijserver.jspImplicitImport=true(英大文字・英小文字の区別なし)

これにより下記のクラスも使用できなくなります。

- java.io.*
- java.util.*

仕様外のクラスを使用している場合は、下記の対処を行ってください。

Tomcat5.5ベースのServletサービスで仕様外のクラスを使用する場合は、以下のいずれかの対処を行ってください。

- **JSPのPageディレクティブで正しくimport宣言を行う。**
- **Servlet2.4(JSP2.0)の機能を使用する。**
 1. deployment descriptor(web.xml)を、Servlet2.4用に修正します。(Servlet2.3以前のdeployment descriptorの場合)
詳細は“J2EEユーザズガイド”の“Webアプリケーション環境定義ファイル(deployment descriptor)”を参照してください。
 2. JSPの必要なimport文を含んだファイルを用意します。

【例】header.jspf

```
<%@ page import="java.io.*, java.util.*" %>
```

3. deployment descriptor(web.xml)に<jsp-config>タグを追加し、対象とするJSPに対して、上記header.jspfを設定します。
この例では、header.jspfファイルをアプリケーションのWEB-INF/jspfディレクトリ配下に格納し、すべてのJSPに対して設定しています。

【例】deployment descriptor(web.xml)

```
<?xml version="1.0" encoding="UTF-8" ?>
<web-app xmlns="http://java.sun.com/xml/ns/j2ee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
    http://java.sun.com/xml/ns/j2ee/web-app_2_4.xsd"
  version="2.4">
  . . .
  <jsp-config>
    <jsp-property-group>
      <url-pattern>*.jsp</url-pattern>
      <include-prelude>/WEB-INF/jspf/header.jspf</include-prelude>
    </jsp-property-group>
  </jsp-config>
  . . .
</web-app>
```

■ 配備時のWebアプリケーション名

- Tomcat3.1ベースのServletサービスから移行時の注意

Webアプリケーションの配備を行う場合、Webアプリケーション名の省略値には以下の値が使用されます。

- ー EARファイルの場合

- application.xmlのcontext-rootに値が設定されている場合には、その設定されている名前が使用されます。
- application.xmlのcontext-rootに値が設定されていない場合には、WARファイル名から拡張子を除いた名前が使用されます。

- ー WARファイルの場合

WARファイル名から拡張子を除いた名前が使用されます。

- Tomcat4.1ベースのServletサービスから移行時の注意

特にありません。

■ HTTPトンネリングの移行

- Tomcat4.1ベースのServletサービスから移行時/Tomcat3.1ベースのServletサービスから移行時の共通の注意

CORBA_ORB_init()関数でのHTTPトンネリング機能の設定は有効となりません。

HTTPトンネリング機能を使用する場合は、IJSerのJavaVMオプション、またはFJjndi.propertiesファイルで、環境プロパティとして「HTTPGW」を設定してください。

環境プロパティ「HTTPGW」の値と設定内容については、“J2EEユーザーズガイド”の“JNDIサービスプロバイダの環境設定”の“J2EEアプリケーションクライアント”を参照してください。

注) アプリケーションでのJNDI機能の使用有無にかかわらず設定してください。

なお、CORBA_ORB_init()関数で設定していた場合でも、アプリケーションの修正は必要ありません。

- Tomcat3.1ベースのServletサービスから移行時の注意

HTTPトンネリング機能は、Webアプリケーションのみ運用のタイプのIJSerで使用可能です。以下のタイプのIJSerではHTTPトンネリング機能は使用できません。

- ー WebアプリケーションとEJBアプリケーションを同一JavaVMで運用

- ー WebアプリケーションとEJBアプリケーションを別JavaVMで運用

- Tomcat4.1ベースのServletサービスから移行時の注意

特にありません。

■ マルチコンテナ機能の移行

- Tomcat3.1ベースのServletサービスから移行時の注意

マルチコンテナ機能を使用していた場合には、WebアプリケーションをIJSerへ配備し直す必要があります。“付録A V5.1以前のServletサービス環境定義の移行”の“サーブレット・ゲートウェイ環境の移行”を参照してください。

- Tomcat4.1ベースのServletサービスから移行時の注意

特にありません。

■ JSPのコンパイル

Java言語仕様に準拠していないコードや推奨されないコードを含むJSPは、以前のバージョンからの移行により、コンパイルに失敗したり、実行結果が変わったりすることがあります。

移行時にはJSPを見直し、Java言語仕様に従っているか確認してください。



例

JSP内のJavaコードで、ローカル変数名に“org”を使用している。

⇒ユニークなパッケージ名の先頭に使用される文字列(“org”や“com”など)を、ローカル変数名に使用すべきではありません。

■Systemwalker Service Quality Coordinatorと連携したトランザクション内訳分析の対象

- Tomcat3.1ベースのServletサービスから移行時の注意
未サポートです。
- Tomcat4.1ベースのServletサービスから移行時の注意
[ワークユニット] > “ワークユニット名” > [環境設定]タブ > [Servletコンテナ設定]で、「マッピングがなくてもサーブレットが動作する」が設定されている場合、以下の形式で呼び出したサーブレットの情報は、すべてまとめて「invoker」として出力されていましたが、この情報は出力されなくなります。
`http://servername[:port]/Webアプリケーション名/servlet/サーブレットクラス名`

■Interstage管理コンソールのWebアプリケーションのモニタの対象

- Tomcat3.1ベースのServletサービスから移行時の注意
未サポートです。
- Tomcat4.1ベースのServletサービスから移行時の注意
[ワークユニット] > “ワークユニット名” > [環境設定]タブ > [Servletコンテナ設定]で、「マッピングがなくてもサーブレットが動作する」が設定されている場合、以下の形式で呼び出したサーブレットの情報は、すべてまとめて「invoker」として出力されていましたが、この情報は出力されなくなります。
`http://servername[:port]/Webアプリケーション名/servlet/サーブレットクラス名`

■JSPのリロード

クラスのオートリロード機能とは別に設定可能となりました。
[ワークユニット] > [新規作成]タブ > [詳細設定] > [Servletコンテナ設定] > [JSPのリロード]で設定可能です。

■セッション管理用CookieにSecure属性を常に付加する

Tomcat5.5ベースのServletサービスでは、Interstage管理コンソールより以下で設定を行ってください。

[システム] > [ワークユニット] > “ワークユニット名” > “Webアプリケーション名” > [環境設定]タブ > [コンテキスト設定] > [セッション] > クッキーにSecure属性を常に付加する

既存通りJavaVMオプションとして定義することも可能(※)ですが、Interstage管理コンソールからの設定を推奨します。

※) いずれかで設定されていれば有効となります。

■リクエストのエンコーディング

Servletサービスのバージョンによって、リクエストを処理するエンコーディングにデフォルトでは以下の動作差分があります。

対象	Tomcat4.1ベースのServletサービス	Tomcat5.5ベースのServletサービス
リクエストURI	UTF-8 (注1)	ISO-8859-1(設定可能)
GETメソッド時のリクエストパラメータ(クエリストリング)	リクエストボディを処理するエンコーディング (注2)	以下から選択可能。 ・リクエストURIを処理するエンコーディング ・リクエストボディを処理するエンコーディング (注2)
POSTメソッド時のリクエストパラメータ(リクエストボディ)	リクエストボディを処理するエンコーディング (注2)	リクエストボディを処理するエンコーディング (注2)
リクエストボディを処理するReader ServletRequest#getReader()	リクエストボディを処理するエンコーディング (注2)	リクエストボディを処理するエンコーディング (注2)

注1) PG48587の修正を適用した場合は以下の順で処理されます。

1. UTF-8

2. OSのデフォルトエンコーディング

3. 自動判定 (Java言語の自動判定機能(JISAutoDetect)を使用)

注2)「リクエストボディを処理するエンコーディング」は以下が該当します。

- ・ リクエストのContent-Typeヘッダで示されるcharset
- ・ WebアプリケーションでリクエストにsetCharacterEncodingメソッドで設定したエンコーディング
- ・ Webアプリケーション環境設定で「エンコーディング」に指定したエンコーディング

Tomcat5.5ベースのServletサービスでTomcat4.1ベースのServletサービスと同じ動作にする場合は以下の設定を行ってください。

- ・ [ワークユニット] > “ワークユニット名” > [Servletコンテナ設定] > [リクエストURIのエンコーディング]に“UTF-8”
- ・ [ワークユニット] > “ワークユニット名” > [Servletコンテナ設定] > [リクエストボディ処理のエンコーディングをクエリパラメタに使用する]に“する”

また、V8(Tomcat4.1ベースのServletサービス)とV9(Tomcat5.5ベースのServletサービス)では、Webアプリケーション環境設定で「エンコーディング」を指定した場合の動作に以下の改善が行われています。

これにより、リクエストのContent-Typeヘッダに正しくエンコーディング「charset=」が指定されている場合は、その値が有効となります。

(1)Webアプリケーション環境設定「エンコーディング」	(2)リクエストのContent-Typeのcharset	(3)アプリケーションによるリクエストへのsetCharacterEncoding	Tomcat4.1ベースのServletサービス	Tomcat5.5ベースのServletサービス
指定あり	あり	あり	(3)が有効	(3)が有効
		なし	(1)が有効	(2)が有効
	なし	あり	(3)が有効	(3)が有効
		なし	(1)が有効	(1)が有効
指定なし	あり	あり	(3)が有効	(3)が有効
		なし	(2)が有効	(2)が有効
	なし	あり	(3)が有効	(3)が有効
		なし	デフォルト (ISO-8859-1)	デフォルト (ISO-8859-1)

■HTTP TRACEメソッド

Tomcat5.5ベースのServletサービスでは、HTTP TRACEメソッドはセキュリティ上の理由から使用できません。

なお、HTTP TRACEメソッドは、Webサイトの利用者が通常使用する機能ではありません。

■接続先コネクタの制限

Tomcat4.1ベースのServletサービスで、Webサーバコネクタの証明書のシリアル番号と発行者識別DNを指定して接続先コネクタの制限を行っていた場合、Tomcat5.5ベースのServletサービスでは、Interstage管理コンソールの以下の設定で、WebサーバコネクタとServletコンテナ間の通信で使用するSSL定義を適切に設定して、接続先コネクタの制限を行ってください。

- ・ [システム] > [セキュリティ] > [SSL]の新規作成、または、参照・変更の詳細設定項目である「[認証局証明書のニックネーム]の選択」

■同時処理数

Tomcat5.5ベースのServletサービスでは、同時処理数に指定した値のうちの1つは、クライアントシステムからのリクエスト受け付け用として、Servletコンテナの内部処理で使用されます。そのため、指定できる最小値は2になります。

■JSP Document(XMLベースのJSP)の属性の順番

JSP Documents(XML形式のドキュメント)の場合、XMLでは属性の順番が保証されないため、依存するアプリは移行により動作しない場合があります。

問題が発生する場合は、依存しないよう修正してください。



例

修正前:

```
<vs:loop begin="%= last - s %" end="%= last++ %" />
```

修正後:

```
<vs:loop begin="%= last - s %" end="%= last %" /><jsp:scriptlet>last++;</jsp:scriptlet>
```

■リクエストのURIに¥を含む場合

Tomcat5.5ベースのServletサービスでは、リクエストのURIに¥(エンコードされている場合も該当)を含む場合、Tomcat4.1ベースのServletサービス(PG56136の修正未適用)では呼び出し可能であったコンテンツが呼び出しできなくなることがあります。

4.2.3 アプリケーションの非互換一覧

Tomcat3.1ベースのServletサービス、Tomcat4.1ベースのServletサービスと、Tomcat5.5ベースのServletサービス(V9のJServer)のアプリケーション動作の非互換項目を以下に示します。該当する機能をWebアプリケーションで使用している場合には、Webアプリケーションを修正してください。

Servlet/JSP仕様の差分については、Java(TM) Servlet SpecificationおよびJavaServer Pages(TM) Specificationをあわせて参照してください。

またシステム移行の際には、運用前に業務の観点で十分な動作検証を実施してください。

No.	機能	Tomcat3.1ベースのServletサービス	Tomcat4.1ベースのServletサービス	Tomcat5.5ベースのServletサービス
1	カスタムタグのTag#release()メソッド	カスタムタグの開発で実装する javax.servlet.jsp.tagext.Tagインタフェースのpublic void release()メソッドは、カスタムタグを使用したページの呼び出しが終わるごとに呼び出されます。	カスタムタグの開発で実装する javax.servlet.jsp.tagext.Tagインタフェースのpublic void release()メソッドは、Webアプリケーション終了時に呼び出されます。 V5.1以前と同じタイミングで処理を行う場合には、カスタムタグを以下のように修正してdoFinally()メソッドに処理を記述してください。 javax.servlet.jsp.tagext.TryCatchFinallyインタフェースをimplementsして、以下のメソッドを実装します。 public void doCatch(Throwable t) public void doFinally()	
2	Webブラウザに返却されるステータスコード	Webアプリケーション環境定義ファイルに<error-page>タグが設定されている場合は、ステータスコードは200が返却されます。また、JSPのerrorPage属性に指定したJSPエラーページでisErrorPage属性に"true"が設定されている場合にも、ステータスコードは200です。	Webアプリケーション環境定義ファイルに<error-page>タグが設定されている場合にも、発生した現象のステータスコードがそのまま返却されます。また、JSPのerrorPage属性に指定したJSPエラーページでisErrorPage属性に"true"が設定されている	Webアプリケーション環境定義ファイルに<error-page>タグが設定されている場合にも、発生した現象のステータスコードがそのまま返却されます。また、JSPのerrorPage属性に指定したJSPエラーページでisErrorPage属性に"true"が設定されている

No.	機能	Tomcat3.1ベースのServletサービス	Tomcat4.1ベースのServletサービス	Tomcat5.5ベースのServletサービス
			場合には、ステータスコードは200です。	場合には、ステータスコードは500です。
3	JSPの呼び出し	JSPファイルを更新した場合、前回コンパイルされたJSPファイルとファイル更新日時が異なる場合に反映されます。(反映はServletコンテナ再起動、またはリロード機能有効時) Windows 大文字・小文字は区別されません。	JSPまたはJSPから静的includeしているファイルを更新した場合、前回コンパイルされた日時よりも新しい日付のJSPファイルの場合に反映されます。 “使用上の注意”の“Servletサービスの注意事項”も参照してください。 Windows 大文字・小文字は区別されます。	
4	Webアプリケーション環境定義ファイルの<servlet>タグ内の<load-on-startup>タグ	0を指定した場合は、サーブレット・コンテナ起動時にロードされません。	0を指定した場合は最後にロードされます。 負の数を指定した場合はWebアプリケーション起動(活性化)時にロードされません。	Servletの仕様に準拠し、0を指定した場合は最初にロードされます。 負の数を指定した場合はWebアプリケーション起動(活性化)時にロードされません。
5	Webブラウザにスタックトレースを表示	サーブレット・コンテナ環境定義ファイルのContextManagerタグのerrResponse属性で、リクエストの処理でErrorやExceptionが発生した場合、Webブラウザにスタックトレースを表示するかどうかを指定します。	内部の情報が漏洩する可能性があるため、リクエストの処理でErrorやExceptionが発生した場合、Webブラウザにスタックトレースを表示しません。	
6	Webアプリケーション環境定義ファイルの<servlet-mapping>タグ内の<servlet-name>タグ	指定していない名前を記述した場合は、無効となります。	指定していない名前を記述した場合は、Webアプリケーションの起動に失敗します。	
7	Webアプリケーション環境定義ファイルの<welcome-file-list>タグ内の<welcome-file>タグ	welcome fileを省略した場合、サーブレット・コンテナ環境定義ファイルで記述するContext定義のdirList属性の指定により、その実体となるディレクトリ配下のディレクトリやファイルの一覧、またはステータスコード404(ファイルが存在しない)が表示されます。	welcome fileを省略した場合、デフォルト設定が使用されます。デフォルト設定のファイルは以下の順番で有効になります。 1. index.html 2. index.htm 3. index.jsp welcome file(省略時はデフォルト設定)に該当するファイルがない場合は、Interstage管理コンソールの[ワークユニット] > “ワークユニット名” > [環境設定]タブ > [詳細設定] > [Servletコンテナ設定] > [ファイルの一覧表示]に設定されている値によって、ステータスコード404(ファイルが存在しない)、またはその実体となるディレクトリ配下のディレクトリやファイルの一覧が表示されます。	

No.	機能	Tomcat3.1ベースのServletサービス	Tomcat4.1ベースのServletサービス	Tomcat5.5ベースのServletサービス
8	Webアプリケーション環境定義ファイルの<env-entry>タグ内の<env-entry-value>タグ	<env-entry-value>タグを省略して環境エントリをlookupした場合には、javax.naming.NamingExceptionまたはそのサブクラスが投げられます。	<env-entry-value>タグを省略した場合には、デフォルト値が設定されます。(注1) Tomcat3.1ベースのServletサービスと同じ動作をさせる場合には、Interstage管理コンソールの[ワークユニット]>“ワークユニット名”>[環境設定]タブ>[詳細設定]>[ワークユニット設定]>[JavaVMオプション]に“-Dcom.fujitsu.interstage.jservlet.naming.defaultenventry=FALSE”を設定してください。	Tomcat3.1ベースのServletサービスと同じ動作をさせる場合には、deployment descriptor(web.xml)から、該当する<env-entry>タグを削除してください。
9	Webアプリケーション環境定義ファイルの<taglib-location>タグで存在しないパスを指定した場合	無効となります。	Webアプリケーションの起動に失敗します。	
10	WEB-INF配下にXMLの文法に誤りのあるTagLibraryDescriptorファイルが存在した場合	“XML parsing error on file”がコンテナの標準出力、標準エラー出力のログに出力されるが、サーブレット・コンテナの起動に成功します。	メッセージ“IJServer15111”がコンテナログに出力され、Servletコンテナの起動に失敗します。	
11	Webアプリケーション環境定義ファイルの<servlet-mapping>タグ内の<url-pattern>タグ	特定の拡張子をもつURLを指定した場合(例:*.jsp)は、URL内の“/”と“/”で囲まれた箇所もマッピング対象となります。	特定の拡張子をもつURLを指定した場合(例:*.jsp)は、URL内の“/”と“/”で囲まれた箇所もマッピング対象とならず、最後の“/”以降がマッピング対象となります。	
12	Webアプリケーション環境定義ファイルの<welcome-file-list>タグ内の<welcome-file>タグに指定されたファイルを表示する場合の動作	- URLが“/”で終わっていない場合、URLの最後に“/”を追加した値をLocationヘッダに指定し、ステータスコード:302を返します。 - URLが“/”で終わっている場合、<welcome-file>タグに指定されたコンテンツに内部でforwardします。そのため、指定したコンテンツ(JSP等)先で取得できるリクエストのURLなどのパス情報は、forwardされたコンテンツのものとなります。	- URLが“/”で終わっていない場合、URLの最後に“/”を追加した値をLocationヘッダに指定し、ステータスコード:302を返します。 - URLが“/”で終わっている場合、<welcome-file>タグに指定されたファイル名を追加したURLをLocationヘッダに指定し、ステータスコード:302を返します。そのため、指定したコンテンツ先で取得できるリクエストのURLなどのパス情報は、ブラウザが再度リクエストを	- URLが“/”で終わっていない場合、URLの最後に“/”を追加した値をLocationヘッダに指定し、ステータスコード:302を返します。 - URLが“/”で終わっている場合、<welcome-file>タグに指定されたコンテンツを直接処理して返します。そのため、指定したコンテンツ先で取得できるリクエストのURLなどのパス情報は、ブラウザからリクエストされてきたパスのままとなり、<welcome-file>タグで指定したコンテンツ

No.	機能	Tomcat3.1ベースのServletサービス	Tomcat4.1ベースのServletサービス	Tomcat5.5ベースのServletサービス
			発行(リダイレクト)した時の情報となります。	を含むパスとはなりません。 (注3)
13	複数のAccept-Languageヘッダを持つリクエストを受け付けた場合のHttpServletRequest#getHeaders()メソッドの動作 (注2)	- version属性(※)に4.1を指定または省略している場合 “,”で区切られた値を1つの要素としてEnumerationが返されます。 例) ja, es - version属性(※)に5.0または5.1を指定している場合 1つ1つの値が1つの要素としてEnumerationが返されます。 例) ja es ※サーブレット・コンテナ環境定義ファイルのContextManagerタグ内	“,”で区切られた値を1つの要素としてEnumerationが返されます。 例) ja, es	
14	レスポンスのデフォルトLocale	"en"です。	"en_US"です。	Servletの仕様に準拠し、デフォルトではサーバのLocaleとなります。
15	レスポンスのコミット後またはgetWriterメソッド呼び出し後の、setLocaleメソッドまたはsetContentTypeメソッド呼び出しのレスポンスのエンコーディングへの反映 (注4)	「setLocaleメソッドまたはsetContentTypeメソッド呼び出しのレスポンスのエンコーディングへの反映」を参照してください。		
16	レスポンスのsetLocaleメソッドまたはsetContentTypeメソッド(引数のmimeタイプ文字列にcharset属性を含む) (注5)によりレスポンスのエンコーディングを設定済みで、setContentTypeメソッドをcharset属	レスポンスのWriterによるデータの書き出しには反映されますが、レスポンスのContent-Typeヘッダにはcharset属性は付加されません。		Servletの仕様に従い、レスポンスのWriterによるデータの書き出し、Content-Typeヘッダのcharset属性ともに、先に設定済みのエンコーディングが反映されます。

No.	機能	Tomcat3.1ベースのServletサービス	Tomcat4.1ベースのServletサービス	Tomcat5.5ベースのServletサービス
	性を含まないmimeタイプ文字列を指定して呼び出した場合の動作 (注6)			
17	レスポンスのsetContentTypeメソッドをcharset属性ありのmimeタイプ文字列を指定して呼び出した (注5)後の、setLocaleメソッドのレスポンスのエンコーディングへの反映 (注6)	反映されます。		Servlet2.4の仕様に従い、反映されません。
18	setLocaleメソッドまたはsetContentTypeメソッドでエンコーディングを指定しない場合のレスポンスのエンコーディング	レスポンスのWriterによるデータの書き出しはISO-8859-1となりますが、レスポンスのContent-Typeヘッダに、charset属性は付加されません。		レスポンスのWriterによるデータの書き出し、およびContent-Typeヘッダのcharset属性はISO-8859-1となります。
19	JSPのpageディレクティブで明示的にpageEncodingまたはcontentType属性を指定しない場合のデフォルトエンコーディング	レスポンスのデータの書き出しはISO-8859-1、Content-Typeヘッダは"text/html"となります。	レスポンスのデータの書き出しはISO-8859-1、Content-Typeヘッダは"text/html;charset=ISO-8859-1"となります。	
20	サーブレットのserviceが、永久的に使用不能な状態であることを示すUnavailableExceptionをthrowした時の動作	Servlet2.3では規定されていませんが、基本的にステータスコード500を返却。	Servlet2.3では規定されていませんが、基本的にステータスコード503を返却。	Servlet2.4の仕様に従い、ステータスコード404を返却。
21	deployment descriptor(web.xml)のtaglibタグ	<taglib>タグは、<web-app>タグの直下に記載します。	<taglib>タグは、<web-app>タグの直下に記載します。	<taglib>タグは、<jsp-config>タグ配下に記載します。
22	server.xml定義ファイルへのrealmの設定	—	Tomcat4.1と同様のrealmタグを設定することができます。	できません。ディレクトリサービスを使用してください。
23	「コンテキストの共有」有効時のセッション	—	Webアプリケーション環境設定の「コンテキストの共有」を「する」に設定し、Webアプリケーションから別のWebアプリケーションに、RequestDispatcherを使用してディスパッチした場合、ディスパッチ前と後で使用するセッションは同じとなります。	Webアプリケーション環境設定の「コンテキストの共有」を「する」に設定し、Webアプリケーションから別のWebアプリケーションに、RequestDispatcherを使用してディスパッチした場合、Servletの仕様に厳密に従い、セッションは別のものとなります。セッションの

No.	機能	Tomcat3.1ベースのServletサービス	Tomcat4.1ベースのServletサービス	Tomcat5.5ベースのServletサービス
				属性は共有されません。ディスパッチ前のセッションの情報(属性)をディスパッチ後に使用したい場合は、リクエストの属性として (ServletRequest#setAttribute) メソッド等を使用して) 受渡を行う等、アプリケーションで対応してください。
24	HttpSessionの getAttribute(null) 呼び出し時	NullPointerExceptionとなります。	nullが返却されます。	NullPointerExceptionとなります。 Servletの仕様では、HttpSessionの setAttributeメソッドのキーにnullは指定できません。 getAttributeしても値の取得はできませんので、キーがnullの場合は呼び出さないようにしてください。
25	JSPのコンパイル <jsp:useBean>タグ のclass属性	JSPの仕様に沿ってなくてもコンパイル/実行できる場合があります。		JSP2.0の仕様に沿っていない場合は動作しない可能性があります。 仕様に沿うよう修正することを推奨します。 以下のような修正が必要です。 ・ フルクラス名で指定する ・ publicなクラスを指定する ・ publicなデフォルトコンストラクタがあるクラスを指定する 修正しない場合は、事前に充分な動作検証を行ってください。 JSP2.0の仕様の詳細は、JavaServer Pages 2.0 Specificationを参照してください。
26	HttpSessionListenerの sessionDestroyed	セッションが破棄された後に呼び出されます。		セッションが破棄される前に呼び出されます。 詳細はServlet2.4仕様を参照してください。
27	HttpSessionBindingListenerの valueBound	オブジェクトがHttpSessionのgetAttributeで取得できる状態になってから呼び出されます。		オブジェクトがHttpSessionの getAttributeで取得できる状態になる前に呼び出されます。 またこの影響により、HttpSessionの

No.	機能	Tomcat3.1ベースのServletサービス	Tomcat4.1ベースのServletサービス	Tomcat5.5ベースのServletサービス
				setAttributeメソッドですでに存在するキーで別のオブジェクトをsetした場合、先に新しいオブジェクトのvalueBoundメソッド、後からもとのオブジェクトのvalueUnboundメソッドが呼び出されます。 詳細はServlet2.4仕様を参照してください。
28	Webアプリケーション環境定義ファイルのタグの順番	<env-entry>タグ配下は以下の順で記載します。 <pre> <env-entry> <env-entry-name> entry-name </env-entry-name> <env-entry-value> entry-value </env-entry-value> <env-entry-type> entry-type </env-entry-type> </env-entry> </pre>		<env-entry>タグ配下は以下の順で記載します。 <pre> <env-entry> <env-entry-name> entry-name </env-entry-name> <env-entry-type> entry-type </env-entry-type> <env-entry-value> entry-value </env-entry-value> </env-entry> </pre>
29	JSPのタグの書式	以下の例のように、タグ～閉じタグ(ETag)間に子タグやBODYがなく、改行などがあっても動作する場合があります。 例: <pre> <jsp:forward page="next.jsp" > </jsp:forward> </pre>		JSP2.0では許可される書式が明示され、左記のような記述はコンパイル時にエラーとなります。JSPの仕様に従い、以下のようになしてください。 例: <pre> <jsp:forward page="next.jsp" ></ jsp:forward> または <jsp:forward page="next.jsp" /> </pre> JSP2.0の仕様の詳細は、JavaServer Pages 2.0 Specificationを参照してください。
30	JSPのBeanのプロパティ値がnullの場合の動作	JSPのBeanのプロパティ値がnullの場合、<jsp:getProperty>アクションの出力は""(空文字列)となります。 例:以下のようなJSPで、プロパティ"age"値がnullの場合 <pre> <jsp:useBean id="eb" class="pkg.EmptyBean" scope="session" /> <jsp:getProperty name="eb" property="age" /> </pre>		JSPの仕様に従い、出力は"null"となります。 以前のServletサービスと同じ出力としたい場合は、以下のいずれかを行ってください。 プロパティの初期値を""とするなど、JSPの仕様に従うようにアプリケーションを修正する。(推奨)

No.	機能	Tomcat3.1ベースのServletサービス	Tomcat4.1ベースのServletサービス	Tomcat5.5ベースのServletサービス
				Interstage 管理コンソールの[ワークユニット]>“ワークユニット名”>[環境設定]タブ>[詳細設定]>[ワークユニット設定]>[JavaVMオプション]に“-Dcom.fujitsu.interstage.jservlet.jsp.bean.getProperty.UseEmptyStringForNull=true”を設定する。
31	Webアプリケーション環境定義ファイルの<security-constraint>タグ内の<auth-constraint>タグ内の<role-name>タグ	"*"を指定する場合、<security-role>タグに有効な<role-name>タグが定義されていなければリソースにアクセスできません。	"*"を指定する場合、<security-role>タグを定義しなくてもリソースにアクセスできます。	"*"を指定する場合、<security-role>タグに有効な<role-name>タグが定義されていなければリソースにアクセスできません。
32	JSPのファイルエンコーディング	JSP1.2に従い、JSPファイルを読み込むエンコーディングは、コンパイル単位でした。include先のJSPファイルも、include元とおなじエンコーディングで読み込まれます。 include先のJSPがinclude元のJSPファイルのエンコーディングに依存していた場合、Tomcat5.5ベースのServletサービスでは、include先のJSPファイルにエンコーディングを明に設定することで、同様の動作とすることが可能です。 例:deployment descriptor(web.xml)をServlet2.4のものに修正し、JSPプロパティグループで指定する。 <pre><jsp-property-group> <url-pattern>/ja/*</url-pattern> <page-encoding>Shift_JIS</page-encoding> </jsp-property-group></pre> 詳細は、JSP2.0の仕様(JavaServer Pages 2.0 Specification)を参照してください。		JSP2.0に従い、JSPファイルを読み込むエンコーディングは、ファイル単位となります。include先のJSPファイルを読み込むエンコーディングは、include元のJSPファイルを読み込むエンコーディングの影響を受けません。(デフォルトISO-8859-1)
33	JSPでカスタムタグライブラリ使用時の動作	- TLD(Tag Library Description file)にて、対象のタグのbody-contentに“TAGDEPENDENT”(大文字)指定でも動作可能です。 - XML構文のJSP(JSP Documents)の場合、TLD(Tag Library Description file)にて、対象のタグのbody-contentに“tagdependent”(“TAGDEPENDENT”)を指定し、タグのボディで<jsp:text>などを使用した場合、ボディは処理された後にタグライブラリに渡されます。		- TLD(Tag Library Description file)にて、対象のタグのbody-contentに“TAGDEPENDENT”(大文字)は指定できません。仕様に従い、“tagdependent”(小文字)を指定してください。 - XML構文のJSP(JSP Documents)の場合、TLD(Tag Library

No.	機能	Tomcat3.1ベースのServletサービス	Tomcat4.1ベースのServletサービス	Tomcat5.5ベースのServletサービス
				Description file)にて、対象のタグのbody-contentに“tagdependent”を指定し、タグのボディで<jsp:text>などを使用した場合、ボディは処理されずにそのままタグライブラリに渡されます。仕様通りの正しい動作です。 Tomcat3.1ベース / Tomcat4.1ベースのServletサービスと同じ動作をさせたい場合は、TLD(Tag Library Description file)にて、body-contentに“JSP”を指定してください。
34	サーブレットコンテキストの初期化パラメタ	サーブレットコンテキストの初期化パラメタ(<context-param>タグ)の複数指定時、重複したパラメタ名(<param-name>タグ)がある場合、最後に指定したものが有効となります。		重複したパラメタ名は指定できません。定義エラーとなります。
35	無効なセッションのgetLastAccessedTimeメソッド呼び出し	無効なセッション(破棄やタイムアウトしたセッション)のgetLastAccessedTimeメソッドを呼び出し可能です。		無効なセッション(破棄やタイムアウトしたセッション)のgetLastAccessedTimeメソッドを呼び出した場合、IllegalStateExceptionとなります。

注1)

以下のデフォルト値が設定されます。

env-entry-type	エントリ値
java.lang.Boolean	Boolean.FALSE
java.lang.Byte	値0のByteオブジェクト
java.lang.Character	値0のCharacterオブジェクト
java.lang.String	なし 注) javax.naming.NamingExceptionまたはそのサブクラスが投げられます。
java.lang.Short	値0のShortオブジェクト
java.lang.Integer	値0のIntegerオブジェクト
java.lang.Long	値0のLongオブジェクト
java.lang.Float	値0のFloatオブジェクト
java.lang.Double	値0のDoubleオブジェクト

注2)

例)

Accept-Language: ja

Accept-Language: es

注3)

Tomcat4.1ベースのServletサービスからの移行の場合、<welcome-file>でディレクトリ階層の異なるパス(途中に/を含むパス)を指定し、そのパスで示されるコンテンツから相対パスのリンクがあると、リンク先が意図したURIとならず、コンテンツが参照できない場合があります。以下のどちらかの対応を行ってください。

- <welcome-file>でディレクトリ階層の異なるコンテンツを指定しない。
- <welcome-file>で指定したページ中のリンクについて、コンテキストパスを含むパスとする。

例) <welcome-file>が“/jsp/welcome.jsp”、Webアプリケーション名が“sample”、JSPからのリンクが“../next.jsp”の場合、リンク先を、“/sample/next.jsp”とする。

動的コンテンツ(JSPやServlet)の場合は、以下のようにコンテキストパスを動的に取得することも可能です。(推奨)

リンク先を、request.getContextPath()+"/next.jsp" とする。

※実際には、必要に応じてHttpServletResponseのencodeURLメソッド等をあわせて使用してください。

注4)

仕様に従っていないアプリケーションの場合であり、記載のTomcat3.1ベース/Tomcat4.1ベースのServletサービスの動作が保障されるものではありません。

注5)

JSPのpageディレクティブでpageEncoding属性を指定している場合、または、contentType属性にcharsetを指定している場合も含まれます。

注6)

No.15が前提です。

setLocaleメソッドまたはsetContentTypesメソッド呼び出しのレスポンスのエンコーディングへの反映

		Tomcat3.1ベースのServletサービス		Tomcat4.1ベースのServletサービス		Tomcat5.5ベースのServletサービス	
getWriterメソッド	レスポンスのコミット	Content-Typeヘッダのcharset属性	データの書き出し	Content-Typeヘッダのcharset属性	データの書き出し	Content-Typeヘッダのcharset属性	データの書き出し
呼び出し前	コミット前	反映される	反映される	反映される	反映される	反映される	反映される
	コミット後	反映されない	反映される	反映されない	反映されない	反映されない	反映されない
呼び出し後	コミット前	反映される	反映されない	反映される	反映される	反映されない	反映されない
	コミット後	反映されない	反映されない	反映されない	反映されない	反映されない	反映されない

4.3 InfoDirectoryからInterstage ディレクトリサービスへの移行

本製品では、InfoDirectoryを提供していません。以下を参照して、InfoDirectoryからInterstage ディレクトリサービスに移行してください。ここでは、InfoDirectoryからInterstage ディレクトリサービスへ移行する際の対処について説明します。

なお、ディレクトリサーバとしての基本的な動作は移行できますが、InfoDirectoryの機能の一部は、Interstage ディレクトリサービスでは使用できません。この場合、Interstage ディレクトリサービスにおいて該当する機能を使用しないで運用してください。

Interstage ディレクトリサービスで使用できない機能については“[4.3.1 InfoDirectoryとInterstage ディレクトリサービスの機能比較](#)”、および“[4.3.2 Interstage ディレクトリサービスへ移行できないデータ](#)”を参照してください。

4.3.1 InfoDirectoryとInterstage ディレクトリサービスの機能比較

InfoDirectoryとInterstage ディレクトリサービスの機能比較一覧を以下に示します。

○:機能あり ×:機能なし

カテゴリ	概要	項目	InfoDirectory ディレクトリ サービス	Interstage ディレクトリ サービス
認証	クライアントからディレクトリサーバへのLDAPによる認証機能	匿名認証	○	○
		名前のみによる認証	○	×
		名前と非暗号化パスワードによる認証	○	○
		SASLによる認証 SMEE - SSLV3	○	○
パスワードの暗号化	ディレクトリサーバに登録されたユーザエントリパスワードの暗号化機能	暗号化する	○	○
		暗号化しない	○	○
アクセス制御	各ツリーおよびエントリに設定するアクセス制御機能		○	○
SSL	クライアントとディレクトリサーバ間の通信路のセキュリティ機能	SSL簡易認証	○	○
		SSL証明書認証	○	×
ポリシー	ディレクトリサーバ自体にアクセスする場合のセキュリティポリシーを設定する機能		○	×
シャドウイング	X.500 DISPプロトコルによるレプリケーション機能	レプリケーションの主導(サプライヤ:複製元)	○	○(LDAP)
		レプリケーションの主導(コンシューマ:複製先)	○	×
		レプリケーション範囲(全ツリー)	○	○(LDAP)
		レプリケーション範囲(特定ツリー配下)	○	×
		レプリケーション範囲(特定エントリ)	○	×
		レプリケーション範囲(特定属性のみ)	○	×
		レプリケーションのスケジューリング(即時)	○	○(LDAP)
		レプリケーションのスケジューリング(開始時間/実行間隔等)	○	×
チェイニング	X.500 DSPプロトコルによるサーバ間連携機能	各サブツリーをマスタとして、各サーバ間で上位ツリー、下位ツリー参照を行いクライアントからの検索要求に対して、サーバ間で連携し、結果を通知する。	○	×

カテゴリ	概要	項目	InfoDirector yディレクトリ サービス	Interstage ディレクトリ サービス
リフェラル	クライアントに、他参照先ディレクトリサーバ情報を通知する機能	デフォルトリフェラル	○	×
		エントリリフェラル	○	×
スキーマ	実装スキーマ	RFC2256: A Summary of the X.500(96) User Schema for use with LDAPv3	○	○
		RFC2307: An Approach for Using LDAP as a Network Information Service	○	○
		RFC2798: Definition of the inetOrgPerson LDAP Object Class	○	○
		RFC1274: The COSINE and Internet X.500 Schema	○	○
		RFC2079: Definition of an X.500 Attribute Type and an Object Class to Hold Uniform Resource Identifiers (URIs)	○	○
		RFC2587: Internet X.509 Public Key Infrastructure LDAPv2 Schema	○	○
	スキーマ拡張機能	オブジェクトクラス定義	○	○
		属性定義	○	○
		ネームバインド定義	○	×
ログ	採取ログ	アクセスログ	○	○
		オペレーションログ(管理ツール)	○	×
		変更ログ	○	×
		エラーログ	○	○
		スケジューラログ(プロセスのスケジュール)	○	○
		アップデイトログ(シャドウ、DB復旧)	○	○
		トラフィックログ(X.500へのアクセス)	○	×
		トレースログ(プロセス毎)	○	○
	ログ監視設定機能	ログファイルのバックアップ/削除	○	○
		自動停止/開始(動作中の指示)	○	×
		ログレベル設定	○	○
		他(ログファイルのサイクリック化)	○	○
バックアップ/ リストア	DSA(リポジトリ)	DSA資源のバックアップ	○	○

カテゴリ	概要	項目	InfoDirector yディレクトリ サービス	Interstage ディレクトリ サービス
		DSAデータベースのバックアップ	○	○
	管理ツール	管理ツール資源	○	×
	証明書環境	証明書環境資源	○	×
Webコネクタ	Webクライアントからディレクトリサーバにアクセスする機能		○	×
LDAPコマンド		追加/更新(ldapmodify)	○	○
		削除(ldapdelete)	○	○
		検索(ldapsearch)	○	○
		DIT複製(ldapreplica)	○	×
SDK	ユーザアプリケーションインタ フェース	C言語API	○	○
		C++言語API	×	×
		Java言語API(JNDI)	○	○
		LDAPコントロール(LDAP V3 拡張要求)	○	×
検索	検索フィルタ式	同値	○	○
		部分文字列	○	○
		以上	○	○
		以下	○	○
		存在	○	○
		近似	○	×
		否定	○	×(標準DB) ○(RDB)
		論理積	○	○
		論理和	○	○(標準DB は2つまで)
		ワイルドカード指定	○ (制限なし)	○(標準DB は1つまで)
比較	リポジトリに登録されているエ ントリの属性値と比較する機能	暗号化されていないパスワード との比較	○	○
		暗号化されているパスワードと の比較	○	×
その他	1台のサーバに複数DSA(リポジトリ)作成		○	○
	ディレクトリサーバチューニング		○	○

標準DB

リポジトリのデータベースとして標準データベース(Fujitsu Enabler)を使用する場合

RDB

リポジトリのデータベースとしてRDB(Symfoware/RDB、またはOracleデータベース)を使用する場合
(RDBは、Interstage Application Server Enterprise Editionだけで利用できます。)

4.3.2 Interstage ディレクトリサービスへ移行できないデータ

以下に、InfoDirectoryからInterstage ディレクトリサービスへ移行できないデータを示します。
以下のデータは、Interstage ディレクトリサービスでは利用できません。

- アクセス制御情報
- ネームバインド
- 知識情報(チェイニング機能)
- リフェラル情報
- 拡張スキーマ
- 環境ファイル

4.3.3 Interstage ディレクトリサービスへの移行手順

InfoDirectoryディレクトリサービスからInterstage ディレクトリサービスへ、ディレクトリ内のデータを移行する手順を示します。

1. [InfoDirectoryからの移行に関する注意事項を確認](#)
2. [InfoDirectory内からデータを抽出](#)
3. [Interstage ディレクトリサービスでリポジトリを作成](#)
4. [リポジトリの起動](#)
5. [Interstage ディレクトリサービスへのデータの移入](#)

1) InfoDirectoryからの移行に関する注意事項を確認

Interstage ディレクトリサービスとInfoDirectoryでは、エントリを管理するうえで、いくつか異なる点があります。“使用上の注意”の“Interstage ディレクトリサービスの注意事項”－“InfoDirectoryからの移行に関する注意事項”を参照し、注意事項を確認してください。

2) InfoDirectory内からデータを抽出

移行するデータが存在するDSAに対して以下のコマンドを実行し、InfoDirectoryのデータをLDIFファイルへ取り出します。コマンドの詳細は、ご使用になっている製品に添付のマニュアル“InfoDirectory使用手引書”の“LDAPコマンド”を参照してください。

以下の例では、データをLDIFファイル“dir.ldif”へ取り出しています。



例

Windows

```
C:\Interstage\ID\Dir\%sdk%\bin\ldapsearch -h ホスト名 -p LDAPポート番号 -D DSAの管理者DN -w DSAの管理者DNのパスワード -b トップエン트리 "(objectclass=*)" > dir.ldif
```

Solaris

Linux

```
/opt/FJSSVidsdk/C/bin/ldapsearch -h ホスト名 -p LDAPポート番号 -D DSAの管理者DN -w DSAの管理者DNのパスワード -b トップエン트리 "(objectclass=*)" > dir.ldif
```

- LDAPポート番号、DSAの管理者DN、およびトップエン트리については、InfoDirectory管理ツールクライアントで確認してください。
- ldapsearchコマンドは、InfoDirectoryのDSAが動作するホスト上で実行してください。

3) Interstage ディレクトリサービスでリポジトリを作成

Interstage ディレクトリサービスを構築するマシン上でリポジトリを作成します。リポジトリの作成は、Interstage管理コンソールを使用して、[システム] > [サービス] > [リポジトリ] > [新規作成]タブから行います。リポジトリの作成は、“ディレクトリサービス運用ガイド”の“環境構

築”を参照してください。

なお、リポジトリ作成時に指定する[公開ディレクトリ]には、以下の値を指定してください。

- 公開ディレクトリ
InfoDirectoryのDSAで指定されていたトップエントリを指定してください。

4) リポジトリの起動

作成したリポジトリを起動します。リポジトリの起動は、Interstage管理コンソールを使用して、[システム] > [サービス] > [リポジトリ] > [リポジトリ:状態]画面から行います。

5) Interstage ディレクトリサービスへのデータの移入

起動したリポジトリに対して以下のコマンドを実行し、InfoDirectoryから取り出したデータをInterstage ディレクトリサービスへ移入します。コマンドの詳細は、“リファレンスマニュアル(コマンド編)”の“Interstage ディレクトリサービス運用コマンド”を参照してください。

以下の例では、InfoDirectoryからLDIFファイル“dir.ldif”へ取り出したデータを移入しています。



例

Windows

```
C:\¥Interstage¥bin¥ldapmodify -h ホスト名 -p 通常ポート番号(またはSSLポート番号) -D リポジトリの管理者用DN -w リポジトリの管理者用DNのパスワード -c -a -f dir.ldif
```

Solaris

Linux

```
/opt/FJSVirepc/bin/ldapmodify -h ホスト名 -p 通常ポート番号(またはSSLポート番号) -D リポジトリの管理者用DN -w リポジトリの管理者用DNのパスワード -c -a -f dir.ldif
```

- 通常ポート番号(またはSSLポート番号)、リポジトリの管理者用DNには、リポジトリの作成時に指定した値を指定してください。または、Interstage管理コンソールのリポジトリタグで、作成したリポジトリの設定項目を確認してください。
- ldapmodifyコマンドは、Interstage ディレクトリサービスが動作するホスト上で実行してください。

4.3.4 C言語アプリケーションの移行手順

C言語アプリケーションの移行手順について説明します。

■アプリケーションの再コンパイルについて

InfoDirectory SDKを使用したユーザアプリケーションは、Interstage ディレクトリサービス SDKとバイナリの互換性がありません。このため、アプリケーションを再コンパイルする必要があります。以降の記事を参照したうえで、アプリケーションを再コンパイルしてください。また、一部の機能、インタフェースなどに変更がありますので、以下を参照して、アプリケーションを修正してください。

- 機能差異
- 推奨されないインタフェース
- サポートされないインタフェース
- 開発環境の変更
- ライブラリについて

■機能差異

旧バージョン・レベルのAPIでは“draft-ietf-ldapext-ldap-c-api-05.txt”に準拠していますが、一部、古い仕様である“draft-ietf-ldapext-ldap-c-api-00.txt”に従って実装されているものがあります。

Interstage ディレクトリサービス LDAPクライアントAPIライブラリでは、最新の規約に正しく準拠するため、以下のAPIで機能差異が発生します。

- ・ 関数仕様
- ・ 関数宣言

◆関数仕様

関数	変更点	移行方法
ldap_get_option()	セッションハンドルオプションの扱いが変更されています。	
LDAP_OPT_DESC	推奨されないインタフェースに変更されました。	推奨されませんが、従来どおり利用できます。
LDAP_OPT_HOST_NAME LDAP_OPT_ERROR_STRING	アプリケーションが復帰領域をldap_memfree()で解放する必要があります。解放しない場合は、メモリリークとなります。	ldap_get_option()で復帰したアドレスを不要になったタイミングでldap_memfree()で解放してください。
LDAP_OPT_RESULT	パラメタが、"void *"から"int *"に変更されました。旧APIでint型を使用していない場合は、アプリケーションが異常終了する可能性があります。	ldap_get_option()の第3パラメタはint型変数のポインタを指定してください。
ldap_set_option()	セッションハンドルオプションの扱いが変更されています。	
LDAP_OPT_HOST_NAME	パラメタが、"char *"から"char *"に変更されました。旧APIのままでは、アプリケーションが異常終了します。	例1)のように修正してください。
LDAP_OPT_ERROR_NUMBER LDAP_OPT_ERROR_STRING	旧バージョン・レベルでは、ldap_get_option()で参照だけでしたが、新バージョン・レベルでは、ldap_set_option()で更新できるようになりました。	基本的に修正不要ですが、アプリケーションの仕様に依拠して修正してください。
LDAP_OPT_RESULT	旧バージョン・レベルでは、パラメタがマニュアルでは"void *"でしたが、ライブラリでは"int *"で扱っていました。新バージョン・レベルでは、ライブラリでも"void *"で扱います。従来の指定方法のままでは、常にLDAP_OPT_ONが設定されたと認識されるため、アプリケーションが正しく動作しない可能性があります。	例2)のように修正してください。
ldap_version()	構造体メンバのsdk_versionが、旧バージョン・レベルでは100を返しますが、ライブラリのバージョンとして200以上を返します。	構造体のメンバsdk_versionをチェックしている場合は修正してください。
ldap_search() ldap_search_s() ldap_search_st() ldap_search_ext() ldap_search_ext_s()	filterパラメタにNULLを指定した場合、旧バージョン・レベルではパラメタエラー(LDAP_PARAM_ERROR)となりますが、新バージョン・レベルでは、"(objectClass=*)"を指定したとみなします。	基本的に修正不要ですが、アプリケーションの仕様に依拠して修正してください。
ldap_search_st() ldap_search_ext() ldap_search_ext_s()	timeoutパラメタの構造体メンバのtv_sec、tv_usecともに0を指定した場合、または、どちらか一方、または両方に負の値を設定した場合、旧バージョン・レベルでは無限に待ち合わせします。新バージョン・レベルではパラメタエラー(LDAP_PARAM_ERROR)となるため、旧APIのままでは検索できません。	タイムアウトを無制限にする場合は、timeoutパラメタにNULLを指定してください。 ただし、ldap_set_option()のLDAP_OPT_TIMELIMITを使用している場合は、“リファレンスマニュアル(API編)”の“C言語インタフェース”-“Interstage ディレクトリサービスインタフェース”-“セッション

関数	変更点	移行方法
		ハンドルオプションの設定/参照インタフェース”を参照してください。

例1)

旧バージョン・レベル

```
char *host = "localhost";
ldap_set_option( ld, LDAP_OPT_HOST_NAME, &host );
```

新バージョン・レベル

```
char *host = "localhost";
ldap_set_option( ld, LDAP_OPT_HOST_NAME, host );
```

例2)

旧バージョン・レベル

```
int value = LDAP_OPT_ON;
ldap_set_option( ld, LDAP_OPT_RESTART, &value );
```

新バージョン・レベル

```
ldap_set_option( ld, LDAP_OPT_RESTART, LDAP_OPT_ON );
```

◆関数宣言

以下の関数で、関数宣言が変更され、const修飾子(太字部分)により宣言が厳格化されました。旧APIに準じたソースではコンパイル時に警告が出る場合があります。必要に応じて修正してください。

- [ldap_init\(\)](#)
- [ldap_set_option\(\)](#)
- [ldap_simple_bind\(\)](#)
- [ldap_simple_bind_s\(\)](#)
- [ldap_search\(\)](#)
- [ldap_search_s\(\)](#)
- [ldap_search_st\(\)](#)
- [ldap_search_ext\(\)](#)
- [ldap_search_ext_s\(\)](#)
- [ldap_compare\(\)](#)
- [ldap_compare_s\(\)](#)
- [ldap_compare_ext\(\)](#)
- [ldap_compare_ext_s\(\)](#)
- [ldap_modify\(\)](#)
- [ldap_modify_s\(\)](#)
- [ldap_modify_ext\(\)](#)
- [ldap_modify_ext_s\(\)](#)

- [ldap_rename\(\)](#)
- [ldap_rename_s\(\)](#)
- [ldap_add\(\)](#)
- [ldap_add_s\(\)](#)
- [ldap_add_ext\(\)](#)
- [ldap_add_ext_s\(\)](#)
- [ldap_delete\(\)](#)
- [ldap_delete_s\(\)](#)
- [ldap_delete_ext\(\)](#)
- [ldap_delete_ext_s\(\)](#)
- [ldap_get_values\(\)](#)
- [ldap_get_values_len\(\)](#)
- [ldap_explode_dn\(\)](#)
- [ldap_explode_rdn\(\)](#)
- [ldap_dn2ufn\(\)](#)

ldap_init()

```
LDAP *ldap_init(
    const char *hostname,
    int portno );
```

ldap_set_option()

```
int ldap_set_option(
    LDAP *ld,
    int option,
    const void *optdata );
```

ldap_simple_bind()

```
int ldap_simple_bind(
    LDAP *ld,
    const char *dn,
    const char *passwd );
```

ldap_simple_bind_s()

```
int ldap_simple_bind_s(
    LDAP *ld,
    const char *dn,
    const char *passwd );
```

ldap_search()

```
int ldap_search(
    LDAP *ld,
    const char *base,
    int scope,
    const char *filter,
    char **attrs,
    int attrsonly );
```

ldap_search_s()

```
int ldap_search_s(  
    LDAP *ld,  
    const char *base,  
    int scope,  
    const char *filter,  
    char **attrs,  
    int attrsonly,  
    LDAPMessage **res );
```

ldap_search_st()

```
int ldap_search_st(  
    LDAP *ld,  
    const char *base,  
    int scope,  
    const char *filter,  
    char **attrs,  
    int attrsonly,  
    struct timeval *timeout,  
    LDAPMessage **res );
```

ldap_search_ext()

```
int ldap_search_ext(  
    LDAP *ld,  
    const char *base,  
    int scope,  
    const char *filter,  
    char **attrs,  
    int attrsonly,  
    LDAPControl **serverctrls,  
    LDAPControl **clientctrls,  
    struct timeval *timeout,  
    int sizelimit,  
    int *msgidp );
```

ldap_search_ext_s()

```
int ldap_search_ext_s(  
    LDAP *ld,  
    const char *base,  
    int scope,  
    const char *filter,  
    char **attrs,  
    int attrsonly,  
    LDAPControl **serverctrls,  
    LDAPControl **clientctrls,  
    struct timeval *timeout,  
    int sizelimit,  
    LDAPMessage **res );
```

ldap_compare()

```
int ldap_compare(  
    LDAP *ld,  
    const char *dn,  
    const char *attr,  
    const char *value );
```

ldap_compare_s()

```
int ldap_compare_s(  
    LDAP *ld,
```

```
const char *dn,  
const char *attr,  
const char *value );
```

ldap_compare_ext()

```
int ldap_compare_ext(  
    LDAP *ld,  
    const char *dn,  
    const char *attr,  
    const struct berval *bvalue,  
    LDAPControl **serverctrls,  
    LDAPControl **clientctrls,  
    int *msgidp );
```

ldap_compare_ext_s()

```
int ldap_compare_ext_s(  
    LDAP *ld,  
    const char *dn,  
    const char *attr,  
    const struct berval *bvalue,  
    LDAPControl **serverctrls,  
    LDAPControl **clientctrls );
```

ldap_modify()

```
int ldap_modify(  
    LDAP *ld,  
    const char *dn,  
    LDAPMod **mods );
```

ldap_modify_s()

```
int ldap_modify_s(  
    LDAP *ld,  
    const char *dn,  
    LDAPMod **mods );
```

ldap_modify_ext()

```
int ldap_modify_ext(  
    LDAP *ld,  
    const char *dn,  
    LDAPMod **mods,  
    LDAPControl **serverctrls,  
    LDAPControl **clientctrls,  
    int *msgidp );
```

ldap_modify_ext_s()

```
int ldap_modify_ext_s(  
    LDAP *ld,  
    const char *dn,  
    LDAPMod **mods,  
    LDAPControl **serverctrls,  
    LDAPControl **clientctrls );
```

ldap_rename()

```
int ldap_rename(  
    LDAP *ld,  
    const char *dn,  
    const char *newrdn,  
    const char *newparent,
```

```

    int deleteoldrdn,
    LDAPControl **serverctrls,
    LDAPControl **clientctrls,
    int *msgidp );

```

ldap_rename_s()

```

int ldap_rename_s(
    LDAP *ld,
    const char *dn,
    const char *newrdn,
    const char *newparent,
    int deleteoldrdn,
    LDAPControl **serverctrls,
    LDAPControl **clientctrls );

```

ldap_add()

```

int ldap_add(
    LDAP *ld,
    const char *dn,
    LDAPMod **attrs );

```

ldap_add_s()

```

int ldap_add_s(
    LDAP *ld,
    const char *dn,
    LDAPMod **attrs );

```

ldap_add_ext()

```

int ldap_add_ext(
    LDAP *ld,
    const char *dn,
    LDAPMod **attrs,
    LDAPControl **serverctrls,
    LDAPControl **clientctrls,
    int *msgidp );

```

ldap_add_ext_s()

```

int ldap_add_ext_s(
    LDAP *ld,
    const char *dn,
    LDAPMod **attrs,
    LDAPControl **serverctrls,
    LDAPControl **clientctrls );

```

ldap_delete()

```

int ldap_delete(
    LDAP *ld,
    const char *dn );

```

ldap_delete_s()

```

int ldap_delete_s(
    LDAP *ld,
    const char *dn );

```

ldap_delete_ext()

```

int ldap_delete_ext(
    LDAP *ld,
    const char *dn,

```

```
LDAPControl **serverctrls,
LDAPControl **clientctrls,
int *msgidp );
```

ldap_delete_ext_s()

```
int ldap_delete_ext_s(
    LDAP *ld,
    const char *dn,
    LDAPControl **serverctrls,
    LDAPControl **clientctrls );
```

ldap_get_values()

```
char **ldap_get_values(
    LDAP *ld,
    LDAPMessage *entry,
    const char *attr );
```

ldap_get_values_len()

```
struct berval **ldap_get_values_len(
    LDAP *ld,
    LDAPMessage *entry,
    const char *attr );
```

ldap_explode_dn()

```
char **ldap_explode_dn(
    const char *dn,
    int notypes );
```

ldap_explode_rdn()

```
char **ldap_explode_rdn(
    const char *rdn,
    int notypes );
```

ldap_dn2ufn()

```
char *ldap_dn2ufn(
    const char *dn );
```

■推奨されないインタフェース

以下に示すインタフェースは、推奨されないインタフェースとなります。引き続き使用できますが、新インタフェースへの移行を推奨します。

推奨されないインタフェース	新インタフェース
ldap_ber_free()	ber_free()に変更となります。パラメタの変更はありません。
ldap_version()	ldap_get_option()のオプションにLDAP_OPT_API_INFOを指定して獲得します。(注)
ldap_get_option() ldap_set_option()	- LDAP_OPT_CONNTIME オプションを使用している場合は、LDAP_OPT_NETWORK_TIMEOUTを使用してください。修正時は、オプションの値を格納する領域optdataに指定するパラメタの形式を以下のように変更してください。 — ldap_get_option()の場合、 "int *"から"struct timeval *"に変更 — ldap_set_option()の場合、 "int *"から"struct timeval *"に変更

推奨されないインタフェース	新インタフェース
	LDAP_OPT_ERROR_NUMBERは、LDAP_OPT_RESULT_CODEとして再定義されました。今後は、LDAP_OPT_RESULT_CODEを使用してください。

(注)

通知される情報は、以下の形式です

```
typedef struct ldapapiinfo {
    int ldapai_info_version;
    int ldapai_api_version;
    int ldapai_protocol_version;
    char **ldapai_extensions;
    char *ldapai_vendor_name;
    int ldapai_vendor_version;
} LDAPAPIInfo;
```

【構造体メンバの設定値】

ldapai_info_version	本構造体のバージョン情報
ldapai_api_version	LDAP-SDKのバージョン情報
ldapai_protocol_version	LDAPプロトコルのサポートバージョン
ldapai_extensions	未使用(NULL)
ldapai_vendor_name	ベンダー名
ldapai_vendor_version	ベンダーの内部バージョン

■サポートされないインタフェース

InfoDirectoryで使用していたC言語アプリケーションを移行する場合は、一部サポートされない機能がありますので、必要に応じて、アプリケーションを修正してください。

Interstage ディレクトリサービスでサポートされない機能と、その対処方法を以下に示します。

サポートされない機能	対処方法
証明書ベースの認証機能 - ldap_sasl_bind() - ldap_sasl_bind_s()	ldap_simple_bind()、またはldap_simple_bind_s()による認証に修正してください。
リフェラル機能 - ldap_first_reference() - ldap_next_reference() - ldap_count_references() - ldap_parse_reference() - セッションハンドルのオプションの以下のオプション - LDAP_OPT_REFERRALS - LDAP_OPT_REFERRAL_HOP_LIMIT - LDAP_OPT_REFERRAL_REBIND - LDAP_OPT_REFERRAL_REBIND_MAX	リフェラルを含むメッセージは返却されないため、左記の関数を使用している場合は、関数を削除してください。 また、セッションハンドルのオプションの「LDAP_OPT_REFERRALS」に「LDAP_OPT_OFF」を設定するよう修正してください。 例) <pre>rtn = ldap_set_option(ld, LDAP_OPT_REFERRALS, LDAP_OPT_OFF);</pre>
LDAPプロトコル拡張機能 - ldap_extended_operation() - ldap_extended_operation_s() - ldap_parse_extended_result()	左記の関数を使用している場合は、LDAPプロトコル拡張を使用しないように、関数を削除してください。

サポートされない機能	対処方法
ber_bvfree()	
拡張コントロール機能 - ldap_create_paging_control() - ldap_create_more_paging_control() - ldap_create_sort_control() - ldap_control_free() - ldap_controls_free() - ldap_get_entry_controls() - セッションハンドルオプションのLDAP_OPT_SERVER_CONTROLS	左記の関数を削除し次の方法で修正してください。 - ページング機能を使用している場合は、フィルタ条件を変更し、分割して検索してください。 - サーバソート機能を使用している場合は、アプリケーションで受信結果をソートしてください。
エイリアス(別名参照) セッションハンドルオプションのLDAP_OPT_DEREF	エイリアスを使用している場合は、アプリケーションの左記の関数を削除し、Interstage ディレクトリサービスに同一エントリを複数登録してください。

■開発環境の変更

以下の位置が変更されました。

	InfoDirectory	Interstage ディレクトリサービス
インクルードファイル	Windows C:\Interstage\IDSDK\C\include Solaris Linux /opt/FJSVidsdk/C/include	Windows C:\Interstage\IREPSDK\include Solaris Linux /opt/FJSVirepc/include
リンク用ライブラリ	Windows C:\Interstage\IDSDK\C\lib Solaris Linux /opt/FJSVidsdk/C/lib	Windows C:\Interstage\IREPSDK\lib Solaris Linux /opt/FJSVirepc/lib

■ライブラリについて

以下が変更されました。

	InfoDirectory	Interstage ディレクトリサービス
動的ライブラリ	Windows libidldap2.dll Solaris Linux libidldap2.so 上記の名前が右記に変更されました。	Windows F3FMirepldap2.dll Solaris Linux libirepldap2.so (注) アプリケーションで使用するライブラリ名を上記に変更してください。
静的ライブラリ	Windows libidldap2.lib Solaris Linux libidldap2.a	Windows F3FMirepldap2.dll Solaris Linux libirepldap2.so

	InfoDirectory	Interstage ディレクトリサービス
	上記の静的ライブラリは提供されません。	上記の動的ライブラリを使用するように修正してください。

(注)

動的ライブラリは/usr/lib、/usr/lib64へのシンボリックリンクは設定されません。アプリケーション、または環境設定でライブラリへのパスを解決する必要があります。

4.3.5 Java言語(JNDI)アプリケーションの移行

Java言語(JNDI)アプリケーションの移行手順について説明します。

■サポートされないインタフェース

InfoDirectoryで使用していたJNDIアプリケーションを移行する場合は、一部サポートされない機能がありますので、必要に応じて、アプリケーションを修正してください。

Interstage ディレクトリサービスでサポートされない機能と、その対処方法を以下に示します。

サポートされない機能	対処方法
証明書ベースの認証機能 - java.naming.security.authentication - javax.naming.Context.SECURITY_AUTHENTICATION	左記の環境プロパティで"external"を使用している場合は、簡易認証("simple")に修正してください。
リフェラル機能 - java.naming.referral - javax.naming.Context.REFERRAL	リフェラルを含むメッセージは返却されないため、左記の環境プロパティを使用している場合は、環境プロパティを削除してください。 なお、リフェラル先が存在する場合は、例外(PartialResultException)を通知します。
拡張コントロール機能 - 検索結果のページング処理 - 検索結果のソート処理	左記を使用している場合は次の方法で修正してください。 “検索結果のページング処理”を使用している場合は、フィルタ条件を変更し、分割して検索してください。 “検索結果のソート処理”を使用している場合は、アプリケーションで受信結果をソートしてください。
エイリアス(別名参照) java.naming.Ldap.derefAliases	エイリアスを使用している場合は、アプリケーションの左記の環境プロパティを削除し、Interstage ディレクトリサービスに同一エントリを複数登録してください。

4.3.6 LDAPコマンドの移行

LDAPコマンドには、以下の変更がありますので、注意してください。

- ・ [コマンドの格納先](#)
- ・ [オプションの機能差異](#)
- ・ [メッセージ](#)
- ・ [LDIFの仕様](#)

■コマンドの格納先

コマンドの格納先を変更しました。

OS	InfoDirectory	Interstage ディレクトリサービス
Windows	C:\Interstage\ID\Dir\sdk\C\bin	C:\Interstage\bin
Solaris Linux	/opt/FJSVidsdk/C/bin	/opt/FJSVirepc/bin

■オプションの機能差異

Interstage ディレクトリサービスのLDAPクライアントSDKでは、LDAPの最新の規約に正しく準拠するため、InfoDirectory SDKのLDAPコマンドと比較して、LDAPコマンドのオプションに機能差異があります。機能差異となるLDAPコマンドのオプションについて、以下に説明します。

- [ldapdelete](#)
- [ldapmodify](#)
- [ldapsearch](#)

また、各コマンドの詳細情報の出力形式についても違いがあります。詳細は、“◆[-vオプションで出力する詳細情報の差異](#)”で説明しています。

ldapdelete

旧オプション	新オプション	移行方法
-H ヘルプ(Usage)の出力	-m ヘルプ(Usage)の出力	新オプションを指定してコマンドを実行してください。
-C type 文字コードの指定	-G type 文字コードの指定	新オプションを指定してコマンドを実行してください。
なし	-H uri リポジトリのホスト名、およびポート番号をURI形式で指定	新規追加オプション
なし	-r ツリー末端までの削除	新規追加オプション
なし	-V プログラムのバージョン情報を出力	新規追加オプション
-v 詳細情報の出力	-v 詳細情報の出力	機能は同じです。出力形式が違いますので、“◆ -vオプションで出力する詳細情報の差異 ”で確認してください。

ldapmodify

旧オプション	新オプション	移行方法
-H ヘルプ(Usage)の出力	-m ヘルプ(Usage)の出力	新オプションを指定してコマンドを実行してください。
-C type 文字コードの指定	-G type 文字コードの指定	新オプションを指定してコマンドを実行してください。
-b 属性値の外部ファイルからの読み込み	オプション廃止	LDIFファイル内での外部ファイルの指定方法を変更してください。 【旧オプション時の指定方法】 属性名: ファイル 【新指定方法】 属性名:< ファイルをURL形式で指定 ※従来から、上記での指定方法を推奨しています。

旧オプション	新オプション	移行方法
なし	-H uri リポジトリのホスト名、および ポート番号をURI形式で指 定	新規追加オプション
なし	-S file エラー結果をファイル出力	新規追加オプション
なし	-V プログラムのバージョン情報 を出力	新規追加オプション
-v 詳細情報の出力	-v 詳細情報の出力	機能は同じです。出力形式が違いま すので、“◆-vオプションで出力する詳 細情報の差異”で確認してください。

ldapsearch

旧オプション	新オプション	移行方法
-H ヘルプ(Usage)の出力	-m ヘルプ(Usage)の出力	新オプションを指定してコマンドを実行 してください。
-C type 文字コードの指定	-G type 文字コードの指定	新オプションを指定してコマンドを実行 してください。
-U path -tオプション指定で作成する ファイルの出力先指定	-T path -tオプション指定で作成する ファイルの出力先指定	新オプションを指定してコマンドを実行 してください。
-Q attr tmp配下に検索結果の属性 値をファイル出力	オプション廃止	-tオプションを使用してください。
-e 非ASCII文字をUTF-8で Base64符号化して出力	オプション廃止	なし。 用途がないため廃止します。
-E 指定属性をBase64で符号化 して出力	オプション廃止	なし。 用途がないため廃止します。
-S attr クライアントで属性attrについ て結果をソート(昇順)しま す。 attrパラメタが""(空文字)で ある場合はDNでソートされ ます。	-S attr クライアントで属性attrについ て結果をソート(昇順)しま す。 attrパラメタが""(空文字)で ある場合はDNでソートされ ます。 -Sオプションで指定するソー トのための属性を、attributes オプションで指定する検索 属性に含む必要があります。 含まない場合、ソートされま せん。	属性attrで結果をソート(昇順)するた めには、attributesオプションに、属性attr を指定してください。 例) ldapsearch -b "ou=interstage,o=fujitsu,dc=com" -H ldap://hostname:389 -D "cn=manager,ou=interstage,o=fujitsu, dc=com" -w admin -S sn "objectclass=inetorgperson" sn
-t tmp配下に検索結果をファイ ル出力。作成したファイルの 絶対パスを標準出力	-t tmp配下に検索結果をファイ ル出力。作成したファイルの URLを標準出力	機能は同じです。作成したファイルの 絶対パスの表現が違いますので、確認 してください。

旧オプション	新オプション	移行方法
filter filterオプションを省略した場合、エラーになります。	filter filterオプションを省略した場合、"(objectClass=*)"を指定したとみなします。	基本的に修正不要です。必要に応じて対応してください。
なし	-H uri リポジトリのホスト名、およびポート番号をURI形式で指定	新規追加オプション
なし	-V プログラムのバージョン情報を出力	新規追加オプション
-v 詳細情報の出力	-v 詳細情報の出力	機能は同じです。出力形式が違いますので、“◆-vオプションで出力する詳細情報の差異”で確認してください。

◆-vオプションで出力する詳細情報の差異

LDAPコマンドの-vオプションで出力する詳細情報の差異を示します。

ldapdelete、ldapmodify、ldapsearchコマンド共通

項目	旧LDAPコマンド	新LDAPコマンド
接続先ホスト名、ポート番号	ldap server name: XXXX ldap server port: 99999	ldap_init(XXXX, 99999)
LDAPプロトコルバージョン	ldap protocol version: 3	なし。
文字コード系	character type: SJIS	character type: SJIS -Gオプションによるコード系指定時のみ出力
送信コード系	transmission type: UTF8	なし。
認証機構	mechanism: SIMPLE	なし。
認証方式	authentication method: PASSWORD authentication method: SSL	SSL接続時 ldapsl_init(%s, %d)
リフェラル追跡	follow referrals: ON	なし。
接続先ホスト名、ポート番号のURI形式指定	なし。	ldap_initialize(%s)

ldapdeleteコマンド

項目	旧LDAPコマンド	新LDAPコマンド
削除するエントリのDN	deleting entry cn=User001,ou=User,ou=interstage,o=fujitsu,dc=com	deleting entry "cn=User001,ou=User,ou=interstage,o=fujitsu,dc=com"
削除成功時	entry removed	Delete Result: Success (0)
下位エントリ削除	なし。	deleting children of: ou=Users2,ou=interstage,o=fujitsu,dc=com removing ou=Users2,ou=interstage,o=fujitsu,dc=com

項目	旧LDAPコマンド	新LDAPコマンド
		ou=Users2,ou=interstage,o=fujitsu,dc=com removed

ldapmodifyコマンド

項目	旧LDAPコマンド	新LDAPコマンド
追加/更新するエントリのDN	adding new entry cn=User001,ou=User,ou=interstage,o=fujitsu,dc=com エントリの区切りに空白行が存在します。	adding new entry "cn=User001,ou=User,ou=interstage,o=fujitsu,dc=com" エントリの区切りに空白行は存在しません。
変名 (古いRDNを残す場合)	new RDN: cn=User002 (keep existing values) modifying rdn of entry cn=User001,ou=User,ou=interstage,o=fujitsu,dc=com rename completed	modifying rdn of entry "cn=User001,ou=User,ou=interstage,o=fujitsu,dc=com" new RDN: "cn=User002" (keep existing values) modrdn completed
変名 (古いRDNを削除する場合)	new RDN: cn=User002 (do not keep existing values) modifying rdn of entry cn=User001,ou=User,ou=interstage,o=fujitsu,dc=com rename completed	modifying rdn of entry "cn=User001,ou=User,ou=interstage,o=fujitsu,dc=com" new RDN: "cn=User002" (do not keep existing values) modrdn completed

ldapsearchコマンド

項目	旧LDAPコマンド	新LDAPコマンド
サーチベース	search base: ou=interstage,o=fujitsu,dc=com	なし。
-fオプションでの検索フィルタのファイル指定	filter pattern: read filters file	filter pattern: (objectclass=*)
属性指定	returning: dn	requesting: dn
検索フィルタ指定	[filter: objectclass=*]	filter: objectclass=*
検索条件合致数	X matches	なし。

■メッセージ

Interstage ディレクトリサービスのLDAPクライアントSDKでは、LDAPの最新の規約に正しく準拠するため、メッセージを変更/追加しました。また、InfoDirectory SDKのLDAPコマンドで出力していたメッセージラベルのない形式(“メッセージ集”の“Interstage ディレクトリサービスが出力するメッセージ”を参照)から、メッセージラベルのある形式に変更しました。詳細は、“メッセージ内容の変更について”の[“2.5.17 Interstage ディレクトリサービスのメッセージ”](#)を参照してください。

■LDIFの仕様

- Interstage ディレクトリサービスのLDAPクライアントSDKでは、LDAPの最新の規約に正しく準拠するため、InfoDirectory SDKのLDIFでサポートしていた、Base64形式の外部ファイルはサポートしていません。LDIFに直接Base64形式で指定してください。

- 属性値を外部ファイルから参照する場合、URL形式だけを指定できます。
以下のように指定してください。

Windows

description:< file:///C:¥data¥utf8.txt

Solaris

Linux

description:< file:///data/utf8.txt

4.4 EJBサービス(IJServer)への移行方法

Interstage Application Server V9.0以降ではInterstage Application Server V5.1以前から提供していた以下の機能は提供していません。

- EJB1.0規約に準拠したアプリケーション運用
- 高速呼び出し機能
- Light EJBコンテナ機能
- DBアクセス環境定義

EJB1.0規約に準拠したアプリケーションをInterstage V9.0以降に移行する場合には、EJB1.1規約以降に準拠したアプリケーションに変更し、Interstage Application Server V9.0以降が提供するIJServer上での運用に移行してください。

DBアクセス環境定義を使用していた場合にはInterstage管理コンソールもしくはisj2eeadminコマンドでJDBCデータソースを定義する運用に移行してください。DBアクセス環境定義で定義したJDBCデータソースを使用してCMP1.1 Entity Beanで有効だったSQL文のキャッシュ機能は、Oracle10g以降のJDBCドライバを利用した場合に有効なStatementキャッシュ機能を使用した運用に移行してください。ここでは、高速呼び出し機能とLight EJBコンテナ機能からIJServerに移行する方法について説明します。

4.4.1 EJBの高速呼び出し機能とLight EJBコンテナ機能からIJServerへの移行

IJServerは高速呼び出し機能とLight EJBコンテナ機能に対して下表のような差異があるため、移行の際は留意が必要です。

なお、本バージョンのIJServerに対するセットアップ、および運用操作はInterstage管理コンソールまたはisj2eeadminコマンドを用いて簡単に行うことができます。

isj2eeadminコマンドのできる操作などについては、“リファレンスマニュアル(コマンド編)”の“isj2eeadminコマンド”を参照してください。

以下に、それぞれの運用方法の違いを表で表します。

	高速呼び出し機能	旧バージョンにおけるIJServer (Light EJBコンテナ機能)	本バージョンのIJServer
クライアントの環境設定	以下のファイルをクラスパスに設定します。 Windows [JDK1.3の場合] C:¥Interstage¥EJB¥lib ¥fjcontainer32.jar [JDK1.4の場合] C:¥Interstage¥EJB¥lib ¥fjcontainer34.jar Solaris Linux [JDK1.3の場合] /opt/FJSVejb/lib/fjcontainer32.jar [JDK1.4の場合] /opt/FJSVejb/lib/fjcontainer34.jar	以下のファイルをクラスパスに設定します。 Windows [JDK1.3の場合] C:¥Interstage¥EJB¥lib ¥fjcontainer32.jar [JDK1.4の場合] C:¥Interstage¥EJB¥lib ¥fjcontainer34.jar Solaris Linux [JDK1.3の場合] /opt/FJSVejb/lib/fjcontainer32.jar [JDK1.4の場合] /opt/FJSVejb/lib/fjcontainer34.jar	以下のファイルをクラスパスに設定します。 Windows C:¥Interstage¥EJB¥lib ¥fjcontainer94.jar Solaris Linux /opt/FJSVejb/lib/fjcontainer94.jar (fjcontainer72.jarもしくはfjcontainer74.jarを使用することも可能です。)
配備	以下のツールを使用して配備を実行します。 ・ J2EE Deploymentツール(非サーバ管理モード)	以下のツールを使用してIJServer(またはLight EJBコンテナ)を作成します。	以下のツールを使用してIJServerを作成します。 ・ Interstage管理コンソール

	高速呼出し機能	旧バージョンにおけるIJSERVER (Light EJBコンテナ機能)	本バージョンのIJSERVER
	EJB Deploymentツールで展開処理実行後、以下のコマンドを実行してEJBアプリケーションをインストールします。 ejbinstalleb [サーバ用生成物] 配備実行後にカスタマイズツールを使用して、Java VM外から呼び出されるBeanを1つだけ“高速に呼び出すBean”に定義し、Java VM内で呼び出されるBeanを“高速に呼び出されるBean”に定義します。	- J2EE Deployment ツール (isdeployコマンド) - J2EE管理ツール - ejbmakecontainerコマンド 以下のツールを使用してIJSERVERに対してWebアプリケーション/EJBアプリケーションを配備します。 - J2EE Deployment ツール (isdeployコマンド) - isdeploybコマンド - J2EE管理ツール - EJB Deploymentツールで展開処理実行後、以下のコマンドを実行してEJBアプリケーションをインストールします。 ejbinstalleb -s [IJSERVER名] [サーバ用生成物] IJSERVERに配備されたEJBアプリケーションは、同一Java VM上で動作します。ServletとEJBは別Java VMで動作します。	- isj2eeadminコマンド IJSERVERには以下の4種類のIJSERVERを選択できます。 - WebアプリケーションとEJBアプリケーションを同一Java VMで運用 - WebアプリケーションとEJBアプリケーションを別Java VMで運用 - Webアプリケーションのみ運用 - EJBアプリケーションのみ運用 以下のツールを使用してIJSERVERに対してWebアプリケーション/EJBアプリケーションを配備します。同一IJSERVERに配備されたEJBアプリケーションは同一Java VM上で動作します。 - Interstage管理コンソール - ijsdeploymentコマンド (配備解除時はijsundeploymentコマンド)
カスタマイズ	以下のツールを使用してEJBアプリケーションをカスタマイズします。 - ejbcustx - ejbdefimport コマンドと ejbdefexportコマンド	以下のツールを使用してEJBアプリケーションをカスタマイズします。 - ejbcustx -s [IJSERVER名] - ejbdefimport コマンドと ejbdefexportコマンド(-sオプションでIJSERVER名を指定します。) または以下のツールを起動し、配備済みのEJBアプリケーションを選択してカスタマイズします。 - J2EE Deployment ツール (isdeployコマンド)	以下のツールを使用して配備済みのEJBアプリケーションを選択してカスタマイズします。 - Interstage管理コンソール - ejbdefimport コマンドと ejbdefexportコマンド(-iオプションでIJSERVER名を指定します。)
運用	以下のツールを使用して高速に呼び出されるBeanを登録したワークユニットを定義します。 - isaddwudefコマンド(削除時は isdelwudefコマンド) - Interstage運用操作ツール 定義したワークユニットは以下のツールを使用して起動します。 - isstartwuコマンド(停止時は isstopwuコマンド) - Interstage運用操作ツール	IJSERVERを定義するとワークユニットも自動的に定義されます。以下のツールを使用してIJSERVER(またはワークユニット)を起動します。 - J2EE管理ツール - isstartwuコマンド(停止時は isstopwuコマンド) - Interstage運用操作ツール	Interstage V6以降ではIJSERVERは以下のように位置付けています。「J2EEアプリケーションを運用するワークユニットをIJSERVERと呼びます。(IJSERVERとワークユニットは1対1の関係を持ちます。) IJSERVERはJ2EEアプリケーションの配備対象であり、起動/停止の単位です。」 以下のツールを使用してIJSERVERを起動します。 Interstage管理コンソール

	高速呼出し機能	旧バージョンにおけるIJServer (Light EJBコンテナ機能)	本バージョンのIJServer
			isstartwuコマンド(停止時はisstopwuコマンド)
リソース定義	以下のツールを使用してJ2EEの各種リソースを定義します。 - J2EE管理ツール - J2EEリソースアクセス定義 - J2EE Deploymentツール(isdeployコマンド)		以下のツールを使用してJ2EEの各種リソースを定義します。 - Interstage管理コンソール - isj2eeadminコマンド

4.4.2 Interstage Application Server V3.xからの移行

■Interstage V3.xと本バージョン・レベルでの動作の違い

EJBアプリケーション内で以下の機能を使用した場合、Interstage V3.xで動作させた場合と本バージョン・レベルで動作させた場合と処理結果が異なります。

機能	本バージョン・レベルでの動作	Interstage V3.xの動作
リエントラント制御	[Session Beanの場合] 常に1つのスレッドだけがSessionオブジェクトに対して処理を実行できる。1つのSessionオブジェクトに対して複数のクライアントから同時にアクセスしたり、ループバック呼出しを行うと、後から受信した要求に対してjava.rmi.RemoteExceptionがthrowされる。	[Session Beanの場合] 1つのSessionオブジェクトに対して複数のクライアントから同時にアクセスしたり、ループバック呼出しを行ってもエラーとならない。
	[Entity Beanの場合] deployment descriptorのリエントラント種別に非リエントラントが指定されている場合、1つのEntityオブジェクトに対して複数のクライアントから同時にアクセスしたり、ループバック呼出しを行うと、後から受信した要求に対してjava.rmi.RemoteExceptionがthrowされる。 deployment descriptorのリエントラント種別にリエントラントが指定されている場合、例外は発生しない。	[Entity Beanの場合] deployment descriptorのリエントラント種別にかかわらず例外は発生しない。
Session BeanのPrimaryKey関連メソッド	Session BeanのEJBObject.getPrimaryKey()、EJBHome.remove(Object primaryKey)を実行すると以下の例外がthrowされる。 - EJBObject.getPrimaryKey()の場合 java.rmi.RemoteException - EJBHome.remove(Object primaryKey)の場合 javax.ejb.RemoveException	Session BeanのEJBObject.getPrimaryKey()、EJBHome.remove(Object primaryKey)を実行しても正常に動作する。
Session Bean内で発生したシステム例外処理	Sessionオブジェクトは破棄される。	Sessionオブジェクトは破棄されない。
Entity Bean内で発生したシステム例外処理	Entity Beanのインスタンスは破棄される。	Entity Beanのインスタンスは破棄されない。

機能	本バージョン・レベルでの動作	Interstage V3.xの動作
setRollbackOnly())、 getRollbackOnly() メソッド	トランザクション管理種別が“Bean”の場合でインスタンスがトランザクションに関連付けられていないときにEntityContextインタフェース、またはSessionContextインタフェースのsetRollbackOnly()やgetRollbackOnly()を実行するとIllegalStateExceptionが発生する。トランザクション管理種別が“Container”でトランザクション属性が“NotSupported”の場合も同様にIllegalStateExceptionがthrowされる。	EntityContextインタフェースまたはSessionContextインタフェースのsetRollbackOnly()やgetRollbackOnly()は、どのトランザクション属性でも正常に動作する。
コンテナのリソース 接続者管理	リソース接続者に“Container”が指定された場合、Interstage管理コンソールまたはisj2eeadminコマンドで定義したデータソースのユーザIDとパスワードがリソース接続時に使用される。 リソース接続者に“Application”が指定された場合、アプリケーションにユーザID、パスワードが指定された場合はそのユーザID、パスワードがリソース接続時に使用される。アプリケーションに指定されなかった場合はクライアントの認証情報が使用される。	リソース接続者に指定された値にかかわらず、アプリケーションにユーザID、パスワードが指定された場合はそのユーザID、パスワードがリソース接続時に使用される。アプリケーションにユーザID、パスワードが指定されていない場合はDBアクセス環境定義に設定されているユーザIDとパスワードがリソース接続時に使用される。
STATEFUL Session Beanの create()、 remove()メソッド	[create()の場合] トランザクション内で実行されない。トランザクション内で呼び出した場合は、トランザクションが中断されてから実行される。	[create()の場合] 呼出し元で開始されたトランザクション上で動作する。 システム例外が発生した場合はトランザクションにロールバックをマークしてTransactionRolledBackExceptionがthrowされる。
	[remove()の場合] トランザクション内で実行された場合、javax.ejb.RemoveExceptionがthrowされる。	[remove()の場合] 呼出し元で開始されたトランザクション上で動作する。 システム例外が発生した場合はトランザクションにロールバックをマークしTransactionRolledBackExceptionがthrowされる。
Session Beanの Mandatoryトランザクション属性	トランザクション属性にMandatoryが指定された場合、トランザクションが開始されていない状態でSession Beanが呼び出されると、TransactionRequiredExceptionがthrowされる。	トランザクション属性にMandatoryが指定された場合、トランザクションが開始されていない状態でSession Beanが呼び出されても例外は発生せず、正常に動作する。
UserTransaction のlookup	“J2EE ユーザーズガイド”の“Enterprise Beanクラスの作成”の“Enterprise Beanクラスのメソッドが実行可能な操作”に記述されているUserTransactionをlookupできないメソッドでUserTransactionをlookupした場合、java.lang.IllegalStateExceptionが発生する。	すべてのメソッドでUserTransactionをlookupすることが可能。
remove(Primary Key)	[Entity Beanの場合] Home.remove(primarykey)メソッドで、primarykeyに該当するレコード(インスタンス)が存在しない場合、java.rmi.NoSuchObjectExceptionをthrowする。	[Entity Beanの場合] Home.remove(primarykey)メソッドで、primarykeyに該当するレコード(インスタンス)が存在しない場合、javax.ejb.RemoveExceptionをthrowする。

機能	本バージョン・レベルでの動作	Interstage V3.xの動作
Entity Bean Homeインタフェースのremove(primarykey)メソッド	remove(primarykey)メソッドで、primarykeyに該当するレコード(インスタンス)が存在しない場合、RemoteException(NoSuchObjectExcetion)がthrowされる。	remove(primarykey)メソッドで、primarykeyに該当するレコード(インスタンス)が存在しない場合、RemoveException がthrowされる。

◆処理モードの変更方法

Interstage管理コンソールのワークユニット設定に以下の定義をしてください。

設定項目

Java VM オプション(Java Command Option)

設定値

-DFJV3MODE=処理モード

処理モード

yes:Interstage V3の動作

no :本バージョンの動作



注意

以下の場合には本バージョンの動作をします。

- 処理モードを設定しない場合
- 処理モードに上記に示す設定値以外を設定した場合

■対処方法

“[■Interstage V3.xと本バージョン・レベルでの動作の違い](#)”に記載されている機能を使用している場合、“[■Interstage V3.xと本バージョン・レベルでの動作の違い](#)”に注意して、EJBアプリケーションを修正してください。

4.4.3 ロードバランス機能について

Interstage Application Server V5.x(Interstage V5.x)よりTraffic Director、およびInterstage Application Server 8.0.x(Interstage 8.0.x)よりIPCOMがサポートされたため、EJBサービスでロードバランシングする場合に従来使用されていたロードバランス機能が、非推奨となりました。

第5章 Interstage運用操作、環境の移行

ここでは、以下について説明します。

- [Interstage操作の移行](#)
Interstage Application Server 8.0以前から移行する場合に参照してください。
- [Interstage統合コマンドの移行](#)
Interstage Application Server 8.0以前から移行する場合に参照してください。
- [Interstage管理コンソールの変更内容](#)
Interstage Application Server V7.0以前から移行する場合に参照してください。
- [Interstage JMXサービスの移行](#)
Interstage Application Server 8.0以前から移行する場合に参照してください。
- [マルチサーバ管理環境の移行](#)
Interstage Application Server V7.0以降から移行する場合に参照してください。
- [Interstage HTTP Serverの移行](#)
Interstage Application Server 8.0以前から移行する場合に参照してください。
- [ワークユニットの移行](#)
Interstage Application Server V6.0以前から移行する場合に参照してください。
- [JDK/JRE、Javaツールの変更内容](#)
Interstage Application Server V7.0以前から移行する場合に参照してください。
- [Interstage証明書環境の移行](#)
Interstage Application Server V7.0以前から移行する場合に参照してください。
- [Interstage ディレクトリサービスの移行](#)
Interstage Application Server V9.1以前から移行する場合に参照してください。
- [Interstage シングル・サインオンの移行](#)
Interstage Application Server V9.0以前から移行する場合に参照してください。

5.1 Interstage操作の移行

ここでは、Interstage操作について、以下を説明します。

- [Interstage Application Server V9.0での変更内容](#)
- [Interstage Application Server 8.0での変更内容](#)
- [Interstage Application Server V7.0での変更内容](#)

■Interstage Application Server V9.0での変更内容

Interstage資源保護のためのセキュリティ強化  

Interstage資源保護のために、Interstage Application Server 8.0までは製品インストール後にCORBAサービスやコンポーネントトランザクションサービスなどの各機能でセキュリティ強化の作業をしました。

Interstage Application Server V9.0以降は、セキュリティの強化状態をデフォルトとしてインストールを行います。したがって旧版で実施していたインストール後のセキュリティ強化の作業は不要になります。また、セキュリティ強化しない指定でインストールした場合も、後でissetsecuritymodeコマンドによりセキュリティ強化することが可能です。逆に、issetsecuritymodeコマンドでセキュリティ強化の設定を、8.0までの標準的な設定にすることも可能です。

Interstage資源保護のためのセキュリティ強化の詳細は、以下を参照してください。

- “セキュリティシステム運用ガイド”の“共通の対策”
- “リファレンスマニュアル(コマンド編)”の“issetsecuritymode”

V9.0以降でissetsecuritymodeコマンドを提供することにより、以下のコマンドは未サポートになります。

- tdsecmode
- ejbchangemode
- jmschmod
- guisecmode
- isjmxchangemode
- ijschgauthority

■Interstage Application Server 8.0での変更内容

サーバ起動時のInterstage起動について Solaris Linux

Interstage Application Server Web-J Editionをインストールすると、システム初期化スクリプトとして以下のスクリプトが登録されます。本スクリプトにより、サーバの起動時にInterstageが自動的に起動されます。

- S99startisスクリプト Solaris
- startisスクリプト Linux

上記のスクリプトを実行すると、isstartコマンドの実行時に起動するサービスが起動されます。上記のスクリプトの詳細については、“リファレンスマニュアル(コマンド編)”を参照してください。

上記のスクリプトが運用に影響を与える場合には、削除や退避などの対処を行ってください。特に、クラスタ環境を構築し、Interstageを運用している場合には、上記のスクリプトを退避するなどの対処が必要です。詳細は、“高信頼性システム運用ガイド”を参照してください。また、その他、上記のスクリプトが実行されることにより、運用に影響を与える場合には、同様の対処を行ってください。

サーバ停止時のInterstage停止について Solaris Linux

Interstage Application Server Web-J Editionをインストールすると、システムのシャットダウンスクリプトとして以下のスクリプトが登録されます。本スクリプトにより、サーバの停止時に、Interstageが自動的に停止されます。

- K00stopisスクリプト

上記のスクリプトを実行すると、isstop -fコマンドの実行時に停止されるサービスが停止されます。上記のスクリプトの詳細については、“リファレンスマニュアル(コマンド編)”を参照してください。

上記のスクリプトが実行されることにより、運用に影響を与える場合には、削除や退避などの対処を行ってください。

■Interstage Application Server V7.0での変更内容

サーバ起動時のInterstage起動について Solaris Linux

Interstage Application Server Enterprise Edition、Standard-J Editionをインストールすると、システム初期化スクリプトとして以下のスクリプトが登録されます。本スクリプトにより、サーバの起動時にInterstageが自動的に起動されます。

- S99startisスクリプト Solaris

- startisスクリプト **Linux**

上記のスクリプトを実行すると、isstartコマンドの実行時に起動するサービスが起動されます。上記のスクリプトの詳細については、“リファレンスマニュアル(コマンド編)”を参照してください。

上記のスクリプトが運用に影響を与える場合には、削除や退避などの対処を行ってください。特に、クラスタ環境を構築し、Interstageを運用している場合には、上記のスクリプトを退避するなどの対処が必要です。詳細は、“高信頼性システム運用ガイド”を参照してください。また、その他、上記のスクリプトが実行されることにより、運用に影響を与える場合には、同様の対処を行ってください。

サーバ停止時のInterstage停止について **Solaris** **Linux**

Interstage Application Server Enterprise Edition、Standard-J Editionをインストールすると、システムのシャットダウンスクリプトとして以下のスクリプトが登録されます。本スクリプトにより、サーバの停止時に、Interstageが自動的に停止されます。

- K00stopisスクリプト

上記のスクリプトを実行すると、isstop -fコマンドの実行時に停止されるサービスが停止されます。上記のスクリプトの詳細については、“リファレンスマニュアル(コマンド編)”を参照してください。

上記のスクリプトが実行されることにより、運用に影響を与える場合には、削除や退避などの対処を行ってください。

5.2 Interstage統合コマンドの移行

ここでは、Interstage統合コマンドの移行について説明します。

■isstatコマンドの表示形式について

Interstage Application Server V9.0以降では、isstatコマンドで表示されるFJapache (Interstage HTTP Server)の起動状態の表示形式が変更になります。isstatコマンドの表示形式については、“リファレンスマニュアル(コマンド編)”を参照してください。

5.3 Interstage管理コンソールの変更内容

ここでは、Interstage管理コンソールについて、以下を説明します。

- [Interstage管理コンソールの初期値変更について](#)
- [Interstage Application Server V9.2での変更内容](#)
- [Interstage Application Server V9.0での変更内容](#)
- [Interstage Application Server V7.0での変更内容](#)

■Interstage管理コンソールの初期値変更について

Interstage管理コンソールの初期値が以下のように変更されました。以前と同等の環境を構築したい場合は、注意してください。

Interstage Application Server V7.0での変更

- CORBAアプリケーションの動作モードの初期値変更

変更箇所:

[システム] > [ワークユニット] > [ワークユニット名] > [配備] > [動作モード]

Interstage V6.0以前	COMPATIBLE
Interstage V7.0以降	SYNC_END

変更理由:

Linuxで初期スレッドが終了するとプロセスがdefunctとなり、トラブル発生時に資料採取が行えず調査が困難となるために変更しました。

- **Java VMオプションの初期値変更**

変更箇所:

[ワークユニット] > [新規作成]タブ > [詳細設定] > [ワークユニット設定] > [JavaVMオプション]

Interstage V6.0以前	なし (指定しない場合、-Xms2m -Xmx64mとなります。)
Interstage V7.0以降	Windows Server(R) for Itanium-based Systems／Linux for Itaniumの場合:-Xms16m -Xmx512m 上記以外のオペレーティングシステムの場合:-Xms16m -Xmx256m

変更理由:

ヒープ領域不足によるJava VMのダウンを防ぐために変更しました。

- **Servletコンテナのタイムアウトの初期値変更**

変更箇所:

[Webサーバ] > [Webサーバコネクタ] > [新規作成]タブ > [Servletコンテナ設定] > [タイムアウト]、および
[ワークユニット] > [新規作成]タブ > [詳細設定] > [Servletコンテナ設定] > [タイムアウト]

Interstage V6.0以前	300秒
Interstage V7.0以降	480秒

変更理由:

CORBAサービスのデフォルトのクライアントタイムアウト時間よりも大きくするために変更しました。

- **IJServerワークユニット起動待ち時間の初期値変更**

変更箇所:

[ワークユニット] > [新規作成]タブ > [詳細設定] > [ワークユニット設定] > [ワークユニット起動待ち時間]

Interstage V6.0以前	180秒
Interstage V7.0以降	600秒

変更理由:

アプリケーションによって、ワークユニット起動時に時間がかかることがあるために変更しました。

- **アプリケーション最大処理時間の初期値変更**

変更箇所:

[ワークユニット] > [新規作成]タブ > [詳細設定] > [ワークユニット設定] > [アプリケーション最大処理時間]の初期値

Interstage V6.0以前	処理時間の監視は行いません
Interstage V7.0以降	480秒

変更理由:

アプリケーション最大処理時間を超過した場合には、原因調査用にスレッドダンプが採取されます。アプリケーションの障害などにより処理遅延が発生した場合に、原因調査が行えるように変更しました。

■Interstage Application Server V9.2での変更内容

Interstage Application Server V9.2での変更内容を説明します。

[Interstage管理コンソールへのログイン時のオートコンプリート機能について](#)

Interstage 管理コンソールでは、セキュリティ対策としてなりすましによる操作を防止するため、ログイン時のWebブラウザのオートコンプリート機能が無効となりました。Interstage Application Server V9.1以前のように、Webブラウザのオートコンプリート機能を使用してログインを行うことができなくなりますので、注意してください。

■Interstage Application Server V9.0での変更内容

Interstage Application Server V9.0での変更内容を説明します。

Interstage管理コンソールのセキュリティ運用形態について

本製品のインストール時に“運用形態の選択”画面で指定するInterstage管理コンソールのセキュリティ運用形態の初期設定値が以下のように変更となります。

製品バージョン	セキュリティ運用形態の初期設定値	Interstage管理コンソールのURL
Interstage 8.0以前	“SSL暗号化通信を使用しない。”	http://ホスト名:ポート番号/IsAdmin/
Interstage V9.0以降	“SSL暗号化通信を使用する。”	https://ホスト名:ポート番号/IsAdmin/

SSLプロトコルバージョン、暗号アルゴリズムの初期設定の変更

Interstage 管理コンソールとWebブラウザ間のSSL暗号化通信を安全に運用するため、SSLプロトコルバージョンおよび暗号アルゴリズムの初期設定から、弱いSSLプロトコルバージョンおよび弱い暗号アルゴリズムをはずしました。違いについては、以下の表を参考にしてください。

製品バージョン	Webブラウザから接続可能なSSLプロトコルバージョン	Webブラウザから接続可能な暗号アルゴリズム
Interstage 8.0以前	SSL2.0	168bitのトリプルDES暗号,MD5 MAC 128bitのRC4暗号,MD5 MAC 128bitのRC2暗号,MD5 MAC 56bitのDES暗号,MD5 MAC 40bitのRC4暗号,MD5 MAC 40bitのRC2暗号,MD5 MAC
	SSL3.0 TLS1.0	256bitのAES暗号,SHA-1 MAC 128bitのAES暗号,SHA-1 MAC 168bitのトリプルDES暗号,SHA-1 MAC 128bitのRC4暗号,SHA-1 MAC 128bitのRC4暗号,MD5 MAC 56bitのDES暗号,SHA-1 MAC 40bitのRC4暗号,MD5 MAC 40bitのRC2暗号,MD5 MAC
Interstage V9.0以降	SSL3.0 TLS1.0	256bitのAES暗号,SHA-1 MAC 128bitのAES暗号,SHA-1 MAC 168bitのトリプルDES暗号,SHA-1 MAC 128bitのRC4暗号,SHA-1 MAC 128bitのRC4暗号,MD5 MAC

SSLプロトコルバージョン、暗号アルゴリズムの安全性については、“セキュリティシステム運用ガイド”の“セキュリティ動向”を参照してください。



注意

Service Packを適用していないMicrosoft(R) Internet Explorer 5.01を利用すると、弱い暗号しかサポートされていないため、Interstage 管理コンソールに接続できません。強い暗号を利用できるようにするためには、Webブラウザをバージョンアップするか、または、Service Pack 1以降を適用することを推奨します。Service Packは、マイクロソフト株式会社のサイトで入手し適用してください。

なお、SSL暗号化通信の設定を弱く変更することで、Service Packを適用していないMicrosoft(R) Internet Explorer 5.01からも接続できるようになります。ただし、IDやパスワードなどの通信データが解読される危険性が高くなりますので、注意してください。設定の変更方法については、“運用ガイド(基本編)”の“Interstage管理コンソール環境のカスタマイズ”を参照してください。

■Interstage Application Server V7.0での変更内容

Interstage Application Server V7.0で、以下のように仕様が変更されました。

変更点	Interstage V6.0	Interstage V7.0
画面共通のオペレーションボタン、表示情報の配置位置の変更	- ログアウトボタンは画面左下に配置されています。 - ヘルプボタンは画面右上のタイトル部の下に配置されています。 - ユーザ名は画面左下に配置されています。 - 接続先は画面左下に配置されています。	左記の情報はすべて画面内タイトル部右(画面右上)に配置されます。
Interstage管理コンソールの[環境設定]の設定項目の有効範囲を変更	「自動更新の有無」は全ユーザに対して有効です。 「自動更新間隔」は全ユーザに対して有効です。 「項目の説明表示」は設定ユーザに対して有効です。	左記の全項目はユーザ単位で有効となります。
Interstage管理コンソールの[システム]>[ワークユニット]>[IJServerワークユニット名を選択]>[ログ参照]タブで表示される操作ボタンの変更	- 操作ボタンはログ内容の表示域上部に表示されています。 - 操作できないボタンは非表示になっています。	- 操作ボタンはログ内容の表示域下部に表示されます。 - 操作できないボタンはグレーアウトで表示されます。
メッセージ表示域に表示されるメッセージを削除するボタンを追加	メッセージ表示域に表示されたメッセージは削除できませんでした。	メッセージ表示域に表示されたメッセージを削除するボタンをメッセージ領域の左に配置しました。
メッセージ表示域に表示されるメッセージからメッセージ集をリンク	メッセージ表示域に表示されるのは単なるメッセージ文字列でした。	メッセージ表示域に表示されるメッセージはメッセージ集にリンクしています。
ツリーに表示されるノード構成を変更	- Interstage管理コンソールの子ノードとしてシステムノードが表示されていました。 - サービスノードの子ノードにはJDBC、JMS、イベントサービス、Connector、JavaMail、Webサーバが表示されていました。	- Interstage管理コンソールの子ノードとしてInterstage Application Serverノードが追加され、その子ノードとしてシステムノードが表示されます。 - サービスノードの子ノードにはイベントサービス、Webサーバが表示されます。 - Interstage管理コンソールの子ノードとしてリソースノードが追加されその子ノードにはJDBC、JMS、Connector、JavaMailが表示されます。
Interstage管理コンソールと連携する他製品のノードをツリー上に表示する機能を追加	他製品連携機能なし	Interstage管理コンソールと連携する他製品の機能ノードがInterstage

変更点	Interstage V6.0	Interstage V7.0
		管理コンソールのツリーに表示されます。
Interstageのインストールに必要なディスク容量見積りにJSPのコンパイル資材、およびメッセージ集リンク用資材の容量の考慮を追加	容量見積りに考慮なし	インストールに必要なディスク容量としてJSPのコンパイル資材、およびメッセージ集リンク用の資材を含めた容量が表示されます。

Interstage管理コンソールのログインについて

Interstage管理コンソールにログインする場合にLANが切断されていると、ログイン時にis40003のエラーが発生します。また、Interstage管理コンソールにログイン後に、Interstage Operation Toolを再起動、またはLANを切断すると、Interstage管理コンソールのその後の処理でis40003のエラーが発生する場合があります。再度ログイン画面を表示して、ログインし直してください。

5.4 Interstage JMXサービスの移行

■Interstage JMXサービスの初期設定の変更

Interstage Application Server/Interstage Web Server V9.0以降のInterstage JMXサービスでは、サービス起動時の初期設定を以下のように変更しています。Interstage Application Server 8.0以前のInterstage JMXサービスと同じ環境を構築したい場合は注意してください。

・ Perm領域の最大サイズの変更

変更箇所:

isjmx.xmlファイルのserverタグのoptions属性の初期設定値

Interstage 8.0以前	-Xmx256m(Perm領域の最大サイズの指定はなし、指定なしの場合のデフォルトは64MB)
Interstage V9.0以降	-Xmx256m -XX:MaxPermSize=128m(Perm領域の最大サイズを128MBに指定)

変更理由:

Perm領域の不足によるInterstage JMXサービスのメモリ不足発生を防ぐために変更しました。

■RMI通信で要求を受け付けるポート番号を固定ポートに変更

Interstage Application Server/Interstage Web Server V9.0以降のInterstage JMXサービスでは、RMI通信で要求を受け付けるポート番号を固定ポートで設定するように変更しました。デフォルトでは12230のポート番号を使用します。

ポート番号のカスタマイズを行う場合は、“運用ガイド(基本編)”の“Interstage管理コンソール環境のカスタマイズ”を参照してください。

■バックアップ・リストア

Interstage Application Server V7.0(Interstage V7.0)以前のInterstage JMXサービスでは、Windows(R)でInterstage JMXサービスの資源をバックアップまたはリストアする場合に、isguibackupコマンド、isguirestoreコマンドを使用していました。

Interstage Application Server 8.0以降は、Solaris/Linuxと同様にisjmxbackupコマンド、isjmxrestoreコマンドを使用します。

Interstage V7.0以前のInterstage JMXサービス資源をバックアップ・リストアする場合は、isguibackupコマンドでバックアップした資源を指定して、isjmxrestoreコマンドでリストアを行ってください。

■Interstage JMXサービスのタイムアウト値の変更

Interstage V7.0では、Interstage JMXサービスのタイムアウト値が、60分から20分に変更されています。

5.5 マルチサーバ管理環境の移行

ここでは、マルチサーバ管理機能について、以下を説明します。

- [Interstage Application Server V9.0での変更内容](#)
- [Interstage Application Server 8.0での変更内容](#)
- [旧バージョンからの移行手順](#)

■Interstage Application Server V9.0での変更内容

Interstage Application Server V9.0(Interstage V9.0)での変更内容を説明します。

[デフォルトのprotocolsの変更](#)

管理サーバと管理対象サーバ間のデフォルトの通信プロトコルが、RMIからHTTPSに変更されました。管理サーバと管理対象サーバ間の通信プロトコルの詳細については、“運用ガイド(基本編)”の“管理サーバから管理対象サーバの操作”を参照してください。

■Interstage Application Server 8.0での変更内容

Interstage Application Server 8.0(Interstage 8.0)での変更内容を説明します。

[管理対象サーバでの機能の追加と削除の禁止](#)

Interstage 8.0以降では、管理対象サーバにおいて、現在インストールされている機能の追加と削除（インストール機能の選択状態の変更）を行うことができなくなりました。管理対象サーバに対してインストール機能の選択状態を変更する場合は、サイトから削除して行ってください。

[isaddadminfuncコマンドの動作の変更](#)

Interstage 8.0以降のスタンドアロンサーバでisaddadminfuncコマンドを実行すると、以下のメッセージが出力されて、“y”／“n”の入力が求められるように変更されました。

```
Switching from the Admin Server to Standalone Server is not possible...
Are you sure to switch from the Standalone Server to the Admin Server? [y,n]:
```

[サポートするエディションの変更](#)

Interstage 8.0以降では、管理サーバ機能および管理対象サーバ機能が使用可能となるエディションが以下のように変更されました。

機能	製品バージョン	サポートエディション
管理サーバ機能	—	• Interstage Application Server Enterprise Edition
管理対象サーバ機能	Interstage 8.0以降	• Interstage Application Server Enterprise Edition
	Interstage V7.0	• Interstage Application Server Enterprise Edition • Interstage Application Server Standard Edition • Interstage Application Server Plus • Interstage Application Server Web-J Edition

[サーバグループへ異なるJDK/JRE構成のサーバの追加禁止](#)

従来は、サーバグループへのサーバ追加時に、追加するサーバにインストールされているJDK/JREのバージョンがサーバグループ内のサーバと異なっても追加することができました。しかし、Interstage 8.0以降では、サーバグループへのサーバ追加時にエラーが発生して追加に失敗します。追加時にエラーが発生した場合は、エラーメッセージからエラーの原因を判断し、上記についてサーバグループと同一のサーバを選択して、再度実行してください。

■旧バージョンからの移行手順

旧バージョンで構築したマルチサーバ管理環境を本バージョンに移行する手順を説明します。



注意

- Interstage V7.0以前のWebサーバ(InfoProvider Pro)を使用している場合は、Interstage 8.0以降では、Interstage HTTP Serverに移行する必要があります。移行方法の詳細については、“[4.1.2 V7.0以前のInfoProvider Proからの移行](#)”を参照してください。
- Interstage V6.0以前のServletサービス、またはEJBサービスを使用している場合は、Interstage 8.0以降では、IJSERVERを再作成して配備する必要があります。詳細については、“[6.2 J2EEアプリケーションの移行](#)”、“[6.4 Servletサービスの移行](#)”、および“[6.5 EJBサービスの移行](#)”を参照してください。

管理サーバのみの移行手順

バックアップ、およびリストアにより移行します。リストアの作業については“運用ガイド(基本編)”の“管理サーバのバックアップ/リストア”を参照してください。



ポイント

管理サーバをInterstage 8.0以降に移行した場合でも、管理対象サーバは移行前のバージョンでの運用が可能です。

管理サーバおよび管理対象サーバの移行手順

IJSERVERを使用したサイト環境にある、旧バージョンの管理対象サーバ(または共存サーバ)を、本バージョンの環境に移行するためには、以下の手順で操作を行う必要があります。

1. 管理対象サーバ資源のバックアップ
2. Interstage Application Serverのバージョンアップ
3. 管理対象サーバ資源のリストア
4. サイト環境の再構築
5. IJSERVERの結合
6. Webサーバコネクタの分離

リストア、およびサイトの構築については“運用ガイド(基本編)”の“管理対象サーバのバックアップ/リストア”および“サーバグループ/単体運用の管理対象サーバを運用する環境の構築”を参照してください。



注意

管理対象サーバは、管理サーバより上位のバージョンの管理対象サーバは管理できません。管理対象サーバを本バージョンに移行する場合は、管理サーバも合わせて移行する必要があります。

1. 管理対象サーバ資源のバックアップ

管理対象サーバのInterstage Application Server資源のバックアップを行ってください。
サイトに所属しているが、本バージョンへの移行を行わない管理対象サーバについては、操作を行う必要はありません。



- 管理対象サーバが多階層IJSerの配置先の場合は、それぞれの管理対象サーバがどの多階層IJSerの配置先であったかを記録しておいてください。
- 管理対象サーバがサーバグループに所属している場合は、それぞれの管理対象サーバのサーバグループ内通番を記録しておいてください。

2. Interstage Application Serverのバージョンアップ

管理サーバ／管理対象サーバの両方について、旧バージョンのInterstage Application Serverをアンインストールし、本バージョンのInterstage Application Serverをインストールしてください。
サイトに所属していたが、本バージョンへの移行を行わない管理対象サーバについては、操作を行う必要はありません。

3. 管理対象サーバ資源のリストア

管理対象サーバのみについて、旧バージョンのバックアップ資源を、本バージョンのスタンドアロンサーバにリストアしてください。
サイトに所属していたが、本バージョンへの移行を行わない管理対象サーバについては、操作を行う必要はありません。

4. サイト環境の再構築

管理対象サーバをサイトに追加してください。また、サーバグループを作成し、管理対象サーバを追加してください。手順の詳細は、“運用ガイド(基本編)”の“サーバグループ/単体運用の管理対象サーバを運用する環境の構築”を参照してください。



- 移行前の環境で多階層IJSerを構成していたサーバで、WebサーバとIJSerが同一のサーバで運用されていなかった場合は、サーバグループへのサーバ追加に失敗します。その場合はスタンドアロンサーバの状態、Webサーバコネクタを操作してください。手順の詳細は“運用ガイド(基本編)”の“サーバグループ／単体運用の管理対象サーバを運用する環境の構築”の“IJSerを使用する場合の注意事項”を参照してください。
- 移行前の環境で多階層IJSerを構成していたサーバで、ServletコンテナとEJBコンテナが同一のサーバで運用されていなかった場合は、EJBコンテナのサーバグループから先にサーバを追加してください。詳細は“運用ガイド(基本編)”の“サーバグループに所属する管理対象サーバについての前提条件と注意事項”の“Interstage”および“IJSer”を参照してください。
- サーバグループ内のサーバグループ通番は、サーバグループにサーバが追加された順番で付与されます。しかし、サーバグループに複数の管理対象サーバを一度に追加すると、サーバグループ内通番は自動的に名前のアルファベット順に決定されます。
移行前の環境でサーバグループ間のリレーションがラインだった場合、移行前と同じリレーションを復元するためには、復旧前と同じサーバグループ内通番とする必要があります。そのためには、サーバグループにサーバを1台ずつ追加し、復旧前と同じサーバグループ内通番にするようにしてください。

5. IJSerの結合

上記までの手順では、多階層IJSerの復元は行えません。上記までの手順が完了した段階では、サイト内のIJSerはすべて単階層のIJSerとなっています。移行前の環境で多階層IJSerを構成していたサーバで、ServletコンテナとEJBコンテナが同一のサーバで運用されていなかった場合は、移行後の環境でIJSerの結合を行い、多階層IJSerを復元してください。手順の詳細については“運用ガイド(基本編)”の“IJSerの結合”を参照してください。

6. Webサーバコネクタの分離

移行前の環境で多階層IJSerを構成していたサーバで、ServletコンテナとWebサーバコネクタが同一のサーバで運用されていなかった場合は、移行後の環境でWebサーバコネクタの移動を行い、多階層IJSerを復元してください。手順の詳細については“運用ガイド(基本編)”の“Webサーバコネクタの分離”を参照してください。

5.6 Interstage HTTP Serverの移行

Interstage Application Server V9.0以降、Interstage HTTP Serverは、Apache HTTP Server 1.3ベースからApache HTTP Server 2.0ベースのWebサーバへ変更されました。V8.0以前のInterstage HTTP Server (Apache HTTP Server 1.3ベース)からInterstage HTTP Server

(Apache HTTP Server 2.0ベース)へ移行する場合は、“[4.1.1 8.0以前のInterstage HTTP Server \(Apache HTTP Server 1.3ベース\)からの移行](#)”を参照して移行してください。

5.7 ワークユニットの移行

ここでは、以前のバージョン・レベルから改善されたワークユニットの機能、および移行時の注意事項について説明します。

■ワークユニットのカレントディレクトリの世代管理機能強化

ワークユニットのカレントディレクトリの世代管理機能が強化され、過去の起動時のカレントディレクトリを0～5世代の間で任意の世代数だけ残すことができます。

Interstage Application Server V7.0以降

すべてのワークユニット種別で、デフォルトのカレントディレクトリのバックアップが1世代残されます。また、ワークユニット定義を変更することにより、0～5世代の間で任意にバックアップ世代数を変更できます。

なお、以前のバージョン・レベルのバックアップ資源をリストアした場合は、以前のバージョン・レベルでの動作が引き継がれます。そのため、カレントディレクトリのバックアップ世代数を変更する場合は、ワークユニット定義を変更し、再登録してください。

バックアップ世代数は、以下で変更できます。

- IJServerワークユニットおよびCORBAワークユニットの場合
Interstage管理コンソールの、[システム] > [ワークユニット] > [ワークユニット名] > [環境設定]画面で、[ワークユニット設定]の[退避するカレントディレクトリの世代数]を変更してください。
- その他のワークユニット種別の場合
ワークユニット定義の[Control Option]セクションの“Number of Revision Directories”ステートメントで変更してください。
なお、CORBAワークユニットの場合も、ワークユニット定義で変更することが可能です。

Interstage Application Server V6.0

IJServerワークユニットの場合は、無条件に5世代のカレントディレクトリのバックアップが残ります。また、CORBAワークユニットの場合は、デフォルトではカレントディレクトリの世代管理は行われず、ワークユニット定義の[Control Option]セクションの“Environment Variable”ステートメントに、環境変数“EXTP_CURRENTDIR_HISTORY=YES”が設定された場合に、5世代のカレントディレクトリがバックアップされます。

その他のワークユニット種別では、カレントディレクトリの世代管理は行われません。

Interstage Application Server V5.x以前

カレントディレクトリの世代管理は行われません。

5.8 JDK/JRE、Javaツールの変更内容

ここでは、JDK/JRE、Javaツールについて、以下を説明します。

- [デフォルトJDK/JREの変更](#)
- [jheapの変更点](#)
- [Qualyzerの変更点](#)

■デフォルトJDK/JREの変更

Interstage Application Server V9.0以降では、JDK/JRE 5.0が標準になりました。

インストール時に標準インストールを選択した場合は、JDK/JRE 1.4およびJDK/JRE 6はインストールされません。このため、V8からのバックアップ・リストアなどの移行時には注意してください。

JDK/JRE 1.4およびJDK/JRE 6をインストールしたい場合は、カスタムインストールでインストールしてください。

Interstage Application Server 8.0でJDK/JREのパッケージ名が変更になりました。新しいパッケージ名は、“使用上の注意”の“JDK/JREの注意事項”の“JDK/JREのインストール先”を参照してください。

デフォルトJDK/JREの変更に伴って、複数のJDK/JREをインストールした場合に優先的に使用するJDKが変更になります。

変更点	Interstage V6.0	Interstage V7.0以降	Interstage V9.0以降
標準インストールでインストールされるJDK/JRE (注1)	JDK 1.3	JDK 1.4	JDK 5.0
複数JDK/JREをインストールする場合、JDK/JRE選択時に表示されるJDK/JREのバージョン (注1)	JDK 1.3	JDK 1.4	JDK 5.0
複数JDK/JREをインストールした場合、IJSerで使用するJDKのデフォルト (注1)	JDK 1.3 【補足】 Interstage管理コンソールの[ワークユニット]>[新規作成]画面で、ワークユニット設定のJavaバージョンはデフォルトで空欄表示となる。この場合、以下の優先順位で使用するJDK/JREを決定するため、デフォルトでJDK 1.3が使用される。 1. JDK 1.3 2. JRE 1.3 3. JDK 1.4 4. JRE 1.4	複数JDK/JREをインストールした場合、JDK/JRE選択時に表示されるJDK/JREのバージョンが通常使用するJDK/JREのバージョンとして使用される。	IJSer(J2EE)では、JDK/JRE 5.0のみ使用可能。 V8互換IJSerでは、JDK/JRE 1.4のみ使用可能。 IJSer(JavaEE)では、JDK/JRE 6のみ使用可能。 (注2)

注1)

本製品のインストールによって、システム環境変数CLASSPATHが自動的に設定されます。

なお、IJSerを使用する場合は、ワークユニット設定のJavaバージョンを変更するだけで、CLASSPATHも自動的に変更されます。このため、使用するJDKによりCLASSPATHを変更する必要はありません。

注2)

JDK/JRE 6については、Interstage Application Server V9.2からの機能です。

デフォルトJDKの変更に伴い、以下の注意が必要です。

・メモリ使用量の変更

異なるバージョンのJDK/JREにアプリケーションを移行する場合は、メモリ使用量が変動するため、必ずチューニングを実施してください。

メモリ使用量のチューニング方法は、“チューニングガイド”の“JDK/JREのチューニング”を参照してください。

・Shift_JISエンコーディングの変更

JDK/JRE 1.4.1よりShift_JISエンコーディングの扱いが変更になっています。

この変更に伴い、Javaプログラムで文字エンコーディングとして“Shift_JIS”、“ms_kanji”、“x-sjis”または“csshiftjis”を指定した部分を、“Windows-31J”に修正する必要があります。

詳細は、“使用上の注意”の“注意事項”の“JDK/JREの注意事項”の“JDK1.4.1からの文字エンコーディングの変更について”を参照してください。

- Solarisにおけるデフォルト文字エンコーディングの変更

JDK/JREのバージョンによって、Solarisにおけるデフォルトの文字エンコーディングが異なります。
java.lang.Stringを生成する際や、ファイル入出力を行うときは明示的に文字エンコーディングを指定する必要がある場合があります。
詳細は、“使用上の注意”の“注意事項”の“JDK/JREの注意事項”の“Solarisにおけるデフォルト文字エンコーディング”を参照してください。

- “floating-point assist”メッセージの抑制

V8まではプロセス内で実行される浮動小数点演算の結果によって、OSがシステムログに対して、以下のメッセージを出力する場合があります。

プロセス名(プロセスID): floating-point assist fault at ip XXX, isr YYY (XXX, YYY: 16進数)

V9では、Javaプロセスからの本メッセージの出力は抑止するように変更になりました。

■jheapの変更点 Windows

Interstage Application Server V7.0以降では、jheapのオプションの指定形式が変更となりました。

Interstage V6.0	jheap processid [interval]
本バージョン・レベル	jheap [-t] [-i interval] [-c count] processed

本バージョン・レベルでは、Interstage Application Server V6.0と同様のコマンド形式も指定可能ですが、新しい指定形式に移行されることをおすすめします。

jheapの詳細は、“トラブルシューティング集”の“jheap”を参照してください。

■Qualyzerの変更点

Interstage Application Server V9.2から添付されたJDK/JRE 6には、JavaツールのQualyzer機能は未提供です。

JDK/JRE 6には、JVMPI機能が提供されていません。

そのため、JVMPI機能を利用しているQualyzerは動作しないためです。

5.9 Interstage証明書環境の移行

ここでは、Interstage Application Server V7.0以前のInterstage証明書環境の移行時の注意について説明します。

Interstage Application Server 8.0以降では、SSLのセキュリティを強化したため、Interstage証明書環境のSSL定義(Interstage管理コンソールの[システム] > [セキュリティ] > [SSL])において、下記のような仕様変更があります。

■暗号アルゴリズムの追加サポート

[暗号化方法]において、AES暗号アルゴリズムとSC2000暗号アルゴリズムを含む暗号化方法が選択できるようになりました。ただし、今日ではまだAES暗号アルゴリズムをサポートしている製品は限られていますし、SC2000暗号アルゴリズムは富士通研究所で開発された独自の暗号アルゴリズムです。そのため、他製品とSSL通信する場合は、接続性を保証するために、必要に応じて他製品がサポートしている暗号化方式も選択するようにしてください。

■初期選択の変更

最近ではDES暗号アルゴリズムは安全とは言えなくなっており、より強い暗号アルゴリズムを利用することが求められてきています。そのため[暗号化方法]の初期設定で選択されているものから、DES暗号アルゴリズムを含む暗号化方法を外しました。SSL通信を行う製品やアプリケーションの仕様や設定により、DES暗号アルゴリズムを利用したSSL通信をする必要がある場合には、DES暗号アルゴリズムを選択するようにしてください。

また、SSL通信のプロトコルとしてはSSL3.0やTLS1.0を使用するのが主流となっており、SSL2.0は下位互換のために使用されます。そのため、[プロトコルバージョン]の初期設定で選択されているものから「SSL2.0」を外しました。SSL通信を行う製品やアプリケーションの仕様や設定により、SSL2.0で接続する必要がある場合には、「SSL2.0」を選択するようにしてください。

■警告メッセージの表示

[プロトコルバージョン]に「SSL2.0」を選択し、かつ、[クライアント認証]で「する(クライアント証明書を必ず認証する)」を選択した場合、[適用]ボタンを押すと警告メッセージが表示されるようになります。この設定では、ブラウザなどのクライアント側からクライアント自身の証明書が提示されなくてもサーバに接続ができるため、安全性を損なう可能性があるためです。SSL通信を行う製品やアプリケーションの仕様や設定を考慮したうえで、警告を無視して適用するか、または、SSL定義を変更するか、判断してください。

5.10 Interstage ディレクトリサービスの移行

以前のバージョン・レベルから改善されたInterstageディレクトリサービスの機能、および移行時の注意事項について説明します。

- [Interstage Application Server V9.2での変更内容](#)
- [Interstage Application Server V8.0での変更内容](#)

■Interstage Application Server V9.2での変更内容

■フィルタの修正

標準データベース使用時、フィルタ条件「>=」、「<=」の動作が、RDB使用時と異なっていたため、フィルタ条件の動作を変更できるように改善しました。

フィルタ条件	V9.1以前		V9.2	
	標準DB	RDB	標準DB	RDB
>=	より大きい	以上	「より大きい」、「以上」のどちらかを選択	以上
<=	より小さい	以下	「より小さい」、「以下」のどちらかを選択	以下

フィルタ条件の動作を変更する場合は、irepconfigコマンドを使用して、リポジトリの環境定義に、「old_filtersrule: no」を指定します。

irepconfigコマンド、および、リポジトリの環境定義項目の詳細は、「リファレンスマニュアル(コマンド編)」の“Interstage ディレクトリサービス運用コマンド”を参照してください。

■Interstage Application Server V8.0での変更内容

- [データの移行](#)
- [アプリケーションの移行](#)
- [LDAPコマンドの移行](#)
- [レプリケーション運用](#)
- [エントリ管理ツール](#)

■データの移行

使用するデータベースを変更する場合、エントリ情報を移行できません。このため、格納されているエントリ情報をLDIFファイルへ取り出し、新しい環境で資源を復元する必要があります。

1. [旧バージョン・レベルのリポジトリ内からデータを抽出](#)
2. [本バージョン・レベルでリポジトリを作成](#)
3. [本バージョン・レベルのリポジトリの起動](#)
4. [本バージョン・レベルのリポジトリへのデータの移入](#)

1) 旧バージョン・レベルのリポジトリ内からデータを抽出

移行するデータが存在するリポジトリに対して以下のコマンドを実行し、リポジトリのデータをLDIFファイルへ取り出します。コマンドの詳細については、旧バージョン・レベルの“リファレンスマニュアル(コマンド編)”の“Interstage ディレクトリサービス運用コマンド”を参照してください。

以下の例では、データをLDIFファイル“dir.ldif”へ取り出しています。



例

Windows

```
C:\Interstage\ID\Dir\sdk\C\bin\ldapsearch -h ホスト名 -p LDAPポート番号 -D DSAの管理者DN  
-w DSAの管理者DNのパスワード -b トップエントリ "(objectclass=*)" > dir.ldif
```

Solaris

Linux

```
/opt/FJSVidsdk/C/bin/ldapsearch -h ホスト名 -p LDAPポート番号 -D DSAの管理者DN -w DSAの管  
理者DNのパスワード -b トップエントリ "(objectclass=*)" > dir.ldif
```

— ldapsearchコマンドは、旧バージョン・レベルのInterstage ディレクトリサービスが動作するホスト上で実行してください。

2) 本バージョン・レベルでリポジトリを作成

本バージョン・レベルのInterstage ディレクトリサービスを構築するマシン上でリポジトリを作成します。リポジトリの作成は、Interstage 管理コンソールを使用して、[システム] > [サービス] > [リポジトリ] > [新規作成]タブから行います。リポジトリの作成については、“ディレクトリサービス運用ガイド”の“環境構築”を参照してください。

なお、リポジトリ作成時に指定する[公開ディレクトリ]には、以下の値を指定してください。

— 公開ディレクトリ

旧バージョン・レベルのリポジトリで指定されていた公開ディレクトリを指定してください。

3) 本バージョン・レベルのリポジトリの起動

作成したリポジトリを起動します。リポジトリの起動は、Interstage管理コンソールを使用して、[システム] > [サービス] > [リポジトリ] > [リポジトリ:状態]画面から行います。

4) 本バージョン・レベルのリポジトリへのデータの移入

起動したリポジトリに対して以下のコマンドを実行し、旧バージョン・レベルのリポジトリから取り出したデータを本バージョン・レベルのリポジトリへ移入します。コマンドの詳細については、本バージョン・レベルの“リファレンスマニュアル(コマンド編)”の“Interstage ディレクトリサービス運用コマンド”を参照してください。

以下の例では、旧バージョン・レベルのリポジトリからLDIFファイル“dir.ldif”へ取り出したデータを移入しています。



例

Windows

```
C:\Interstage\bin\ldapmodify -h ホスト名 -p 通常ポート番号(またはSSLポート番号) -D リポジトリの  
管理者用DN -w リポジトリの管理者用DNのパスワード -c -a -f dir.ldif
```

Solaris

Linux

```
/opt/FJSVirepc/bin/ldapmodify -h ホスト名 -p 通常ポート番号(またはSSLポート番号) -D リポジトリの  
管理者用DN -w リポジトリの管理者用DNのパスワード -c -a -f dir.ldif
```

— 通常ポート番号(またはSSLポート番号)、リポジトリの管理者用DNについては、リポジトリの作成時に指定した値を指定してください。または、Interstage管理コンソールのリポジトリタグで、作成したリポジトリの設定項目を確認してください。

- ldapmodifyコマンドは、本バージョン・レベルのInterstage ディレクトリサービスが動作するホスト上で実行してください。

■アプリケーションの移行

旧バージョンのInterstage ディレクトリサービスでは、InfoDirectory SDKを使用していましたが、Interstage Application Server 8.0からInterstage ディレクトリサービス SDKを使用するように変更しています。2つのSDKには、バイナリ互換がありませんので、C言語アプリケーションは、再コンパイルが必要です。また、一部の機能、インタフェースに変更があります。詳細は、“[4.3.4 C言語アプリケーションの移行手順](#)”を参照してください。

Java言語(JNDI)アプリケーションは、修正の必要はありません。

■LDAPコマンドの移行

旧バージョン・レベルのInterstage ディレクトリサービスのLDAPコマンドでは、InfoDirectory SDKを使用していましたが、Interstage Application Server 8.0からInterstage ディレクトリサービス SDKを使用するように変更しています。LDAPコマンドにも変更がありますので、“[4.3.6 LDAPコマンドの移行](#)”で確認してください。

■レプリケーション運用

レプリケーション形態での、マスタのマシンとスレーブのマシンのInterstageは、同一のバージョン・レベルのものを使用してください。また、マスタのマシンとスレーブのマシンで使用するリポジトリのデータベースのタイプも同一にする必要があります。

■エントリ管理ツール

エントリ管理ツールの「接続情報設定」はバックアップできません。新しい環境で再設定が必要です。

5.11 Interstage シングル・サインオンの移行

ここでは、Interstage シングル・サインオンの移行について、以下を説明します。

- 変更点
 - [Interstage Application Server V9.1での変更内容](#)
 - [Interstage Application Server V9.0での変更内容](#)
 - [Interstage Application Server 8.0での変更内容](#)
 - [Interstage Application Server V7.0、およびInterstage Application Server Standard Edition、Enterprise Edition V6.0での変更内容](#)
- [旧バージョン・レベルからのサーバ環境の移行](#)
- [Javaアプリケーションの移行](#)

■Interstage Application Server V9.1での変更内容

◆Active Directoryとの連携

Interstage Application Server V9.1から、ユーザ情報を管理するディレクトリサービスとしてActive Directoryを使用することができます。旧バージョン・レベルから移行してActive Directoryを使用する場合は、“[旧バージョン・レベルからのサーバ環境の移行](#)”を行った後、Active Directoryと連携するための変更を行ってください。

Active Directoryと連携するための変更については、“シングル・サインオン運用ガイド”の“旧バージョンの環境定義と機能について” — “ディレクトリサービスにActive Directoryを使用するシステムへの移行について”を参照してください。

また、Active Directoryとの連携については、“シングル・サインオン運用ガイド”の“概要” — “Active Directoryとの連携”を参照してください。

◆ユーザ情報通知の強化

Interstage Application Server V9.1から、Webアプリケーションへのユーザ情報の通知方法を変更しました。Interstage Application Server V5.1から移行した場合、拡張ユーザ情報の通知方法が以下のように異なります。

環境	属性値のURL エンコード	バイナリーデータの通知	複数の属性値の通知
V5.1	なし	可 (サイズ制限あり)	不可
V9.1 (セッション管理 を行わない)	なし	可 (サイズ制限あり)	不可
V9.1 (セッション管理 を行う)	あり	可	可 (区切り文字はカンマ (,))

業務サーバの環境定義ファイルに以下の定義項目を追加することで、拡張ユーザ情報の通知方法を変更することができます。

環境定義ファイル

Windows

C:\¥Interstage¥F3FMsoo¥ssoatzag¥conf¥ssoatzag.conf

Solaris

Linux

/etc/opt/FJSVsssoaz/conf/ssoatzag.conf

追加定義

以下の定義項目が追加できます。

- ・credential-extra-info-compatibility=YES

V5.1の通知方法で通知します。

- ・credential-extra-info-compatibility=NO

V9.1 (セッション管理を行う)の通知方法で通知します。



ポイント

“credential-extra-info-compatibility”を省略した場合、または誤った定義を追加した場合は、セッション管理の運用によって以下のように動作します。

- ・セッション管理を行う場合
V9.1 (セッション管理を行う)の通知方法で通知します。
- ・セッション管理を行わない場合
V5.1の通知方法で通知します。

◆セッションの管理を行うシステムにおける証明書認証

Interstage Application Server V9.1からセッションの管理を行うシステムで証明書認証ができます。旧バージョン・レベルから移行してセッションの管理を行う場合は、“[旧バージョン・レベルからのサーバ環境の移行](#)”を行った後、セッションの管理を行うシステムで証明書認証を行うための設定を行ってください。

セッションの管理を行うシステムで証明書認証を行うための設定については、“[シングル・サインオン運用ガイド](#)”の“セッションの管理を行うシステムで証明書認証を行うための設定”を参照してください。

◆IPv6環境における運用

Interstage Application Server V9.1からIPv6環境で運用することができます。IPv6環境で運用する場合は、“使用上の注意”の“注意事項”－“Interstage シングル・サインオンの注意事項”－“バージョン・エディション混在でシングル・サインオンシステムを構築する場合の注意事項”を参照し、使用できるバージョン、エディションを確認してください。

■Interstage Application Server V9.0での変更内容

◆リポジトリサーバ(更新系)の負荷分散

Interstage Application Server V9.0から、リポジトリサーバ(更新系)を複数配置することで、認証やセッション評価などの要求に対して負荷分散を行うことができます。旧バージョン・レベルから移行してリポジトリサーバ(更新系)による負荷分散を行う場合は、“[旧バージョン・レベルからのサーバ環境の移行](#)”を行った後、リポジトリサーバ(更新系)の負荷分散を行うための変更を行ってください。

リポジトリサーバ(更新系)の負荷分散を行うための変更については、“シングル・サインオン運用ガイド”の“旧バージョンの環境定義と機能について”―“リポジトリサーバ(更新系)の負荷分散を行うシステムへの移行について”を参照してください。

また、Interstage シングル・サインオンにおけるリポジトリサーバ(更新系)の負荷分散については、“シングル・サインオン運用ガイド”の“概要”―“高性能・高信頼性システム”―“負荷分散”を参照してください。

また、Interstage Application Server V9.0から、リポジトリサーバ(更新系)の負荷分散を行うための情報をSSOリポジトリに書き込むために、旧バージョンと比較して、若干の認証性能の低下が発生します。

旧バージョン・レベルから移行した後に、リポジトリサーバ(更新系)の負荷分散を行わない場合には、リポジトリサーバの環境定義ファイルに以下の定義項目を追加することで、性能低下を発生させないようにすることができます。

環境定義ファイル

Windows

C:\¥Interstage¥F3FMso¥ssoatcsv¥conf¥ssoatcsv.conf

Solaris

Linux

/etc/opt/FJSVssosv/conf/ssoatcsv.conf

追加定義

以下の定義項目が追加できます。

・load-balancing-for-update-repository=NO

リポジトリサーバ(更新系)の負荷分散を行いません。



ポイント

“load-balancing-for-update-repository=NO”を追加した場合、リポジトリサーバ(更新系)の負荷分散を行いません。リポジトリサーバ(更新系)の負荷分散を行う場合には、定義しないでください。

◆環境変数によるユーザ情報の通知

Interstage Application Server V9.0から、WebサーバにInterstage HTTP Serverを使用している場合、以下のInterstage シングル・サインオンの認証機能で認証された利用者のユーザ情報が環境変数に通知されるようになります。

ユーザ情報	環境変数名
認証方式	AUTH_TYPE
利用者のユーザID	REMOTE_USER

業務サーバの環境定義ファイルに以下の定義項目を追加することで、ユーザ情報の通知方法を変更することができます。

環境定義ファイル

Windows

C:\¥Interstage¥F3FMso¥ssoatzag¥conf¥ssoatzag.conf

Solaris

Linux

/etc/opt/FJSVssosz/conf/ssoatzag.conf

追加定義

以下の定義項目が追加できます。

・header-auth-type-default=“文字列”

認証方式に関係なく、AUTH_TYPEに“文字列”で指定された値を通知します。

“SSO_AUTH”を通知する場合は、以下のように定義します。

header-auth-type-default=SSO_AUTH

- set-http-header-auth-type=NO

AUTH_TYPEに認証方式を通知しません。

- set-http-header-uid=NO

REMOTE_USERに利用者のユーザIDを通知しません。



- “set-http-header-uid=NO”を追加した場合、監査証跡に利用者のユーザIDが記録されません。監査証跡機能を使用する場合は、定義しないでください。
- header-auth-type-defaultに以下の文字列を指定した場合、旧バージョン環境においてアプリケーションが取得できる文字列と同じ値が、AUTH_TYPEに通知されることになります。アプリケーションにおいて問題がないか十分に確認してください。
 - Basic
 - Digest
 - BASIC
 - DIGEST
 - FORM
 - CLIENT_CERT

◆Interstage HTTP Serverの複数Webサーバ機能、およびバーチャルホスト機能の使用

Interstage Application Server V9.0から、Interstage HTTP Serverの複数Webサーバ機能を使用して、Interstage シングル・サインオンのサーバを追加することができます。また、Interstage HTTP Serverのバーチャルホスト機能を使用して、業務サーバを追加することができます。

Interstage HTTP Serverの複数のWebサーバ、またはバーチャルホストに各サーバを作成する場合は、“シングル・サインオン運用ガイド”の“環境構築(SSO管理者編)”，または“環境構築(業務サーバ管理者編)”を参照してください。

◆ログイン構成ファイルの変更

Interstage Application Server V9.0から、Interstage HTTP Serverの複数Webサーバ機能、およびバーチャルホスト機能が使用できます。Interstage HTTP Serverの複数のWebサーバ、またはバーチャルホストを使用した環境でInterstage シングル・サインオンが提供するJAASを利用したJavaアプリケーションを使用する場合は、[■旧バージョン・レベルからのサーバ環境の移行](#)を行ってください。その後、ログイン構成ファイルのモジュールオプションに“serverport”を使用している場合は、“business-system-name”に変更してください。

ログイン構成ファイルについては、“シングル・サインオン運用ガイド”の“アプリケーションの開発”－“Javaアプリケーションの開発”－“アプリケーション実行環境の設定”－“ログイン構成ファイルの作成”を参照してください。

◆コマンドの変更

Interstage Application Server 8.0以前で提供していた以下のコマンドの扱いが変更されました。変更点を以下に示します。各コマンドの詳細については、“リファレンスマニュアル(コマンド編)”を参照してください。

コマンド	変更点
ssocloneac	• 本バージョンでは使用できません。ssobackupコマンド、およびssorestoreコマンドを使用してください。
ssocloneaz	• 互換コマンドとして提供しています。ssobackupコマンド、およびssorestoreコマンドを使用してください。 • コマンドの拡張子が“.bat”から“.exe”に変わっています。コマンドの機能については、旧バージョン・レベルからの変更はありません。
ssosetsvc	• リポジトリサーバが構築されているInterstage HTTP ServerのWebサーバ名を指定する引数“webname”が追加されました。引数“webname”は必ず指定しなければなりません。
ssounsetsvc	

■Interstage Application Server 8.0での変更内容

◆セッションの管理

Interstage Application Server 8.0からセッションの管理を行うことができます。旧バージョン・レベルから移行してセッションの管理を行う場合は、“[旧バージョン・レベルからのサーバ環境の移行](#)”を行った後、セッションの管理を行うための変更を行ってください。

セッションの管理を行うための変更については、“シングル・サインオン運用ガイド”の“旧バージョンの環境定義と機能について”―“セッションの管理を行う運用への移行について”を参照してください。

また、Interstage シングル・サインオンにおけるセッションの管理については、“シングル・サインオン運用ガイド”の“概要”―“認証”を参照してください。

■Interstage Application Server V7.0、およびInterstage Application Server Standard Edition、Enterprise Edition V6.0での変更内容

◆Interstage管理コンソール

Interstage Application Server Standard Edition、Enterprise Edition V6.0、およびInterstage Application Server V7.0以降では、Interstage 管理コンソールを使用して環境を設定します。Interstage Application Server Plus、Web-J Edition V6.0、およびInterstage Application Server V5.1の環境定義の項目とInterstage管理コンソールによる設定の対応については“シングル・サインオン運用ガイド”の“旧バージョンの環境定義と機能について”を参照してください。

■旧バージョン・レベルからのサーバ環境の移行

Interstage シングル・サインオンのシステムは以下のサーバから構成されます。ここでは、それぞれのサーバの移行方法について説明します。

- ・ リポジトリサーバ
- ・ 認証サーバ
- ・ 業務サーバ

1台のマシンに複数のサーバ(リポジトリサーバと認証サーバなど)を構築して運用している場合、各サーバは同時に移行してください。Interstage シングル・サインオンの各製品における1台のマシンに構築可能なサーバの組み合わせは以下になります。

製品	1台のマシンに構築可能なサーバの組み合わせ
Interstage Web Server	・ 業務サーバ
Interstage Application Server Standard-J Edition Interstage Application Server Enterprise Edition	・ 業務サーバ ・ 認証サーバ ・ リポジトリサーバ ・ リポジトリサーバ、および認証サーバ

また、リポジトリサーバ(更新系)とリポジトリサーバ(参照系)に分散して運用している場合、移行後の環境にて以下のいずれかの運用を行うと、リポジトリサーバ(参照系)は使用されなくなります。リポジトリサーバ(更新系)とリポジトリサーバ(参照系)に分散している場合は、リポジトリサーバ(更新系)を複数台設置して負荷分散を行うシステムへ移行することを推奨します。

- ・ 統合Windows認証だけで認証を行う場合
- ・ 以下をすべて満たす環境で、認証サーバ間連携を行う場合
 - ― 保護リソースは、相手シングル・サインオンシステムにだけ定義されていること。
 - ― 自シングル・サインオンシステムの認証基盤に直接アクセスして認証を行わないこと。

リポジトリサーバ(更新系)を複数台設置して負荷分散を行うシステムについては、“シングル・サインオン運用ガイド”の“概要”―“高性能・高信頼性システム”―“負荷分散”を参照してください。

リポジトリサーバ(更新系)の負荷分散を行うシステムへの移行方法については、“シングル・サインオン運用ガイド”の“旧バージョンの環境定義と機能について”―“リポジトリサーバ(更新系)の負荷分散を行うシステムへの移行について”を参照してください。

◆Interstage Application Server Standard Edition、Enterprise Edition V6.0、およびInterstage Application Server V7.0以降からのサーバの移行

以下の製品からのサーバの移行方法を説明します。

- Interstage Application Server V9.0以降
- Interstage Application Server V7.0/8.0
- Interstage Application Server Standard Edition V6.0
- Interstage Application Server Enterprise Edition V6.0

シングル・サインオンのシステムを構成するサーバの移行手順については以下を参照してください。

- [Interstage Application Server V9.0以降からのサーバの移行](#)
- [Interstage Application Server V7.0/8.0からのサーバの移行](#)
- [Interstage Application Server Standard Edition、Enterprise Edition V6.0からのサーバの移行](#)

◆Interstage Application Server Plus、Web-J Edition V6.0、およびInterstage Application Server V5.1からのサーバの移行

以下の製品からのサーバの移行方法を説明します。

- Interstage Application Server Plus V6.0
- Interstage Application Server Web-J Edition V6.0
- Interstage Application Server V5.1

シングル・サインオンのシステムを構成するサーバの移行手順については以下を参照してください。

- [Interstage Application Server Plus V6.0からのリポジトリサーバの移行](#)
- [Interstage Application Server Plus V6.0からの認証サーバの移行](#)
- [Interstage Application Server Plus V6.0からのリポジトリサーバ、および認証サーバの移行](#)
- [Interstage Application Server Plus、Web-J Edition V6.0からの業務サーバの移行](#)
- [Interstage Application Server V5.1からのサーバの移行](#)



ポイント

- 移行前の製品で提供されていたシングル・サインオンのリポジトリサーバ、認証サーバおよび業務サーバを高速化する設定については、移行する必要はありません。
- Interstage Application Server V5.1からの移行において、各サーバの環境定義ファイルのアクセスログ出力先ファイル名にUTF形式で257バイト以上のファイル名を指定している場合には、UTF形式で256バイト以下のファイル名に変更する必要があります。

■Javaアプリケーションの移行

Interstage シングル・サインオンが提供するJAASを利用したJavaアプリケーションの移行手順については以下を参照してください。

- [Javaアプリケーションの移行](#)

5.11.1 Interstage Application Server V9.0以降からのサーバの移行

以下の製品からのサーバの移行方法を説明します。

- Interstage Application Server V9.0以降

■移行手順

バックアップ、およびリストアにより移行します。

バックアップ、およびリストアの作業については“運用ガイド(基本編)”の“メンテナンス(資源のバックアップ)”を参照してください。



ポイント

・リポジトリサーバの移行について

- リポジトリサーバを複数構築して運用している場合は、同時に移行してバージョン・レベルを合わせてください。
- 旧バージョン環境にてssosetsvcコマンドを使用してリポジトリサーバのサービス依存関係の設定を行なっている場合は、移行後、再度ssosetsvcコマンドを使用してリポジトリサーバのサービス依存関係の設定を行なってください。

・認証サーバの移行について

- 認証サーバを複数構築して運用している場合は、同時に移行してバージョン・レベルを合わせてください。

・業務サーバの移行について

- Interstage証明書環境が作成されていない場合は、Interstage証明書環境を作成してください。Interstage証明書環境の作成については“シングル・サインオン運用ガイド”の“SSL通信で運用するための準備”―“Interstage証明書環境の作成”を参照してください。
- 負荷分散のために、業務サーバを複数構築して運用している場合は、同時に移行してバージョン・レベルを合わせてください。

Solaris

- WebサーバにSun Java System Web Server 6.0を使用している場合は、Sun Java System Web Serverの環境定義ファイルを編集してください。環境定義ファイルの編集方法については、“シングル・サインオン運用ガイド”の“環境構築(業務サーバ管理者編)”―“Webサーバへの組み込み”を参照してください。

5.11.2 Interstage Application Server V7.0/8.0からのサーバの移行

以下の製品からのサーバの移行方法を説明します。

- ・ Interstage Application Server V7.0/8.0

■移行手順

バックアップ、およびリストアにより移行します。

ただし、上記製品でバックアップした各サーバの資源を、本バージョンでリストアする場合、ssorestoreコマンドを使用することはできません。以下の手順でリストアしてください。

■リストア方法(リポジトリサーバの資源)

バックアップ用ディレクトリのリポジトリサーバの資源をもとのディレクトリにコピーします。

Windows

1.xcopyコマンド(またはエクスプローラ)を使用して、バックアップ用ディレクトリのリポジトリサーバの資源をもとのディレクトリにコピーします。

```
xcopy /E /I X:\Backup¥ssoatcsv¥conf C:\¥Interstage¥F3FM¥ssoatcsv¥conf
```

2.リポジトリサーバの環境をリストアしたマシンのInterstage管理コンソールを使用して、[システム]>[セキュリティ]>[シングル・サインオン]>[認証基盤]>[リポジトリサーバ]>[環境設定]タブの[適用]ボタンをクリックします。

Solaris

Linux

1. cpコマンドを使用して、バックアップ用ディレクトリのリポジトリサーバの資源をもとのディレクトリにコピーします。
cp -rp /backup/FJSVssosv/conf /etc/opt/FJSVssosv
2. リポジトリサーバの環境をリストアしたマシンのInterstage管理コンソールを使用して、[システム] > [セキュリティ] > [シングル・サインオン] > [認証基盤] > [リポジトリサーバ] > [環境設定]タブの[適用]ボタンをクリックします。

■リストア方法(認証サーバの資源)

バックアップ用ディレクトリの認証サーバの資源(ディレクトリ)をもとのディレクトリにコピーします。

Windows

- 1.xcopyコマンド(またはエクスプローラ)を使用して、バックアップ用ディレクトリの認証サーバの資源をもとのディレクトリにコピーします。
xcopy /E /I X:\Backup¥ssoatcag¥conf C:\¥Interstage¥F3FMssos¥ssoatcag¥conf
- 2.xcopyコマンド(またはエクスプローラ)を使用して、バックアップ用ディレクトリの認証サーバの資源(ディレクトリ)をもとのディレクトリにコピーします。
xcopy /E /I X:\Backup¥ssoatcag¥template C:\¥Interstage¥F3FMssos¥ssoatcag¥pub¥template
3. 認証サーバの環境をリストアしたマシンのInterstage管理コンソールを使用して、[システム] > [セキュリティ] > [シングル・サインオン] > [認証基盤] > [認証サーバ] > [環境設定]タブの[適用]ボタンをクリックします。(注)

Solaris

Linux

- 1.cpコマンドを使用して、バックアップ用ディレクトリの認証サーバの資源をもとのディレクトリにコピーします。
cp -rp /backup/FJSVssosac/conf /etc/opt/FJSVssosac
- 2.cpコマンドを使用して、バックアップ用ディレクトリの認証サーバの資源をもとのディレクトリにコピーします。
cp -rp /backup/FJSVssosac/template /etc/opt/FJSVssosac/pub
3. 認証サーバの環境をリストアしたマシンのInterstage管理コンソールを使用して、[システム] > [セキュリティ] > [シングル・サインオン] > [認証基盤] > [認証サーバ] > [環境設定]タブの[適用]ボタンをクリックします。(注)

(注)セッションの管理を行っていない場合は、[詳細設定[表示]]をクリックし、[認証基盤の情報] > [認証基盤のURL]に認証基盤のURLを指定してください。

■リストア方法(業務サーバの資源)

バックアップ用ディレクトリの業務サーバの資源(ディレクトリ)をもとのディレクトリにコピーします。

Windows

- 1.xcopyコマンド(またはエクスプローラ)を使用して、バックアップ用ディレクトリの業務サーバの資源をもとのディレクトリにコピーします。
xcopy /E /I X:\Backup¥ssoatzag¥conf C:\¥Interstage¥F3FMssos¥ssoatzag¥conf
- 2.xcopyコマンド(またはエクスプローラ)を使用して、バックアップ用ディレクトリの業務サーバの資源(ディレクトリ)をもとのディレクトリにコピーします。
xcopy /E /I X:\Backup¥ssoatzag¥template C:\¥Interstage¥F3FMssos¥ssoatzag¥pub¥template
3. 業務サーバの環境をリストアしたマシンのInterstage管理コンソールを使用して、[システム] > [セキュリティ] > [シングル・サインオン] > [業務システム] > [業務システム名] > [環境設定]タブの[適用]ボタンをクリックします。

Solaris

Linux

- 1.cpコマンドを使用して、バックアップ用ディレクトリの業務サーバの資源をもとのディレクトリにコピーします。
cp -rp /backup/FJSVssosaz/conf /etc/opt/FJSVssosaz
- 2.cpコマンドを使用して、バックアップ用ディレクトリの業務サーバの資源をもとのディレクトリにコピーします。
cp -rp /backup/FJSVssosaz/template /etc/opt/FJSVssosaz/pub

3. 業務サーバの環境をリストアしたマシンのInterstage管理コンソールを使用して、[システム] > [セキュリティ] > [シングル・サインオン] > [業務システム] > [業務システム名] > [環境設定]タブの[適用]ボタンをクリックします。

■リストア方法(Javaアプリケーション)

バックアップ用ディレクトリのJavaアプリケーションの資源をもとのディレクトリにコピーします。

Windows

copyコマンド(またはエクスプローラ)を使用して、バックアップ用ディレクトリのJavaアプリケーションの資源をもとのディレクトリにコピーします。
Javaアプリケーションの格納ディレクトリがC:\Interstage\F3FM\sssoat\zag\sampleの場合
copy X:\Backup\sssoat\zag\java\Javaアプリケーション資源ファイル C:\Interstage\F3FM\sssoat\zag\sample
(注)

Solaris

Linux

cpコマンドを使用して、バックアップ用ディレクトリのJavaアプリケーションの資源をもとのディレクトリにコピーします。
Javaアプリケーションの格納ディレクトリが/home/jaas/sampleの場合
cp -p /backup/FJSV\sssoaz/java/Javaアプリケーション資源ファイル /home/jaas/sample (注)

(注)Javaアプリケーションを開発する場合にリストアしてください。



ポイント

・リポジトリサーバの移行について

- リポジトリサーバの資源をリストアする場合は、以下の資源を事前にリストアしてください。
 - Interstage HTTP Server
Webサーバ(Interstage HTTP Server)の移行については、“[4.1.1 8.0以前のInterstage HTTP Server \(Apache HTTP Server 1.3ベース\)からの移行](#)”を参照してください。
 - Interstage ディレクトリサービス
SSOリポジトリとして使用しているInterstage ディレクトリサービスのV7.0以前からの移行については“[5.10 Interstage ディレクトリサービスの移行](#)”を参照してください。
- Interstage管理コンソールで構築したInterstage証明書環境を使用してSSL通信を行っている場合は、バックアップしたInterstage証明書環境資源をリストアする必要があります。Interstage証明書環境のV7.0以前からの移行については、“[5.9 Interstage証明書環境の移行](#)”を参照してください。
- リポジトリサーバを複数構築して運用している場合は、同時に移行してバージョン・レベルを合わせてください。
- 旧バージョン環境にてssosetsvcコマンドを使用してリポジトリサーバのサービス依存関係の設定を行なっている場合は、移行後、再度ssosetsvcコマンドを使用してリポジトリサーバのサービス依存関係の設定を行なってください。
- 移行後、リポジトリサーバで使用しているWebサーバ(Interstage HTTP Server)のホストは、Interstage管理コンソールの[システム] > [サービス] > [Webサーバ] > [FJapache] > [バーチャルホスト] > [_default_]リポジトリサーバのポート番号]で確認できます。

・認証サーバの移行について

- 認証サーバの資源をリストアする場合は、Interstage HTTP Serverの資源を事前にリストアしてください。Webサーバ(Interstage HTTP Server)の移行については、“[4.1.1 8.0以前のInterstage HTTP Server \(Apache HTTP Server 1.3ベース\)からの移行](#)”を参照してください。
- Interstage管理コンソールで構築したInterstage証明書環境を使用してSSL通信を行っている場合は、バックアップしたInterstage証明書環境資源をリストアする必要があります。Interstage証明書環境のV7.0以前からの移行については、“[5.9 Interstage証明書環境の移行](#)”を参照してください。
- 認証サーバを複数構築して運用している場合は、同時に移行してバージョン・レベルを合わせてください。

- ー 移行後、認証サーバで使用しているWebサーバ(Interstage HTTP Server)のホストは、Interstage管理コンソールの[システム] > [サービス] > [Webサーバ] > [FJapache] > [バーチャルホスト] > [_default_]認証サーバのポート番号]で確認できます。

・ 業務サーバの移行について

- ー 業務サーバの資源をリストアする場合は、ご利用のWebサーバの資源も合わせてリストアしてください。
- ー Interstage証明書環境が作成されていない場合は、Interstage証明書環境を作成してください。Interstage証明書環境の作成については“シングル・サインオン運用ガイド”の“SSL通信で運用するための準備” – “Interstage証明書環境の作成”を参照してください。
- ー 負荷分散のために、業務サーバを複数構築して運用している場合は、同時に移行してバージョン・レベルを合わせてください。

Windows

- ー 8.0から移行した環境で、WebサーバにMicrosoft(R) Internet Information Services 7.0を使用する場合は、以下に注意してください。
 - Webブラウザに表示するメッセージのステータスコードを必ず変更してください。ステータスコードの変更については、“[Webブラウザに表示するメッセージのステータスコードの変更](#)”を参照してください。
 - Interstage シングル・サインオンの認可時に発生するエラー要因に対するメッセージのカスタマイズはできません。メッセージのカスタマイズについては、“シングル・サインオン運用ガイド”の“シングル・サインオンのカスタマイズ” – “Webブラウザに表示するメッセージのカスタマイズ”を参照してください。

Solaris

- ー WebサーバにSun Java System Web Server 6.0を使用している場合は、Sun Java System Web Serverの環境定義ファイルを編集してください。環境定義ファイルの編集方法については、“シングル・サインオン運用ガイド”の“環境構築(業務サーバ管理者編)” – “Webサーバへの組み込み”を参照してください。
- ー 本バージョン・レベルは、Solaris 8をサポートしていないため、Interstage V7.0以前でSun Java System Web Server 4.1を使用している場合は、別のWebサーバを使用してください。

5.11.3 Interstage Application Server Standard Edition、Enterprise Edition V6.0からのサーバの移行

以下の製品からのサーバの移行方法を説明します。

- ・ Interstage Application Server Standard Edition V6.0
- ・ Interstage Application Server Enterprise Edition V6.0

■移行手順

Interstage Application Server V7.0/8.0からのサーバの移行手順と同様です。“[5.11.2 Interstage Application Server V7.0/8.0からのサーバの移行](#)”を参照してください。

5.11.4 Interstage Application Server Plus V6.0からのリポジトリサーバの移行

リポジトリサーバの移行手順について説明します。

リポジトリサーバを複数構築して運用している場合は、同時に移行してバージョン・レベルを合わせてください。

■移行手順

移行は以下の手順で行います。

1. [リポジトリサーバ資源のバックアップ](#)
2. [SSOリポジトリからのデータの抽出](#)
3. [パッケージのアンインストールとインストール](#)
4. [SSOリポジトリの作成](#)

5. SSOリポジトリへのデータの移入
6. SSOリポジトリ(マスタ)のバックアップ
7. SSOリポジトリ(スレーブ)の作成
8. SSOリポジトリ(マスタ)の設定変更
9. リポジトリサーバ資源のリストア
10. リポジトリサーバ定義ファイルの編集
11. Webサーバ(Interstage HTTP Server)定義の編集
12. リポジトリサーバの環境設定

1) リポジトリサーバ資源のバックアップ

以下のリポジトリサーバの資源をバックアップします。

- リポジトリサーバ定義ファイル
- サービスIDファイル

リポジトリサーバを複数台で運用している場合、すべてのリポジトリサーバに対して、資源をバックアップしてください。資源のバックアップ時はWebサーバ(Interstage HTTP Server)を停止してください。



例

リポジトリサーバの資源をバックアップ用ディレクトリにコピーする例

Windows

バックアップ用ディレクトリ: X:¥Backup¥ssoatcsv
copyコマンド(またはエクスプローラ)を使用して、コピーします。

```
copy C:¥Interstage¥F3FMssos¥ssoatcsv¥conf¥ssoatcsv.conf X:¥Backup¥ssoatcsv
copy C:¥Interstage¥F3FMssos¥ssoatcsv¥conf¥serviceid X:¥Backup¥ssoatcsv (注)
```

Solaris

Linux

バックアップ用ディレクトリ: /backup/FJSVssosv
cpコマンドを使用して、コピーします。

```
cp -p /etc/opt/FJSVssosv/conf/ssoatcsv.conf /backup/FJSVssosv
cp -p /etc/opt/FJSVssosv/conf/serviceid /backup/FJSVssosv (注)
```

(注) サービスIDファイルはリポジトリサーバ定義ファイルの“serviceidpath”定義に設定したファイルを指定してください。

Webサーバ(Interstage HTTP Server)の移行については“4.1.1 8.0以前のInterstage HTTP Server (Apache HTTP Server 1.3ベース)からの移行”を参照してください。

なお、Webサーバ(Interstage HTTP Server)にInterstage シングル・サインオン以外のサービスを設定している場合は、設定している各サービスの移行方法に従ってください。

2) SSOリポジトリからのデータの抽出

本バージョン・レベルでは、SSOリポジトリにInterstage ディレクトリサービスを使用します。SSOリポジトリをInfoDirectoryからInterstage ディレクトリサービスに移行する際には、“使用上の注意”の“Interstage ディレクトリサービスの注意事項”ー“InfoDirectoryからの移行に関する注意事項”を参照し、注意事項を確認してください。使用できない文字などを指定している場合は、InfoDirectoryに登録したデータ、またはInfoDirectoryから抽出したデータを、Interstage ディレクトリサービスに移入できるデータに変更してください。

InfoDirectoryに登録したデータをすべて抽出します。データの抽出は、ldapsearchコマンドを実行して行います。コマンドの詳細については旧バージョン・レベルの“InfoDirectory使用手引書”を参照してください。

なお、リポジトリサーバを複数台で運用している場合、リポジトリサーバ(更新系)で、ldapsearchコマンドを実行してください。



例

InfoDirectoryに登録したデータを抽出する例

Windows

データの抽出先パス:X:\Backup

データの抽出ファイル:X:\Backup\dir.ldif

```
C:\Interstage\ID\Dir\sdks\bin\ldapsearch -h localhost -p ポート番号 -D DSA管理者dn -w DSA管理者dnに設定したパスワード -b トップエントリ "(objectclass=*)" > X:\Backup\dir.ldif
```

Solaris

Linux

データの抽出先パス:/backup

データの抽出ファイル:/backup/dir.ldif

```
/opt/FJSVidsdk/C/bin/ldapsearch -h localhost -p ポート番号 -D DSA管理者dn -w DSA管理者dnに設定したパスワード -b トップエントリ "(objectclass=*)" > /backup/dir.ldif
```

- ・ポート番号、DSA管理者dn、トップエントリについては、InfoDirectory管理ツールクライアントで確認してください。トップエントリはInterstage ディレクトリサービスでSSOリポジトリを作成する際に必要となります。
- ・ldapsearchコマンドは、InfoDirectoryのDSAが動作するホスト上で実行してください。

3) パッケージのアンインストールとインストール

現在インストールされているパッケージをアンインストール後、本バージョン・レベルのパッケージをインストールします。リポジトリサーバを複数台で運用している場合、すべてのリポジトリサーバに対して、パッケージのアンインストールとインストールを行ってください。

4) SSOリポジトリの作成

SSOリポジトリをInterstage ディレクトリサービスで作成します。SSOリポジトリの作成は構築するマシンのInterstage管理コンソールを使用して作成します。詳細については“シングル・サインオン運用ガイド”の“環境構築(SSO管理者編)” – “リポジトリサーバの構築” – “SSOリポジトリの作成”を参照してください。

なお、リポジトリサーバを複数台で運用している場合は、リポジトリサーバ(更新系)において、SSOリポジトリ(マスタ)をInterstage ディレクトリサービスで作成します。

また、リポジトリサーバを複数台で運用している場合において、レプリケーションをSSL通信で行うときは、あらかじめリポジトリサーバ(更新系)にSSL通信環境を構築しておく必要があります。構築方法については、“シングル・サインオン運用ガイド”の“環境構築(SSO管理者編)” – “リポジトリサーバの構築” – “SSOリポジトリ(マスタ)のSSL通信環境の構築”を参照してください。

SSOリポジトリ作成時には、以下に注意して設定してください。他の項目については“シングル・サインオン運用ガイド”を参照してください。

- ・ [公開ディレクトリ]
InfoDirectoryのDSAで指定されたトップエントリを指定します。
InfoDirectoryからInterstage ディレクトリサービスに移行する場合、公開ディレクトリに指定できる文字が制限されます。“2) SSOリポジトリからのデータの抽出”でデータを変更した場合は、変更後のディレクトリを指定してください。

5) SSOリポジトリへのデータの移入

Interstage ディレクトリサービスで作成したSSOリポジトリにInfoDirectoryから抽出したデータを移入します。ldapmodifyコマンドを実行し、InfoDirectoryから取り出したLDIFファイルのデータをInterstage ディレクトリサービスへ移入してください。コマンドの詳細については“リファレンスマニュアル(コマンド編)”を参照してください。

リポジトリサーバを複数台で運用している場合、リポジトリサーバ(更新系)で、ldapmodifyコマンドを実行します。



SSOリポジトリにデータを移入する例

例

Windows

データの抽出先パス:X:\Backup

データの抽出ファイル:X:\Backup\dir.ldif


```
C:\Interstage\bin\ldapmodify -h localhost -p 通常(非SSL)ポート番号 -D リポジトリ管理者用DN -W -a  
-c -f X:\Backup\dir.ldif
```

Solaris

Linux

データの抽出先パス:/backup

データの抽出ファイル:/backup/dir.ldif

```
/opt/FJSVirepc/bin/ldapmodify -h localhost -p 通常(非SSL)ポート番号 -D リポジトリ管理者用DN -W -a  
-c -f /backup/dir.ldif
```

パスワードの入力を求められますので、SSOリポジトリ作成時に設定した管理者用DNのパスワードを入力してください。

- ポート番号、リポジトリ管理者用DNについては、SSOリポジトリ作成時にInterstage管理コンソールで設定した値を指定してください。
- ldapmodifyコマンドは、Interstage ディレクトリサービスが動作するホスト上で実行してください。
- ldapmodifyコマンド実行後は、エントリ管理ツールなどを使用してデータの移入が正常に行われているか確認してください。データの移入については“ディレクトリサービス運用ガイド”(Interstage Application Server Enterprise Editionで提供)の“エントリの管理”を参照してください。
- InfoDirectoryより抽出したデータに“userSMIMECertificate;binary:”が含まれる場合は、“userCertificate;binary:”に変換してからデータを移入してください。

6) SSOリポジトリ(マスタ)のバックアップ

リポジトリサーバを複数台で運用している場合にのみ必要な作業です。

リポジトリサーバ(参照系)のSSOリポジトリ(スレーブ)を作成するため、リポジトリサーバ(更新系)のSSOリポジトリ(マスタ)のデータをバックアップします。バックアップの方法については“シングル・サインオン運用ガイド”の“環境構築(SSO管理者編)”-“リポジトリサーバの構築”-“リポジトリサーバ(参照系)の追加”-“リポジトリサーバ(更新系)のSSOリポジトリのバックアップ”を参照してください。

7) SSOリポジトリ(スレーブ)の作成

リポジトリサーバを複数台で運用している場合にのみ必要な作業です。

リポジトリサーバ(参照系)において、SSOリポジトリ(スレーブ)をInterstage ディレクトリサービスで作成します。SSOリポジトリ(スレーブ)の作成は構築するマシンのInterstage管理コンソールを使用して作成します。

また、レプリケーションをSSL通信で行う場合は、あらかじめ、すべてのリポジトリサーバ(参照系)にSSL通信環境を構築しておく必要があります。構築方法については、“シングル・サインオン運用ガイド”の“環境構築(SSO管理者編)”-“リポジトリサーバの構築”-“リポジトリサーバ(参照系)の追加”-“SSOリポジトリ(スレーブ)のSSL通信環境の構築”を参照してください。

以下に、SSOリポジトリ(スレーブ)の作成手順を示します。以下の手順は、すべてのリポジトリサーバ(参照系)に対して行ってください。

1. SSOリポジトリ(スレーブ)の作成

リポジトリサーバ(参照系)でSSOリポジトリ(スレーブ)を作成します。

作成方法については“シングル・サインオン運用ガイド”の“環境構築(SSO管理者編)”-“リポジトリサーバの構築”-“リポジトリサーバ(参照系)の追加”-“リポジトリサーバ(参照系)のSSOリポジトリ(スレーブ)の作成”を参照してください。

2. SSOリポジトリ(スレーブ)へのリストア

バックアップしたSSOリポジトリ(マスタ)のデータをリストアします。

リストアの方法については“シングル・サインオン運用ガイド”の“環境構築(SSO管理者編)”-“リポジトリサーバの構築”-“リポジトリサーバ(参照系)の追加”-“リポジトリサーバ(参照系)へのSSOリポジトリのリストア”を参照してください。

3. SSOリポジトリ(スレーブ)の設定変更

SSOリポジトリ(スレーブ)に、レプリケーションを行うための設定を行います。

設定方法については“シングル・サインオン運用ガイド”の“環境構築(SSO管理者編)”-“リポジトリサーバの構築”-“リポジトリサーバ(参照系)の追加”-“リストアしたリポジトリサーバ(参照系)のSSOリポジトリの設定変更”を参照してください。

なお、SSOリポジトリ(スレーブ)作成時には、以下に注意して設定してください。他の項目については“シングル・サインオン運用ガイド”を参照してください。

- ・ [公開ディレクトリ]
InfoDirectoryのDSAで指定されたトップエントリを指定します。
InfoDirectoryからInterstage ディレクトリサービスに移行する場合、公開ディレクトリに指定できる文字が制限されます。“2) SSOリポ
ジトリからのデータの抽出”でデータを変更した場合は、変更後のディレクトリを指定してください。

8) SSOリポジトリ(マスタ)の設定変更

リポジトリサーバを複数台で運用している場合にのみ必要な作業です。
リポジトリサーバ(更新系)のSSOリポジトリ(マスタ)に、レプリケーションを行うための設定を行います。設定方法については“シングル・
サインオン運用ガイド”の“環境構築(SSO管理者編)”－“リポジトリサーバの構築”－“リポジトリサーバ(参照系)の追加”－“リポジトリ
サーバ(更新系)のSSOリポジトリの設定変更”を参照してください。

9) リポジトリサーバ資源のリストア

“1) リポジトリサーバ資源のバックアップ”でバックアップしたリポジトリサーバの資源をリストアします。
リポジトリサーバを複数台で運用している場合、すべてのリポジトリサーバに対して、資源をリストアしてください。



バックアップ用ディレクトリのリポジトリサーバの資源をリストアする例

例

Windows

バックアップ用ディレクトリ:X:\Backup¥ssoatcsv
copyコマンド(またはエクスプローラ)を使用して、コピーします。

```
copy X:\Backup¥ssoatcsv¥ssoatcsv.conf C:\Interstage¥F3FMso¥ssoatcsv¥conf
copy X:\Backup¥ssoatcsv¥serviceid C:\Interstage¥F3FMso¥ssoatcsv¥conf (注)
```

Solaris

Linux

バックアップ用ディレクトリ:/backup/FJSVssosv
cpコマンドを使用して、コピーします。

```
cp -p /backup/FJSVssosv/ssoatcsv.conf /etc/opt/FJSVssosv/conf
cp -p /backup/FJSVssosv/serviceid /etc/opt/FJSVssosv/conf (注)
```

(注) サービスIDファイルは必ず上記にリストアし、リポジトリサーバ定義ファイルの“serviceidpath”定義に設定してください。

10) リポジトリサーバ定義ファイルの編集

alternative-certificate-attribute定義に“userSMIMECertificate;binary”を設定している場合は、リポジトリサーバ定義ファイルをリストア後、
以下のいずれかを行ってください。

- ・ alternative-certificate-attribute定義を削除してください。
- ・ alternative-certificate-attribute定義の設定を“userCertificate;binary”に変更してください。

また、accesslog-filename定義を以下に変更してください。

Windows

```
accesslog-filename=C:\Interstage¥F3FMso¥ssoatcsv¥log¥ssoatcsv.log
```

Solaris

Linux

```
accesslog-filename=/var/opt/FJSVssosv/log/ssoatcsv.log
```

11) Webサーバ(Interstage HTTP Server)定義の編集

Webサーバ(Interstage HTTP Server)の移行作業を実施後、Interstage HTTP Serverの環境定義ファイル(httpd.conf)に設定されてい
るリポジトリサーバの設定を削除し、以下の設定を追加してください。なお、ポート番号は移行前に使用していたポート番号を設定して

ください。

リポジトリサーバを複数台で運用している場合、すべてのリポジトリサーバに対して、本設定を行ってください。

削除する設定については、旧バージョン・レベルの“シングル・サインオン運用ガイド”の“環境構築”－“リポジトリサーバの環境構築”－“Interstage HTTP Serverの環境定義ファイルの設定”を参照してください。

なお、Interstage HTTP Serverの環境定義の編集に失敗している場合は、[システム] > [セキュリティ] > [シングル・サインオン] > [認証基盤] > [一覧]に表示されるリポジトリサーバのポート番号に“不明”と表示されます。編集内容に間違いがないか確認してください。

環境定義ファイルに追加する項目

項目名	内容
Listen	リポジトリサーバが起動するときのネットワークポート番号
LoadModule	リポジトリサーバのプログラム
<VirtualHost _default_:ポート番号> <Location> SetHandler </Location> DocumentRoot </VirtualHost>	リポジトリサーバの実行に必要な項目



Interstage HTTP Serverの環境定義ファイル(httpd.conf)の設定例

例

環境定義ファイルの最終行に以下の記述を追加し、太字部分(ポート番号)を運用に合わせて変更してください。なお、以下はリポジトリサーバを運用するポート番号に“**10550**”を使用する場合を例にしています。

Windows

```
Listen 10550
LoadModule ssoatcsv_module "C:¥Interstage¥F3FMso¥ssoatcsv¥lib¥F3FMsoatcsv.dll"
<VirtualHost _default_:10550>
<Location /ssoatcsv>
SetHandler ssoatcsv-authenticate-handler
</Location>
<Location /resinfo>
SetHandler ssoatcsv-resourceinfo-handler
</Location>
<Location /roleref>
SetHandler ssoatcsv-rolerefresh-handler
</Location>
DocumentRoot "C:¥Interstage¥F3FMso¥ssoatcsv¥pub¥docroot"
</VirtualHost>
```

Solaris

Linux

```
Listen 10550
LoadModule ssoatcsv_module "/opt/FJSVssosv/lib/ssoatcsv.so"
<VirtualHost _default_:10550>
<Location /ssoatcsv>
SetHandler ssoatcsv-authenticate-handler
</Location>
<Location /resinfo>
SetHandler ssoatcsv-resourceinfo-handler
</Location>
<Location /roleref>
SetHandler ssoatcsv-rolerefresh-handler
</Location>
```

```
DocumentRoot "/opt/FJSVssosv/pub/docroot"
</VirtualHost>
```

12) リポジトリサーバの環境設定

環境移行後、リポジトリサーバの環境を移行したマシンのInterstage管理コンソールから以下の手順で環境を設定します。各項目の詳細についてはInterstage管理コンソールのヘルプを参照してください。

なお、リポジトリサーバを複数台で運用している場合、本設定は、すべてのリポジトリサーバに対して行う必要があります。

1. [システム] > [セキュリティ] > [シングル・サインオン] > [認証基盤] > [リポジトリサーバ] > [環境設定] タブ > [リポジトリサーバ詳細設定[表示]]、または[詳細設定[表示]](リポジトリサーバを複数台で運用している場合)をクリックし、[認証基盤の情報]の[認証基盤のURL]に業務システムからアクセスを受け付ける認証基盤のURLを設定します。認証基盤のURLについては“シングル・サインオン運用ガイド”の“概要”－“URLの決定”－“認証基盤のURLについて”を参照してください。
2. [リポジトリ]の[リポジトリ名]にSSOリポジトリとして使用するInterstage ディレクトリサービスのリポジトリ名を選択します。SSOリポジトリ作成時に指定した“リポジトリ名”を選択してください。
3. [適用]ボタンをクリックします。

5.11.5 Interstage Application Server Plus V6.0からの認証サーバの移行

認証サーバの移行手順について説明します。

認証サーバを複数構築して運用している場合は、同時に移行してバージョン・レベルを合わせてください。

■移行手順

移行は以下の手順で行います。

1. [認証サーバ資源のバックアップ](#)
2. [SSL通信を行うための環境のバックアップ](#)
3. [パッケージのアンインストールとインストール](#)
4. [SSL通信を行うための環境のリストア、または環境の構築](#)
5. [認証サーバ資源のリストア](#)
6. [認証サーバ定義ファイルの編集](#)
7. [Webサーバ\(Interstage HTTP Server\)定義の編集](#)
8. [認証サーバの環境設定](#)

1) 認証サーバ資源のバックアップ

以下の認証サーバの資源をバックアップします。

- ・ 認証サーバ定義ファイル
- ・ サービスIDファイル
- ・ メッセージファイル (V6からの移行時のみ)

資源のバックアップ時はWebサーバ(Interstage HTTP Server)を停止してください。



例

認証サーバの資源をバックアップ用ディレクトリにコピーする例

Windows

バックアップ用ディレクトリ:X:\¥Backup¥ssoatcag

copy/xcopyコマンド(またはエクスプローラ)を使用して、コピーします。

```
copy C:\Interstage\F3FM\soatcag\conf\soatcag.conf X:\Backup\soatcag
copy C:\Interstage\F3FM\soatcag\conf\serviceid X:\Backup\soatcag (注1)
xcopy /E /I C:\Interstage\F3FM\soatcag\pub\template X:\Backup\soatcag\template (注2)
```

Solaris

Linux

バックアップ用ディレクトリ:/backup/FJSVsoac
cpコマンドを使用して、コピーします。

```
cp -p /etc/opt/FJSVsoac/conf/soatcag.conf /backup/FJSVsoac
cp -p /etc/opt/FJSVsoac/conf/serviceid /backup/FJSVsoac (注1)
cp -rp /etc/opt/FJSVsoac/pub/template /backup/FJSVsoac (注2)
```

注1) サービスIDファイルは認証サーバ定義ファイルの“serviceidpath”定義に設定したファイルを指定してください。

注2) V6からの移行時のみ必要な作業です。

Webサーバ(Interstage HTTP Server)の移行については“4.1.1 8.0以前のInterstage HTTP Server(Apache HTTP Server 1.3ベース)からの移行”を参照してください。

なお、Webサーバ(Interstage HTTP Server)にInterstage シングル・サインオン以外のサービスを設定している場合は、設定している各サービスの移行方法に従ってください。

2) SSL通信を行うための環境のバックアップ

認証サーバでSSL通信を行うための環境構築を行っている場合 (V6.0からの移行時)

Interstage証明書環境資源をバックアップします。



Interstage証明書環境資源をバックアップ用ディレクトリにコピーする例

例

Windows

バックアップ用ディレクトリ:X:\Backup\scs
xcopyコマンド(またはエクスプローラ)を使用して、コピーします。

```
xcopy /E C:\Interstage\etc\security X:\Backup\scs
```

Solaris

Linux

バックアップ用ディレクトリ:/backup/scs
cpコマンドを使用して、コピーします。

```
cp -rp /etc/opt/FJSVsscs/security /backup/scs
```

注) Interstage証明書環境は、環境構築時に指定したグループからアクセス可能となっています。そのため、ユーザアカウントやグループ等のシステムの情報についてもバックアップするようにしてください。

認証サーバでSSL通信を行うための環境構築を行っている場合 (V5.1からの移行時)

本バージョン・レベルではSSL環境にInterstage証明書環境を使用します。V5.1のシステムに構築したSSL環境からサイト証明書を抽出し移行後のInterstage証明書環境にサイト証明書を移入してください。(注)

サイト証明書の抽出はV5.1のシステムに構築したSSL環境からcmmkpfxコマンドを使用してPKCS#12形式で取り出します。cmmkpfxコマンドの詳細については、V5.1の“リファレンスマニュアル(コマンド)編”を参照してください。

注) 移行に関してはサイト証明書を発行した認証局のライセンス契約を確認した上で行ってください。



サイト証明書を抽出する例

例

Windows

cmmkpfxコマンドが格納されているディレクトリ:C:\Program Files\Common Files\Fujitsu Shared\F3FSSMEE
PKCS#12ファイル出力先:X:\Backup\server-cert.pfx
運用管理ディレクトリ:C:\sslenv\manage

サイト証明書が存在するトークンのラベル:SSOToken
SSL通信に使用する証明書のニックネーム:SERVERCERT

```
"C:\Program Files\Common Files\Fujitsu Shared\F3SSMEE\cmmkpfx" X:\Backup\server-cert.pfx -  
ed C:\sslnv\manage -tl SSOToken -nn SERVERCERT
```

Solaris

Linux

PKCS#12ファイル出力先:/backup/server-cert.pfx
運用管理ディレクトリ:/sslnv/manage
サイト証明書が存在するトークンのラベル:SSOToken
SSL通信に使用する証明書のニックネーム:SERVERCERT

```
/opt/FJSVsmee/bin/cmmkpfx /backup/server-cert.pfx -ed /sslnv/manage -tl SSOToken -nn  
SERVERCERT
```

User-PINの入力を求められますので入力してください。
PKCS#12データに設定するPassword、および再入力求められます。パスワードを設定してください。

**認証サーバでSSL Accelerator 7117などのSSLアクセラレータ、またはInterstage Security Directorと連携してSSL通信を行うための環境構築を行っている場合、または
認証サーバでSSL通信を行うための環境構築を行っていない場合 (V5.1からの移行時)**

SSL通信を行うための環境をバックアップする必要はありません。

3) パッケージのアンインストールとインストール

現在インストールされているパッケージをアンインストール後、本バージョン・レベルのパッケージをインストールします。

4) SSL通信を行うための環境のリストア、または環境の構築

認証サーバでSSL通信を行うための環境構築を行っている場合 (V6.0からの移行時)

バックアップしたInterstage証明書環境資源をリストアします。



例

バックアップ用ディレクトリのInterstage証明書環境資源をリストアする例

Windows

バックアップ用ディレクトリ:X:\Backup\scs
xcopyコマンド(またはエクスプローラ)を使用して、コピーします。

```
xcopy /E X:\Backup\scs C:\Interstage\etc\security
```

Solaris

Linux

バックアップ用ディレクトリ:/backup/scs
cpコマンドを使用して、コピーします。

```
cp -rp /backup/scs/security /etc/opt/FJSVsscs
```

注) バックアップ前と同じディレクトリ、同じ権限でリストアしてください。また、Interstage証明書環境は、環境構築時に指定したグループからアクセス可能となっています。そのため、必要に応じ、ユーザアカウントやグループ等のシステムの情報についてもリストアするようにしてください。

認証サーバでSSL通信を行うための環境構築を行っている場合 (V5.1からの移行時)

2) SSL通信を行うための環境のバックアップでPKCS#12形式で取り出したサイト証明書をInterstage証明書環境に移入します。サイト証明書を移行する場合、以下の手順で行います。

1. scsimpfxコマンドを使用してInterstage証明書環境にサイト証明書を登録します。
scsimpfxコマンドの詳細については“リファレンスマニュアル(コマンド)編”を参照してください。
なお、サイト証明書登録時には事前にInterstage証明書環境を作成する必要があります。Interstage証明書環境の作成方法につ

いては“セキュリティシステム運用ガイド”の“Interstage証明書環境の構築と利用”―“環境の構築方法”の“PKCS#12データを使用する方法”を参照してください。

2. 証明書認証時にCRL(Certificate Revocation List)による証明書の有効性確認を行う場合には、Interstage証明書環境に認証局より取得したCRLを登録する必要があります。
Interstage証明書環境にCRLを登録する方法については“セキュリティシステム運用ガイド”の“Interstage証明書環境の構築と利用”―“CSRによるInterstage証明書環境の構築方法”―“証明書・CRLの登録”を参照してください。
3. サイト証明書の登録後、Interstage管理コンソールを使用してSSLの定義を作成します。
SSLの定義の設定については“シングル・サインオン運用ガイド”の“環境構築(SSO管理者編)”―“認証サーバの構築”―“SSL通信環境の構築”―“SSL通信を行うための設定”を参照してください。



Interstage証明書環境を構築する実行例

例

Windows

```
scsmakeenv -e
```

Solaris

Linux

Bourneシェルを使用した実行例です。

コマンド実行時には環境変数JAVA_HOMEにJDK、またはJREのインストールパスを設定してください。

Interstage証明書環境の所有グループ:iscertg

```
JAVA_HOME=/opt/FJSVawjbjk/jdk5;export JAVA_HOME
scsmakeenv -e -g iscertg
```

パスワードの入力を求められますので、Interstage証明書環境に設定するパスワードを指定してください。



Interstage証明書環境にサイト証明書を移入する実行例

例

Windows

取り出したPKCS#12ファイル:X:\Backup¥server-cert.pfx

```
scsimppfx -f X:\Backup¥server-cert.pfx
```

Solaris

Linux

取り出したPKCS#12ファイル:/backup/server-cert.pfx

```
scsimppfx -f /backup/server-cert.pfx
```

Interstage証明書環境に設定したパスワードの入力を求められますので、入力してください。

PKCS#12データに設定したパスワードの入力を求められますので入力してください。



Interstage証明書環境にCRLを登録する実行例

例

Windows

取得したCRL:C:\temp¥crl.crl

```
scscenter -c -f C:\temp¥crl.crl
```

Solaris

Linux

取得したCRL:/tmp/crl.crl

```
scscenter -c -f /tmp/crl.crl
```

Interstage証明書環境に設定したパスワードの入力を求められますので、入力してください。

認証サーバでSSL Accelerator 7117などのSSLアクセラレータ、またはInterstage Security Directorと連携してSSL通信を行うための環境構築を行っている場合

SSL Accelerator 7117と連携する場合には、SSL Accelerator 7117の“取扱説明書”を参照してください。また、Interstage Security Directorと連携する場合には、Interstage Security Directorのマニュアルを参照してください。

認証サーバでSSL通信を行うための環境構築を行っていない場合 (V5.1からの移行時)

V5.1の認証サーバでSSL通信を行うための環境構築を行っていない場合、SSL通信環境を構築する必要があります。SSL通信環境には、認証サーバでSSL通信を行う場合とSSL Accelerator 7117などのSSLアクセラレータやInterstage Security Directorなど認証サーバ以外でSSL通信を行う場合があり、それぞれ環境の構築方法が異なります。

SSL通信の環境構築については“シングル・サインオン運用ガイド”の“環境構築(SSO管理者編)”―“認証サーバの構築”―“SSL通信環境の構築”を参照してください。

なお、SSL Accelerator 7117と連携する場合には、SSL Accelerator 7117の“取扱説明書”を参照してください。また、Interstage Security Directorと連携する場合には、Interstage Security Directorのマニュアルを参照してください。



注意

SSL通信環境を構築する場合には、業務サーバから認証サーバにアクセスするためのURLを“http”から“https”に変更する必要があります。Interstage Application Server Standard Edition、Enterprise Edition V5.1の業務サーバを本バージョン・レベルに移行する際に、業務サーバの定義ファイル(ssoatzag.conf)の“AuthServerURL”に認証サーバへアクセスするためのURLを設定してください。

5) 認証サーバ資源のリストア

“1) 認証サーバ資源のバックアップ”でバックアップした認証サーバの資源をリストアします。



例

バックアップ用ディレクトリの認証サーバの資源をリストアする例

Windows

バックアップ用ディレクトリ: X:¥Backup¥ssoatcag

copy/xcopyコマンド(またはエクスプローラ)を使用して、コピーします。

```
copy X:¥Backup¥ssoatcag¥ssoatcag.conf C:¥Interstage¥F3FMso¥ssoatcag¥conf
copy X:¥Backup¥ssoatcag¥serviceid C:¥Interstage¥F3FMso¥ssoatcag¥conf (注1)
xcopy /E /I X:¥Backup¥ssoatcag¥template C:¥Interstage¥F3FMso¥ssoatcag¥pub¥template (注2)
```

Solaris

Linux

バックアップ用ディレクトリ: /backup/FJSVsssoac

cpコマンドを使用して、コピーします。

```
cp -p /backup/FJSVsssoac/ssoatcag.conf /etc/opt/FJSVsssoac/conf
cp -p /backup/FJSVsssoac/serviceid /etc/opt/FJSVsssoac/conf (注1)
cp -rp /backup/FJSVsssoac/template /etc/opt/FJSVsssoac/pub (注2)
```

注1) サービスIDファイルは必ず上記にリストアし、認証サーバ定義ファイルの“serviceidpath”定義に設定してください。

注2) V6からの移行時のみ必要な作業です。

6) 認証サーバ定義ファイルの編集

environment-directory定義が設定されている場合には、定義を削除してください。

また、accesslog-filename定義を以下に変更してください。

Windows

```
accesslog-filename=C:\¥Interstage¥F3FMso¥ssoatcag¥log¥ssoatcag.log
```

Solaris

Linux

```
accesslog-filename=/var/opt/FJSVsoac/log/ssoatcag.log
```

V5.1からの移行時

Interstage Application Server V5.1から移行する場合は、上記に加えて以下の作業を行ってください。

V5.1でのリポジトリサーバの以下の定義は、本バージョン・レベルでは認証サーバで定義します。認証サーバの定義ファイルをリストア後、リポジトリサーバの定義に設定されている以下の定義を認証サーバ定義ファイルに追加して設定してください。(注1)

- password-max-failure-count
- lock-out-time
- certificate-mapping-attribute (注2)
- default-credential-expiration-time
- certificate-identification

注1) リポジトリサーバが複数台で運用されていた場合は、リポジトリサーバ(更新系)の定義を認証サーバ定義ファイルに追加してください。また、認証サーバが複数台で運用されていた場合は、すべての認証サーバの定義ファイルに追加してください。

注2) 本バージョン・レベルでは、本定義名に複数の属性名を設定することはできません。V5.1で、本定義に複数の属性名を設定していた場合には、使用する証明書にあわせて“mail”、“employeeNumber”、“uid”、“serialNumber”、“dnQualifier”、“cn”のいずれか1つを設定してください。

7) Webサーバ(Interstage HTTP Server)定義の編集

Webサーバ(Interstage HTTP Server)の移行作業を実施後、Interstage HTTP Serverの環境定義ファイル(httpd.conf)に設定されている認証サーバの設定を削除し、以下の設定を追加してください。また、V6.0/V5.1の認証サーバでSSL通信を行うための環境構築を行っている場合は、認証サーバで使用しているSSL通信の設定も削除してください。なお、ポート番号は移行前に使用していたポート番号を設定してください。

削除する設定については、旧バージョン・レベルの“シングル・サインオン運用ガイド”の“環境構築”－“認証サーバの環境構築”－“Interstage HTTP Serverの環境定義ファイルの設定”を参照してください。

なお、Interstage HTTP Serverの環境定義の編集に失敗している場合は、[システム]>[セキュリティ]>[シングル・サインオン]>[認証基盤]>[一覧]に表示される認証サーバのポート番号に“不明”と表示されます。編集内容に間違いがないか確認してください。

環境定義ファイルに追加する項目

項目名	内容
Listen	認証サーバが起動するときのネットワークポート番号
LoadModule	認証サーバのプログラム
<VirtualHost _default_:ポート番号> <Location> SetHandler </Location> DocumentRoot </VirtualHost>	認証サーバの実行に必要な項目



Interstage HTTP Serverの環境定義ファイル(httpd.conf)の設定例

例

環境定義ファイルの最終行に以下の記述を追加し、太字部分(ポート番号)を運用に合わせて変更してください。なお、以下は認証サーバを運用するポート番号に“443”を使用する場合を例にしています。

Windows


```

Listen 443
LoadModule ssoatcag_module "C:\Interstage\F3FMso\ssoatcag\lib\F3FMsoatcag.dll"
<VirtualHost _default_:443>
<Location /ssoatcag>
SetHandler ssoatcag-handler
</Location>
DocumentRoot "C:\Interstage\F3FMso\ssoatcag\pub\docroot"
</VirtualHost>

```

Solaris

Linux

```

Listen 443
LoadModule ssoatcag_module "/opt/FJSVssoac/lib/ssoatcag.so"
<VirtualHost _default_:443>
<Location /ssoatcag>
SetHandler ssoatcag-handler
</Location>
DocumentRoot "/opt/FJSVssoac/pub/docroot"
</VirtualHost>

```

8) 認証サーバの環境設定

環境移行後、認証サーバの環境を移行したマシンのInterstage管理コンソールから以下の手順で環境を設定します。各項目の詳細についてはInterstage管理コンソールのヘルプを参照してください。

1. [システム] > [セキュリティ] > [シングル・サインオン] > [認証基盤] > [認証サーバ] > [環境設定]タブ > [詳細設定[表示]]をクリックし、[認証基盤の情報]の[認証基盤のURL]に業務システムからアクセスを受け付ける認証基盤のURLを設定します。認証基盤のURLについては“シングル・サインオン運用ガイド”の“概要”－“URLの決定”－“認証基盤のURLについて”を参照してください。
2. V5.1から移行する場合は、[システム] > [サービス] > [Webサーバ] > [Webサーバ名] > [環境設定]タブより、認証サーバで使用するWebサーバに、[4\) SSL通信を行うための環境のリストア](#)、または[環境の構築](#)で作成したSSL定義を指定してください。認証サーバで使用するWebサーバは、[システム] > [セキュリティ] > [シングル・サインオン] > [認証基盤] > [認証サーバ] > [環境設定]タブ > [詳細設定[表示]]をクリックし、[通信の設定]の[使用しているWebサーバ]より確認できます。
なお、SSL Accelerator 7117などのSSLアクセラレータ、またはInterstage Security Directorと連携し、利用者の証明書が失効されているかの確認を行わない場合にはSSL定義の指定は不要です。
3. 業務サーバを非SSL(http)通信で運用している場合は、[業務システムとの通信の設定]の[HTTP通信]に“許可する”を選択します。
4. SSL Accelerator 7117などのSSLアクセラレータ、またはInterstage Security Directorと連携し、証明書認証を行う場合には[証明書認証の動作]の[ユーザ証明書を獲得するHTTPヘッダ名]にユーザ証明書を獲得するHTTPヘッダ名を指定してください。
5. [適用]ボタンをクリックします。



ポイント

- ・ 業務サーバの保護リソースへのアクセスは、Interstage シングル・サインオンにより制御されますが、業務サーバを非SSL(http)通信で運用している場合、ネットワークから保護リソースが盗聴される危険性があります。業務サーバをSSL(https)通信で運用することにより通信内容を暗号化し、盗聴から守ることができます。業務サーバはSSL(https)通信で運用することを強く推奨します。
また、より安全に運用していただくために、すべての業務サーバをSSL(https)通信で運用している場合には、以下の設定を行うことにより、非SSL(http)通信で運用している業務サーバからのアクセスを禁止できます。

1. [システム] > [セキュリティ] > [シングル・サインオン] > [認証基盤] > [認証サーバ] > [環境設定]タブを選択し、[詳細設定[表示]]をクリックします。

2. [業務システムとの通信の設定]の[HTTP通信]に“許可しない”を選択します。

3. [適用]ボタンをクリックします。

- 本バージョン・レベルでは、セキュリティ上、Webブラウザに表示する一部のメッセージを「ユーザ名、またはパスワードが正しくありません。」に統一しています。今までのメッセージをそのまま表示したい場合は、Interstage管理コンソールを使用して、[システム]>[セキュリティ]>[シングル・サインオン]>[認証基盤]>[認証サーバ]>[環境設定]タブの[詳細設定[表示]]をクリックし、[業務システムとの通信の設定]の[利用者への認証失敗原因の通知]に“通知する”を選択してください。

5.11.6 Interstage Application Server Plus V6.0からのリポジトリサーバ、および認証サーバの移行

リポジトリサーバ、および認証サーバが1台のマシンに構築されている場合の移行手順について説明します。

■移行手順

移行は以下の手順で行います。

1. リポジトリサーバ資源のバックアップ

リポジトリサーバ移行時の手順と同様です。“リポジトリサーバの移行”の“[1\) リポジトリサーバ資源のバックアップ](#)”を参照してください。

2. SSOリポジトリからのデータの抽出

リポジトリサーバ移行時の手順と同様です。“リポジトリサーバの移行”の“[2\) SSOリポジトリからのデータの抽出](#)”を参照してください。

3. 認証サーバ資源のバックアップ

認証サーバ移行時の手順と同様です。“認証サーバの移行”の“[1\) 認証サーバ資源のバックアップ](#)”を参照してください。ただし、Webサーバ(Interstage HTTP Server)の環境定義ファイルのバックアップは、リポジトリサーバ資源のバックアップに含まれるため不要です。

4. SSL通信を行うための環境のバックアップ

認証サーバ移行時の手順と同様です。“認証サーバの移行”の“[2\) SSL通信を行うための環境のバックアップ](#)”を参照してください。

5. パッケージのアンインストールとインストール

現在インストールされているパッケージをアンインストール後、本バージョン・レベルのパッケージをインストールします。

6. SSOリポジトリの作成

リポジトリサーバ移行時の手順と同様です。“リポジトリサーバの移行”の“[4\) SSOリポジトリの作成](#)”を参照してください。

7. SSOリポジトリへのデータの移入

リポジトリサーバ移行時の手順と同様です。“リポジトリサーバの移行”の“[5\) SSOリポジトリへのデータの移入](#)”を参照してください。

8. リポジトリサーバ資源のリストア

リポジトリサーバ移行時の手順と同様です。“リポジトリサーバの移行”の“[9\) リポジトリサーバ資源のリストア](#)”を参照してください。

9. リポジトリサーバ定義ファイルの編集

リポジトリサーバ移行時の手順と同様です。“リポジトリサーバの移行”の“[10\) リポジトリサーバ定義ファイルの編集](#)”を参照してください。

10. SSL通信を行うための環境のリストア、または環境の構築

認証サーバ移行時の手順と同様です。“認証サーバの移行”の“[4\) SSL通信を行うための環境のリストア、または環境の構築](#)”を参照してください。

11. 認証サーバ資源のリストア

認証サーバ移行時の手順と同様です。“認証サーバの移行”の“[5\) 認証サーバ資源のリストア](#)”を参照してください。ただし、

Webサーバ(Interstage HTTP Server)の環境定義ファイルのリストアは、リポジトリサーバ資源のリストアに含まれるため不要です。

12. 認証サーバ定義ファイルの編集

認証サーバ移行時の手順と同様です。“認証サーバの移行”の“[6\) 認証サーバ定義ファイルの編集](#)”を参照してください。

13. Webサーバ(Interstage HTTP Server)定義の編集

Webサーバ(Interstage HTTP Server)定義を編集します。詳細は、“[◆Webサーバ\(Interstage HTTP Server\)定義の編集](#)”を参照してください。

14. リポジトリサーバの環境設定

リポジトリサーバ移行時の手順と同様です。“リポジトリサーバの移行”の“[12\) リポジトリサーバの環境設定](#)”を参照してください。

15. 認証サーバの環境設定

認証サーバ移行時の手順と同様です。“認証サーバの移行”の“[8\) 認証サーバの環境設定](#)”を参照してください。

◆Webサーバ(Interstage HTTP Server)定義の編集

Webサーバ(Interstage HTTP Server)の移行作業を実施後、Interstage HTTP Serverの環境定義ファイル(httpd.conf)に設定されているリポジトリサーバ、および認証サーバの設定を削除し、以下の設定を追加してください。なお、ポート番号は移行前に使用していたポート番号を設定してください。

削除する設定については、旧バージョン・レベルの“シングル・サインオン運用ガイド”の“環境構築”－“認証サーバの環境構築”－“Interstage HTTP Serverの環境定義ファイルの設定”を参照してください。

なお、Interstage HTTP Serverの環境定義の編集に失敗している場合は、[システム]>[セキュリティ]>[シングル・サインオン]>[認証基盤]>[一覧]に表示されるリポジトリサーバ、または認証サーバのポート番号に“不明”と表示されます。編集内容に間違いがないか確認してください。

環境定義ファイルに追加する項目

項目名	内容
Listen	リポジトリサーバおよび認証サーバが起動するときのネットワークポート番号
LoadModule	リポジトリサーバおよび認証サーバのプログラム
<VirtualHost _default_:ポート番号> <Location> SetHandler </Location> DocumentRoot </VirtualHost>	リポジトリサーバおよび認証サーバの実行に必要な項目



Interstage HTTP Serverの環境定義ファイル(httpd.conf)の設定例

例

環境定義ファイルの最終行に以下の記述を追加し、太字部分(ポート番号)を運用に合わせて変更してください。なお、以下はリポジトリサーバを運用するポート番号に“**10550**”、認証サーバを運用するポート番号に“**443**”を使用する場合を例にしています。

Windows

```
Listen 10550
LoadModule ssoatcsv_module "C:\¥Interstage¥F3FMssso¥ssoatcsv¥lib¥F3FMsssoatcsv.dll"
<VirtualHost _default_:10550>
<Location /ssoatcsv>
SetHandler ssoatcsv-authenticate-handler
</Location>
<Location /resinfo>
```

```

SetHandler ssoatcsv-resourceinfo-handler
</Location>
<Location /roleref>
SetHandler ssoatcsv-rolerefresh-handler
</Location>
DocumentRoot "C:\Interstage\F3FM\so\ssoatcsv\pub\docroot"
</VirtualHost>

Listen 443
LoadModule ssoatcag_module "C:\Interstage\F3FM\so\ssoatcag\lib\F3FM\soatcag.dll"
<VirtualHost _default_:443>
<Location /ssoatcag>
SetHandler ssoatcag-handler
</Location>
DocumentRoot "C:\Interstage\F3FM\so\ssoatcag\pub\docroot"
</VirtualHost>

```

Solaris

Linux

```

Listen 10550
LoadModule ssoatcsv_module "/opt/FJSVssosv/lib/ssoatcsv.so"
<VirtualHost _default_:10550>
<Location /ssoatcsv>
SetHandler ssoatcsv-authenticate-handler
</Location>
<Location /resinfo>
SetHandler ssoatcsv-resourceinfo-handler
</Location>
<Location /roleref>
SetHandler ssoatcsv-rolerefresh-handler
</Location>
DocumentRoot "/opt/FJSVssosv/pub/docroot"
</VirtualHost>

Listen 443
LoadModule ssoatcag_module "/opt/FJSVssoc/lib/ssoatcag.so"
<VirtualHost _default_:443>
<Location /ssoatcag>
SetHandler ssoatcag-handler
</Location>
DocumentRoot "/opt/FJSVssoc/pub/docroot"
</VirtualHost>

```

5.11.7 Interstage Application Server Plus、Web-J Edition V6.0からの業務サーバの移行

業務サーバの移行手順について説明します。

負荷分散のために、業務サーバを複数構築して運用している場合は、同時に移行してバージョン・レベルを合わせてください。

■移行手順

移行は以下の手順で行います。

1. 業務サーバ資源のバックアップ
2. SSL通信を行うための環境のバックアップ
3. パッケージのアンインストールとインストール

4. SSL通信を行うための環境のリストア、または環境の構築
5. 業務サーバ資源のリストア
6. 業務サーバ定義ファイルの編集
7. 業務サーバの環境設定
8. Webサーバの設定
9. Interstage証明書環境の作成



- ・ 業務サーバの保護リソースへのアクセスは、Interstage シングル・サインオンにより制御されますが、業務サーバを非SSL(http)通信で運用している場合、ネットワークから保護リソースが盗聴される危険性があります。業務サーバをSSL(https)通信で運用することにより通信内容を暗号化し、盗聴から守ることができます。業務サーバは、SSL(https)通信で運用することを強く推奨します。非SSL(http)通信で運用しているV6.0以前の業務サーバを本バージョン・レベルに移行する場合には、認証サーバの定義ファイル(ssoatcag.conf)の“allow-redirect-over-http”に“YES”を設定してください。
- ・ 本バージョン・レベルは、Windows NT(R) Server 4.0をサポートしていないため、V6.0以前でMicrosoft(R) Internet Information Server 4.0を使用している場合は、別のWebサーバを使用してください。
- ・ 本バージョン・レベルは、Solaris 8をサポートしていないため、V6.0以前でSun Java System Web Server 4.1を使用している場合は、別のWebサーバを使用してください。
- ・ 本バージョン・レベルでは、InfoProvider Proをサポートしていないため、V6.0以前でInfoProvider Proを使用している場合は、別のWebサーバを使用してください。

1) 業務サーバ資源のバックアップ

以下の業務サーバの資源をバックアップします。

- ・ 業務サーバ定義ファイル
- ・ サービスIDファイル
- ・ アクセス制御情報ファイル
- ・ メッセージファイル (V6からの移行時のみ)

資源のバックアップ時は業務サーバに使用しているWebサーバを停止してください。



業務サーバの資源をバックアップ用ディレクトリにコピーする例

例

Windows

バックアップ用ディレクトリ:X:\Backup¥ssoatcag

copy/xcopyコマンド(またはエクスプローラ)を使用して、コピーします。

```
copy C:\¥Interstage¥F3FMso¥ssoatcag¥conf¥ssoatcag.conf X:\¥Backup¥ssoatcag
copy C:\¥Interstage¥F3FMso¥ssoatcag¥conf¥domainsid X:\¥Backup¥ssoatcag (注1)
copy C:\¥Interstage¥F3FMso¥ssoatcag¥conf¥ssoacscrl X:\¥Backup¥ssoatcag (注2)
xcopy /E /I C:\¥Interstage¥F3FMso¥ssoatcag¥pub¥template X:\¥Backup¥ssoatcag¥template (注3)
```

Solaris

Linux

バックアップ用ディレクトリ:/backup/FJSVssoaz

cpコマンドを使用して、コピーします。

```
cp -p /etc/opt/FJSVssoaz/conf/ssoatcag.conf /backup/FJSVssoaz
cp -p /etc/opt/FJSVssoaz/conf/domainsid /backup/FJSVssoaz (注1)
```

```
cp -p /etc/opt/FJSVssoaz/conf/ssoacscctl/backup/FJSVssoaz (注2)
cp -rp /etc/opt/FJSVssoaz/pub/template/backup/FJSVssoaz (注3)
```

- 注1) サービスIDファイルは業務サーバ定義ファイルの“ServiceIDPath”定義に設定したファイルを指定してください。
注2) アクセス制御情報ファイルは業務サーバ定義ファイルの“AccessCtl”定義に設定したファイルを指定してください。
注3) V6からの移行時のみ必要な作業です。

WebサーバにInterstage HTTP Serverを使用している場合は、Webサーバ(Interstage HTTP Server)の移行が必要です。Webサーバ(Interstage HTTP Server)の移行については“4.1.1 8.0以前のInterstage HTTP Server (Apache HTTP Server 1.3ベース)からの移行”を参照してください。

なお、Webサーバ(Interstage HTTP Server)にInterstage シングル・サインオン以外のサービスを設定している場合は、設定している各サービスの移行方法に従ってください。

2) SSL通信を行うための環境のバックアップ

業務サーバでSSL通信を行うための環境構築を行っている場合は、SSL通信を行うための環境をバックアップします。

WebサーバにInterstage HTTP Serverを使用している場合は、“4.1.1 8.0以前のInterstage HTTP Server (Apache HTTP Server 1.3ベース)からの移行”を参照してください。

3) パッケージのアンインストールとインストール

現在インストールされているパッケージをアンインストール後、本バージョン・レベルのパッケージをインストールします。

4) SSL通信を行うための環境のリストア、または環境の構築

バックアップしたSSL通信を行うための環境をリストアします。

WebサーバにInterstage HTTP Serverを使用している場合は、“4.1.1 8.0以前のInterstage HTTP Server (Apache HTTP Server 1.3ベース)からの移行”を参照してください。

5) 業務サーバ資源のリストア

“1) 業務サーバ資源のバックアップ”でバックアップした業務サーバの資源をリストアします。



バックアップ用ディレクトリの業務サーバの資源をリストアする例

例

Windows

バックアップ用ディレクトリ:X:\¥Backup¥ssoatzag
copy/xcopyコマンド(またはエクスプローラ)を使用して、コピーします。

```
copy X:\¥Backup¥ssoatzag¥ssoatzag.conf C:\¥Interstage¥F3FM¥ssoatzag¥conf
copy X:\¥Backup¥ssoatzag¥domainsid C:\¥Interstage¥F3FM¥ssoatzag¥conf (注1)
copy X:\¥Backup¥ssoatzag¥ssoacscctl C:\¥Interstage¥F3FM¥ssoatzag¥conf (注2)
xcopy /E /I X:\¥Backup¥ssoatzag¥template C:\¥Interstage¥F3FM¥ssoatzag¥pub¥template (注3)
```

Solaris

Linux

バックアップ用ディレクトリ:/backup/FJSVssoaz
cpコマンドを使用して、コピーします。

```
cp -p /backup/FJSVssoaz/ssoatzag.conf /etc/opt/FJSVssoaz/conf
cp -p /backup/FJSVssoaz/domainsid /etc/opt/FJSVssoaz/conf (注1)
cp -p /backup/FJSVssoaz/ssoacscctl /etc/opt/FJSVssoaz/conf (注2)
cp -rp /backup/FJSVssoaz/template /etc/opt/FJSVssoaz/pub (注3)
```

- 注1) サービスIDファイルは必ず上記にリストアし、業務サーバ定義ファイルの“ServiceIDPath”定義に設定してください。
注2) アクセス制御情報ファイルは必ず上記にリストアし、業務サーバ定義ファイルの“AccessCtl”定義に設定してください。
注3) V6からの移行時のみ必要な作業です。

6) 業務サーバ定義ファイルの編集

業務サーバ定義ファイルをリストア後、定義ファイルに以下の定義項目を追加、および変更してください。なお、1台のマシンに業務サーバが複数構築されている場合は、業務サーバ定義ファイルに設定されている業務サーバごとに定義を追加してください。



ポイント

業務サーバ定義ファイル内の業務サーバの定義は、ServerPort定義が先頭に定義されています。以下の各定義は、該当する業務サーバのServerPort定義の後に設定してください。

Windows

- business-system-name
- web-server-name

Solaris

Linux

- business-system-name
- web-server-name
- web-user-name

定義項目	追加/変更	説明
business-system-name	追加	業務システム名(業務システムの名称)を設定します。 • 業務システム名は1～32文字までの文字列を設定します。下記の文字を使用してください。下記以外の文字を使用した場合は、業務サーバの環境が壊れていると扱われます。 また、本定義が未設定の場合は、業務システム名はポート番号(業務サーバ定義ファイルのServerPort定義)の値となります。 — 英数字 — ハイフン (-) — アンダースコア (_) — 左括弧 (() — 右括弧 ()) — 左角括弧 ([) — 右角括弧 (]) • 業務サーバを複数設定している場合、各業務サーバのbusiness-system-nameの設定値を一意に設定してください。 一意ではない場合には、ポート番号(業務サーバ定義ファイルのServerPort定義)の値が最も小さい業務サーバの定義のみが有効となり、その他の業務サーバの定義は無視されます。
web-server-name	追加	業務サーバが動作しているWebサーバの種類を設定します。以下のいずれかの値を設定します。 • WebサーバにInterstage HTTP Serverを使用している場合 設定値:IHS • WebサーバにInterstage HTTP Server以外を使用している場合 設定値:OTHER
Solaris Linux	追加	Webサーバが動作する実効ユーザ名を設定します。設定するユーザ名については以下を参照してください。

定義項目	追加/変更	説明
web-user-name		- Interstage HTTP Server 環境定義ファイル(httpd.conf)の“User”に設定されたユーザIDを参照してください。デフォルトは“nobody”です。 - Sun Java System Web Server 4.1、またはSun Java System Web Server 6.0 環境定義ファイル(“Sun Java System Web Serverのインストールパス/https-INSTANCE_NAME (注)/config/obj.conf”)の“User”に設定されたユーザアカウントを参照してください。デフォルトは“nobody”です。 注) https-INSTANCE_NAMEとは、ユーザがサーバとして設定したマシン名です。マシン名が www.fujitsu.com の場合、https-www.fujitsu.comとなります。
FQDN	変更	業務システムが利用者に公開するURLとして、プロトコル(スキーム)およびポート番号を必ず以下の形式(太字部分を追加)で設定してください。すでに以下の形式で設定されている場合は、変更する必要はありません。 http://FQDN:ポート番号 https://FQDN:ポート番号 なお、業務システムが利用者に公開するURLは、ロードバランサなど、他の装置、製品との組み合わせにより設定方法が異なります。設定方法の詳細については、“シングル・サインオン運用ガイド”の“概要”－“URLの決定”－“業務システムの公開URLについて”を参照してください。
accesslog-filename	変更	アクセスログの出力先ファイル名を以下のように設定してください。 Windows C:\Interstage\F3FMssso\ssoatzag\log\ssoatzag443.log Solaris Linux /var/opt/FJSVsssoaz/log/ssoatzag443.log ファイル名についている“443”には、業務サーバのポート番号を設定します。マルチポート運用の場合は、すでに存在するファイル名と重複しないよう注意してください。

また、リストアした業務サーバ定義ファイルに以下の**定義項目**が、異なる内容で設定されている場合には、設定内容を“変更前の設定”から“変更後の設定”に変更してください。

定義項目

- CredentialDN
- CredentialUID
- CredentialROLELIST
- CredentialROLECOUNT
- CredentialIPADDRESS
- CredentialAUTHMETHOD
- CredentialFIRSTACCESS
- CredentialEXPIRATION
- CredentialDOMAIN

設定内容

各定義項目の変更前の設定	各定義項目の変更後の設定
すべて“YES”の場合	変更する必要はありません。
すべて“NO”の場合	変更する必要はありません。
“YES”と“NO”が混在している場合	すべて“YES”に変更してください。
“YES”と省略値(注)が混在している場合	すべて“YES”に変更してください。
“NO”と省略値(注)が混在している場合	すべて“NO”に変更してください。
上記の定義項目のうち、設定されていない定義項目が存在する場合	設定されていない定義項目を追加してください。 追加した定義項目は、設定済みの定義項目の内容によって、以下のように設定してください。 ・すべて“YES”の場合は“YES” ・すべて“NO”の場合は“NO” ・“YES”と“NO”が混在している場合は“YES” ・“YES”と省略値(注)が混在している場合は“YES” ・“NO”と省略値(注)が混在している場合は“NO”

注) 省略値:設定値がない、または“YES”や“NO”以外が設定されている



例

業務サーバ定義ファイルの編集例

以下は、業務システム名に“**Business001**”、使用しているWebサーバに“**IHS**”を設定しています。

また、V6.0/V5.1の業務サーバをSSL(**https**)通信で運用し、業務サーバのFQDNが“**www.fujitsu.com**”、ポート番号に“**443**”を使用している場合を例にしています。CredentialDN、CredentialUID、CredentialROLELIST、CredentialROLECOUNT、CredentialIPADDRESS、CredentialAUTHMETHOD、CredentialFIRSTACCESS、CredentialEXPIRATION、CredentialDOMAINについてはすべて“**YES**”に変更しています。

太字部分は運用に合わせて変更してください。

Windows

アクセスログの出力先ファイル:C:\¥Interstage¥F3FMso¥ssoatzag¥log¥ssoatzag443.log

```
business-system-name=Business001
web-server-name=IHS
FQDN=https://www.fujitsu.com:443
accesslog-filename=C:\¥Interstage¥F3FMso¥ssoatzag¥log¥ssoatzag443.log
CredentialDN=YES
CredentialUID=YES
CredentialROLELIST=YES
CredentialROLECOUNT=YES
CredentialIPADDRESS=YES
CredentialAUTHMETHOD=YES
CredentialFIRSTACCESS=YES
CredentialEXPIRATION=YES
CredentialDOMAIN=YES
```

Solaris

Linux

Webサーバが動作する実効ユーザ名:nobody

アクセスログの出力先ファイル:/var/opt/FJSVsssoaz/log/ssoatzag443.log

```
business-system-name=Business001
web-server-name=IHS
web-user-name=nobody
FQDN=https://www.fujitsu.com:443
```



```

accesslog-filename=/var/opt/FJSVssoaz/log/ssoatzag443.log
CredentialDN=YES
CredentialUID=YES
CredentialROLELIST=YES
CredentialROLECOUNT=YES
CredentialIPADDRESS=YES
CredentialAUTHMETHOD=YES
CredentialFIRSTACCESS=YES
CredentialEXPIRATION=YES
CredentialDOMAIN=YES

```



ポイント

V5.1からの移行において、業務サーバがアクセスする認証サーバの環境を、SSL通信を行うように変更した場合には、業務サーバの定義ファイル(ssoatzag.conf)の“AuthServerURL”の設定を「認証基盤のURL + “/ssoatzag”」の形式に変更してください。

なお、認証基盤のURLについては、アクセスする認証基盤をInterstage管理コンソールで環境定義している場合には、Interstage管理コンソールの[システム] > [セキュリティ] > [シングル・サインオン] > [認証基盤] > [リポジトリサーバ] > [環境設定]タブを選択して、[リポジトリサーバ詳細設定[表示]]をクリックし、[認証基盤の情報]の[認証基盤のURL]を参照してください。

7) 業務サーバの環境設定

環境移行後、業務サーバの環境を移行したマシンのInterstage管理コンソールから以下の手順で環境を設定します。各項目の詳細についてはInterstage管理コンソールのヘルプを参照してください。

1. [システム] > [セキュリティ] > [シングル・サインオン] > [業務システム] > [業務システム名] > [環境設定]タブ > [詳細設定[表示]]をクリックし、[認証基盤の情報]の[リポジトリサーバのURL]にリポジトリサーバのURLを設定します。
2. [適用]ボタンをクリックします。

前項の“6) 業務サーバ定義ファイルの編集”で設定した値が正しく設定されているか確認してください。設定する各項目と環境定義ファイルの定義項目との対応表を以下に示します。

定義項目	Interstage管理コンソールの設定
business-system-name	[詳細設定[表示]] > [業務システムの情報] > [業務システム名]
web-server-name	[詳細設定[表示]] > [Webサーバの設定] > [使用しているWebサーバ]
Solaris Linux web-user-name	[詳細設定[表示]] > [Webサーバの設定] > [使用しているWebサーバ] > [実効ユーザ名] (注1)
accesslog-filename	[アクセスログ] > [ファイル名]
CredentialDN	[詳細設定[表示]] > [Webアプリケーションとの連携] > [ユーザ情報の通知] (注2)
CredentialUID	
CredentialROLELIST	
CredentialROLECOUNT	
CredentialIPADDRESS	
CredentialAUTHMETHOD	
CredentialFIRSTACCESS	
CredentialEXPIRATION	
CredentialDOMAIN	

注1) web-server-nameにOTHERを設定した場合のみ表示されます。

注2) 定義項目を“YES”と設定した場合には“通知する”、“NO”と設定した場合には“通知しない”と表示されます。

8) Webサーバの設定

Solaris

WebサーバにSun Java System Web Server 4.1.6.0を使用している場合は、Sun Java System Web Serverの環境定義ファイルを編集してください。環境定義ファイルの編集方法については、“シングル・サインオン運用ガイド”の“環境構築(業務サーバ管理者編)”―“Webサーバへの組み込み”を参照してください。

9) Interstage証明書環境の作成

Interstage証明書環境が作成されていない場合は、Interstage証明書環境を作成してください。Interstage証明書環境の作成については“シングル・サインオン運用ガイド”の“SSL通信で運用するための準備”―“Interstage証明書環境の作成”を参照してください。

5.11.8 Interstage Application Server V5.1からのサーバの移行

以下の製品からのサーバの移行方法を説明します。

- Interstage Application Server V5.1

■移行手順

Interstage Application Server Plus、Web-J Edition V6.0からの各サーバの移行手順と同様です。シングル・サインオンのシステムを構成する各サーバの移行手順については以下を参照してください。

- リポジトリサーバを移行する場合
[Interstage Application Server Plus V6.0からのリポジトリサーバの移行](#)
- 認証サーバを移行する場合
[Interstage Application Server Plus V6.0からの認証サーバの移行](#)
- 1台のマシンに構築されているリポジトリサーバ、および認証サーバを移行する場合
[Interstage Application Server Plus V6.0からのリポジトリサーバ、および認証サーバの移行](#)
- 業務サーバを移行する場合
[Interstage Application Server Plus、Web-J Edition V6.0からの業務サーバの移行](#)



注意

- リポジトリサーバの以下の定義は、本バージョン・レベルでは認証サーバで定義します。必ず、認証サーバ定義ファイルに設定してください。認証サーバでの定義方法については、“[5.11.5 Interstage Application Server Plus V6.0からの認証サーバの移行](#)”を参照してください。
 - password-max-failure-count
 - lock-out-time
 - certificate-mapping-attribute
 - default-credential-expiration-time
 - certificate-identification
- 本バージョン・レベルではネットワークから利用者のパスワードが盗聴される危険性に対し、より安全な運用を行うために認証サーバは必ずSSL通信を行う必要があります。V5.1で認証サーバをSSL通信で運用していなかった場合は、移行時にSSL通信の環境を構築してください。

5.11.9 Javaアプリケーションの移行

Javaアプリケーションの移行について、以下を説明します。業務サーバの定義を使用する(ログイン構成ファイルにserverportオプションを指定する)アプリケーションを移行する場合には、“[5.11.2 Interstage Application Server V7.0/8.0からのサーバの移行](#)”の業務サーバの移行、または“[5.11.7 Interstage Application Server Plus、Web-J Edition V6.0からの業務サーバの移行](#)”も行ってください。

- [移行手順](#)

- ・ アプリケーションごとの移行作業
- ・ V5.1のマニュアルから削除された事項

■移行手順

移行は以下の手順で行います。

1. Javaアプリケーション資源のバックアップ
2. パッケージのアンインストールとインストール
3. Javaアプリケーション資源のリストア
4. アプリケーションの再配備

1) Javaアプリケーション資源のバックアップ

以下のJavaアプリケーション資源をバックアップします。

- ・ Javaソースファイル
- ・ Javaクラスファイル
- ・ Javaアプリケーション
 - Java Archive(jar)ファイル
 - Web Archive(war)ファイル
 - Enterprise Archive(ear)ファイル
- ・ ログイン構成ファイル
- ・ トラストストアファイル
- ・ セキュリティポリシーファイル
- ・ サービスIDファイル
- ・ その他のファイル(起動シェルスクリプトや、Javaアプリケーションが使うデータファイルなど)



Javaアプリケーション資源をバックアップ用ディレクトリにコピーする例

例

Windows

バックアップ用ディレクトリ: X:\Backup¥ssoatzag¥javaapi

Javaアプリケーション資源の格納ディレクトリ: C:\¥Interstage¥F3FMssso¥ssoatzag¥sample¥javaapi

xcopy コマンド(またはエクスプローラ)を使用して、コピーします。

```
xcopy C:\¥Interstage¥F3FMssso¥ssoatzag¥sample¥javaapi X:\¥Backup¥ssoatzag¥javaapi /E
```

Solaris

Linux

バックアップ用ディレクトリ: /backup/FJSVsssoaz/javaapi

Javaアプリケーション資源の格納ディレクトリ: /home/jaas/sample/javaapi

cp コマンドを使用して、コピーします。

```
cp -pr /home/jaas/sample/javaapi/* /backup/FJSVsssoaz/javaapi
```

2) パッケージのアンインストールとインストール

現在インストールされているパッケージをアンインストール後、本バージョン・レベルのパッケージをインストールします。

3) Javaアプリケーション資源のリストア

“1) Javaアプリケーション資源のバックアップ”でバックアップしたJavaアプリケーション資源をリストアします。



例

バックアップ用ディレクトリのJavaアプリケーション資源をリストアする例

Windows

バックアップ用ディレクトリ:X:\Backup¥ssoatzag¥javaapi

Javaアプリケーション資源の格納ディレクトリ:C:\Interstage¥F3FMso¥ssoatzag¥sample¥javaapi

xcopyコマンド(またはエクスプローラ)を使用して、コピーします。

```
xcopy X:\Backup¥ssoatzag¥javaapi C:\Interstage¥F3FMso¥ssoatzag¥sample¥javaapi /E
```

Solaris

Linux

バックアップ用ディレクトリ:/backup/FJSVsoaz/javaapi

Javaアプリケーション資源の格納ディレクトリ:/home/jaatest/sample/javaapi

cpコマンドを使用して、コピーします。

```
cp -pr /backup/FJSVsoaz/javaapi/* /home/jaatest/sample/javaapi
```



ポイント

本バージョン・レベルではネットワークから利用者のパスワードが盗聴される危険性に対し、より安全な運用を行うために認証サーバは必ずSSL通信を行う必要があります。

V5.1において、クライアントからユーザID/パスワードを受け取り認証するJavaアプリケーションを認証サーバとSSL通信を行わない設定で運用していた場合には、“[クライアントからユーザID/パスワードを受け取り認証するJavaアプリケーションで認証サーバとSSL通信を行わない設定の移行](#)”を参照してJavaアプリケーションを移行してください。

4) アプリケーションの再配備

アプリケーションがJ2EEアプリケーションの場合で本バージョン・レベルのIJServerで実行するためには、アプリケーションを再配備してください。詳細は“[6.2 J2EEアプリケーションの移行](#)”を参照してください。

クラスパス、JavaVMオプションおよびJavaVMのバージョンの指定は、Interstage管理コンソールの[システム] > [ワークユニット] > [IJServer] > [環境設定]タブを選択して[ワークユニット]の設定で行います。

■アプリケーションごとの移行作業

以下のアプリケーションを移行する場合は、“[移行手順](#)”に加えて以下の作業を行ってください。

ISAuthorizationCredentialオブジェクトのgetExtraDataメソッドでユーザ属性を取得するアプリケーションの移行

getExtraDataメソッドでユーザ属性を取得するアプリケーションを移行する時には、リポジトリサーバの定義ファイルを用いて、リポジトリサーバを移行してください。

ISAuthorizationCredentialオブジェクトから利用者情報を取得するアプリケーションの移行

ISAuthorizationCredentialオブジェクトから利用者情報を取得するアプリケーションを移行する時には、Interstage管理コンソールの[セキュリティ] > [シングル・サインオン] > [業務システム] > [業務システム名] > [環境設定]タブの[詳細設定[表示]]をクリックし、[Webアプリケーションとの連携]の[ユーザ情報の通知]の設定を“通知する”に変更して[適用]ボタンをクリックしてください。

クライアントからユーザID/パスワードを受け取り認証するJavaアプリケーションで認証サーバとSSL通信を行わない設定の移行

本バージョン・レベルでは認証サーバはSSL環境で運用されます。以下の手順でJavaアプリケーションを移行してください。

1. ログイン構成ファイルのauthserverオプションで指定するURLは、アクセスする認証基盤のURLを参照して変更してください。なお、指定する形式は「認証基盤のURL + “/ssoatzag”」です。アクセスする認証基盤をInterstage管理コンソールで環境定義している場合には、Interstage管理コンソールの[システム] > [セキュリティ] > [シングル・サインオン] > [認証基盤] > [リポジトリサーバ] > [環境設定]タブを選択して、[リポジトリサーバ詳細設定[表示]]をクリックし、[認証基盤の情報]の[認証基盤のURL]に設定している認証基盤のURLを参照してください。
2. 認証サーバのサイト証明書、およびその証明書の発行者である認証局証明書を取得してトラストストアファイルを作成します。

3. JavaVM起動時にシステムプロパティ`javax.net.ssl.trustStore`で作成したトラストストアファイルを指定します。

■V5.1のマニュアルから削除された事項

本バージョン・レベルのマニュアルからは以下の事項についてのマニュアル記事が削除されました。説明については、V5.1の“シングル・サインオン運用ガイド”―“アプリケーションの開発”―“Javaアプリケーションの開発”の以下を参照してください。

マニュアル削除事項	旧マニュアル参照箇所
ログインモジュール <code>com.fujitsu.interstage.sso.auth.module.ISCredentialLoginModule</code> の <code>serviceidpath</code> オプション	“アプリケーション実行環境の設定”―“ログイン構成ファイルの作成”
<code>com.fujitsu.interstage.sso.auth.ISAuthenticationCredential</code>	“プログラムの開発”―“利用者情報の取得”
クライアントからユーザID/パスワードを受け取り認証するJavaアプリケーションで認証サーバとSSL通信を行わない設定 (注)	“アプリケーション実行環境の設定”―“ログイン構成ファイルの作成”

(注) 本バージョン・レベルの認証サーバは、SSL通信のみサポートしているため、SSL通信を行わない設定はできません。

第6章 J2EEの移行

ここでは、以下について説明します。

- [J2EEの追加機能](#)
Interstage Application Server V7.0以前から移行する場合に参照してください。
- [J2EEアプリケーションの移行](#)
Interstage Application Server 8.0以前から移行する場合に参照してください。
- [INTERSTAGE V3.xからJ2EEアプリケーションへの移行](#)
INTERSTAGE Application Server V3.x以前から移行する場合に参照してください。
- [Servletサービスの移行](#)
Interstage Application Serverの旧バージョン・レベルから移行する場合に参照してください。
- [EJBサービスの移行](#)
Interstage Application Server 8.0以前から移行する場合に参照してください。
- [Interstage JMSの移行](#)
Interstage Application Server 8.0以前から移行する場合に参照してください。
- [Interstage Webサービスの移行](#)
Interstage Application Server 8.0以前から移行する場合に参照してください。

6.1 J2EEの追加機能

J2EEの以下の追加機能について説明します。

■Interstage Application Server 8.0での追加機能

- [J2EE定義コマンド](#)
- [Webサービス機能](#)
- [アプリケーションファイル保護レベル](#)
- [Javaヒープ/Java Permanent領域不足時の制御](#)

6.1.1 J2EE定義コマンド

Interstage Application Server 8.0(Interstage 8.0)から、J2EEのすべての定義更新処理がコマンドで実行できるisj2eeadminコマンドが提供されました。

従来のInterstage管理コンソールでの操作や以下のコマンドも提供されますが、サーバパッケージでは、以下のコマンドは互換機能(Interstage 8.0以降の機能エンハンスは行いません)となりますので、本機能(isj2eeadminコマンド)への移行を推奨します。

クライアントパッケージではInterstage JMXサービスが存在しないため、従来通り以下のコマンドを使用してください。

- jmsrmfact
- jmsmkdst

また、従来クライアントパッケージにfjj2eeadminコマンドを提供していましたが、クライアントからのデータベース接続は非推奨です。IJSERVER上のJ2EEアプリケーションからデータベース操作を行ってください。

isj2eeadminコマンドの操作範囲を、以下の表に示します。

操作	V7.0以前	8.0以降
IJServer作成	なし	isj2eeadmin
ログ定義	なし	isj2eeadmin
実行クラス	なし	isj2eeadmin
IJServer定義更新	なし	isj2eeadmin
IJServer削除	なし	isj2eeadmin
配備	ijsdeployment	ijsdeployment
配備解除	ijsundeployment	ijsundeployment
J2EEプロパティ定義	定義ファイル公開	isj2eeadmin
Servletサービス設定	なし	isj2eeadmin
EJBサービス設定	定義ファイル公開	isj2eeadmin
Webサーバコネクタ定義	なし	isj2eeadmin
JDBCリソース作成・削除	fjj2eeadmin	isj2eeadmin
JMS ConnectionFactory作成	jmsmkfact	isj2eeadmin
JMS ConnectionFactory削除	jmsrmfact	isj2eeadmin
JMS Destination作成	jmsmkdst	isj2eeadmin
JMS Destination削除	jmsrmdst	isj2eeadmin
connectorリソース定義更新	fjj2eeadmin	isj2eeadmin
JavaMailリソース作成・削除	fjj2eeadmin	isj2eeadmin
EJBアプリケーション環境設定	ejbdefexport/ejbdefimport	ejbdefexport/ejbdefimport
名前変換定義	ijsdeployment	ijsdeployment

isj2eeadminコマンドについては、“リファレンスマニュアル(コマンド編)”の“J2EE運用コマンド”の“isj2eeadmin”を参照してください。

6.1.2 Webサービス機能

Interstage Application Server 8.0から、J2EEの機能の1つとして WebService for J2EE規約に準拠したWebサービスのWARファイルおよびクライアントが提供されました。本機能は、旧バージョン・レベルのSOAPサービスと異なる、新規のWebサービス実行環境を提供しており、機能差異があります。

また、本機能とあわせて、旧バージョン・レベルのSOAPサービスのアプリケーション実行環境も、旧バージョン・レベル通りSOAPサービスとして提供しています。旧バージョン・レベルのSOAPサービスを利用していた場合は、“[8.3 SOAPサービスの移行](#)”を参照してください。

6.1.3 アプリケーションファイル保護レベル

Interstage Application Server 8.0から、アプリケーションファイルの権限を変更する機能が提供されました。

アプリケーションファイルのアクセス権は、IJServer定義のアプリケーションファイル保護レベルの設定に応じて、下記のとおりを設定されます。

アプリケーションファイル保護レベル	アプリケーションファイルのアクセス権	
	ディレクトリ	ファイル
高(管理者のみ)	755	644
低(一般ユーザ)	777	666

アクセス権の変更対象となる資源は、下記のとおりです。

- IJServerディレクトリ/appsディレクトリ、および配下の資源

- IJServerディレクトリ/Shared/libディレクトリ
- IJServerディレクトリ/Shared/classesディレクトリ
- IJServerディレクトリ/extディレクトリ

デフォルトの設定では、“高(管理者のみ)”が設定されています。

これにより、従来は一般ユーザでアクセス可能であったejbdefexport、ejbdefimportコマンドで参照するファイルのアクセス権が、デフォルトの設定では管理者でのみアクセス可能な状態となります。

必要に応じてアプリケーションファイル保護レベルの設定を変更してください。

6.1.4 Javaヒープ/Java Permanent領域不足時の制御

以下の製品から、Javaヒープ/Java Permanent領域不足時の制御を変更する機能が提供されました。

- SolarisおよびRHEL-AS4(x86)のInterstage Application Server 8.0
- 上記以外のプラットフォームのInterstage Application Server/Interstage Web Server V9.0

制御方法は、以下から選択できます。

- アプリケーションにjava.lang.OutOfMemoryErrorを返却する
- プロセスを再起動する

本機能サポートバージョン以降で作成したIJServerは、「プロセスを再起動する」がデフォルトで設定されます。本機能サポートバージョン以前では「アプリケーションにjava.lang.OutOfMemoryErrorを返却する」が設定された場合と同様の制御となっていたため、必要に応じて設定を変更してください。

本機能サポートバージョン以前に作成したIJServerをリストアした場合には、「アプリケーションにjava.lang.OutOfMemoryErrorを返却する」がデフォルトで設定されます。

Interstage Application Server V6.0で作成したIJServerをリストアした場合は、本機能は設定できません。

Interstage Application Server V7.0以降で作成したIJServerの設定値は、Interstage管理コンソールまたは、isj2eadminコマンドで変更することができます。

6.2 J2EEアプリケーションの移行

ここでは、J2EEアプリケーションの移行について、以下を説明します。

- [IJServerの移行について](#)

Interstage Application Server 8.0以前から移行する場合に参照してください。

- [コンパイル時・実行時にクラスパスに追加するjarについて](#)

Interstage Application Server 8.0以前に作成したWebアプリケーションをコンパイルする場合に参照してください。

- [XMLパーサのXerces2サポート](#)

Interstage Application Server 8.0以前から移行する場合に参照してください。

- [Webサーバのバーチャルホストの表示変更について](#)

Interstage Application Server 8.0以前から移行する場合に参照してください。

- [データベースについて](#)

データベースを使用する場合に参照してください。

- [JDBCドライバ](#)

Interstage Application Server 8.0以前から移行する場合に参照してください。

- [自動再接続機能について](#)
Interstage Application Server 8.0以前から移行する場合に参照してください。
- [SQL実行の通信待ち時間監視について](#)
Interstage Application Server 8.0以前から移行する場合に参照してください。
- [コネクション使用時間監視について](#)
Interstage Application Server 8.0以前から移行する場合に参照してください。
- [クラスローダの変更について](#)
Interstage Application Server V6.0以前から移行する場合に参照してください。
- [返却される例外の詳細文字列について](#)
Interstage Application Server V6.0以前から移行する場合に参照してください。
- [JNDIから返却される例外について](#)
Interstage Application Server 5.x以前から移行する場合に参照してください。
- [J2EEアプリケーションの移行時のその他の注意事項](#)
Interstage Application Server 8.0以前から移行する場合に参照してください。

6.2.1 IJServerの移行について

IJServerの移行について、以下を説明します。

- [バックアップ/リストアについて](#)
- [V8.0互換モードのIJServerの機能範囲](#)
- [リトライカウントリセット契機の変更について](#)
- [XMLパーサの設定について](#)
- [Javaバージョンについて](#)
- [IJServerのファイル構成について](#)
- [\[IJServerディレクトリ\]/commonについて](#)
- [IJServerに対するconnectorの配備について](#)
- [名前変換機能について](#)
- [名前変換定義で参照名を重複して定義した場合について](#)
- [Interstage Application Server V5.x以前から移行する場合](#)
- [IJServerのクラスパス自動設定](#)
- [配備解除に失敗した場合について](#)
- [クラスのオートリロード機能](#)
- [クラスローダのトレース機能](#)
- [配備時に指定するモジュール名について](#)

■バックアップ/リストアについて

Interstage Application Server V7.0/8.0からの移行時

Interstage Application Server V7.0(Interstage V7.0)および8.0(Interstage 8.0)で作成したIJSerをバックアップ、およびリストアにより本バージョン・レベルへ移行した場合、V8.0互換モードのIJSerとなりIJSer(IJSer配下を含む)の機能範囲は、Interstage 8.0相当となります。V8.0互換モードのIJSerの説明については、“J2EEユーザーズガイド”の“J2EEアプリケーションが運用される環境(IJSer)”の“V8.0互換モードのIJSer”を参照してください。

本バージョン・レベルの機能を使用する場合は、V9.0以降のIJSerを作成し直して配備を行う必要があります。



注意

Interstage V7.0およびInterstage 8.0で作成したIJSerをバックアップ、およびリストアにより本バージョン・レベルへ移行する場合には、事前にTomcat4.1ベースのServletサービスがインストールされている必要があります。

なお、[RHEL5\(x86\)](#)、[RHEL5\(Intel64\)](#)、[RHEL5\(IPF\)](#)、[RHEL6\(x86\)](#)、[RHEL6\(Intel64\)](#)の場合にはTomcat4.1ベースのServletサービスは提供していないため、V9.0以降のIJSerを作成し直して配備を行う必要があります。

Interstage Application Server V6.0からの移行時

Interstage Application Server V6.0(Interstage V6.0)で作成したIJSerをバックアップ、およびリストアにより本バージョン・レベルへ移行した場合、IJSer(IJSer配下を含む)の機能範囲は、Interstage V6.0相当となります。

既存のIJSerの操作は、Interstage管理コンソールを使用して行います。

既存のIJSerに配備されたアプリケーションの配備解除、および、アプリケーションに対応する画面への遷移は、Interstage管理コンソールの[ワークユニット]>“ワークユニット名”>[アプリケーション状態/配備解除]タブで行ってください。

本バージョン・レベルの機能を使用する場合は、V9.0以降のIJSerを作成し直して配備を行う必要があります。

Interstage V6.0既存のIJSerは、本バージョン・レベルをインストールした後に作成し直すことはできません。Interstage V6.0既存のIJSerに万一のトラブルが発生した場合のため、本バージョン・レベルのインストール後も、環境が整った段階でバックアップを実施することを推奨します。



注意

Interstage V6.0で作成したIJSerをバックアップ、およびリストアにより本バージョン・レベルへ移行する場合には、事前にTomcat4.1ベースのServletサービスがインストールされている必要があります。

なお、[RHEL5\(x86\)](#)、[RHEL5\(Intel64\)](#)、[RHEL5\(IPF\)](#)、[RHEL6\(x86\)](#)、[RHEL6\(Intel64\)](#)の場合にはTomcat4.1ベースのServletサービスは提供していないため、V9.0以降のIJSerを作成し直して配備を行う必要があります。

■V8.0互換モードのIJSerの機能範囲

V8.0互換モードのIJSer(Interstage V7.0/8.0から移行したIJSer含む)の機能範囲はInterstage 8.0相当、Interstage V6.0から移行したIJSerの機能範囲はInterstage V6.0相当ですが、一部異なる機能範囲があります。異なる機能は以下のとおりです。

サービス	機能	変更点			
		新規作成のIJSer (V8.0互換モードのIJSer作成「しない」で作成)	V8.0互換モードのIJSer Interstage V7.0/8.0から移行したIJSer	Interstage 8.0のIJSer	Interstage V6.0から移行したIJSer
IJSer全体	サポートJavaバージョン	5.0	1.4	1.4 1.3	1.4
	Javaヒープ/Java Permanent領域不足時の制御	全OSサポート	全OSサポート	Solaris、 RHEL-AS4(x86) のみサポート	なし
Servletサービス	サポート規約	Servlet2.4/JSP2.0 (Tomcat5.5ベース)	Servlet2.3/JSP1.2 (Tomcat4.1ベース)	Servlet2.3/JSP1.2 (Tomcat4.1ベース)	Servlet2.3/JSP1.2 (Tomcat4.1ベース)

サービス	機能	変更点			
		新規作成の IJSERVER (V8.0互換モード のIJSERVER作 成「しない」で作 成)	V8.0互換モード のIJSERVER	Interstage 8.0 のIJSERVER	Interstage V6.0から移行し たIJSERVER
			Interstage V7.0/8.0から移 行したIJSERVER		
	セッションリカバリ機 能	全OSサポート	全OSサポート	Solaris、 RHEL- AS4(x86)のみ サポート	なし
EJBサー ビス	Message-driven Beanのアプリケー ション最大処理時 間監視 (注)	全OSサポート	全OSサポート	なし	全OSサポート
	EJBアプリケーション 運用におけるリ トライカウントリセ ット時間(注)	全OSサポート	全OSサポート	なし	全OSサポート
Webサー ビス	WS-I Attachment Profile 1.0に対応 した添付ファイル対 応	全OSサポート	全OSサポート	なし	なし

(注) IJSERVERタイプが以下の場合に提供される機能です。

- WebアプリケーションとEJBアプリケーションを別JavaVMで運用
- EJBアプリケーションのみ運用

■リトライカウントリセット契機の変更について

IJSERVERのアプリケーション異常終了した場合のリトライカウントリセット時間は、Interstage 8.0までは、以下のIJSERVERタイプでは設定できませんでしたが、Interstage V9.0以降では、以下のIJSERVERタイプでも設定可能になりました。

- EJBアプリケーションのみ運用するIJSERVER
- WebアプリケーションとEJBアプリケーションを別JavaVMで運用するIJSERVERのEJBアプリケーション運用

上記の場合、従来の監視方式では、EJBアプリケーションへのビジネスロジック(Remoteインタフェース)へアクセスする前に、必ずHomeインタフェースメソッド(初期処理)実行が正常終了した時点でリトライカウントリセットがクリアされていましたが、Interstage V9.0以降では、リトライカウントリセット時間で指定した時間が経過すると、リトライカウントリセットされます。

この変更はV8.0互換モードのIJSERVERや8.0以前の環境から移行されたIJSERVERについても適用されるため、必要に応じて再設定してください。

リトライカウントリセット時間はInterstage管理コンソールまたはisj2eeadminコマンドを使用して定義変更が可能です。必要に応じて変更を行ってください。

- Interstage管理コンソールの場合
以下の画面の「リトライカウントリセット時間」の値を変更してください。
ー [ワークユニット] > [IJSERVER名] > [環境設定]
- isj2eeadminコマンドの場合
IJSERVER定義ファイルの以下の項目の値を変更してください。
ー IJSERVER > Common > RetryCountResetTime

■XMLパーサの設定について

Interstage 8.0までは、Crimsonが配備時のXMLパーサとして使用されていました。

Interstage V9.0以降では、IJServerの環境設定で指定したXMLパーサが、配備時のXMLパーサとして使用されるようになりました。XMLパーサによりサポートしている文字エンコーディングが異なるため、IJServerの環境設定で指定したXMLパーサが、deployment descriptorを記述している文字エンコーディングをサポートしているか確認してください。

なお、従来通りにCrimsonを使用する場合は、IJServerの環境設定でCrimsonが有効となるように設定してください。

また、Webサービスを利用する場合、JAXP1.2以上をサポートしているXMLパーサを指定する必要があります。

Interstage 8.0までは、IJServerの環境設定で「コンテナのWebサービス機能」の定義を“有効”に設定している場合に「使用するXMLパーサの種別」の“その他”にJAXP1.2をサポートしていないXMLパーサ指定してもチェックが行われていませんでしたが、Interstage V9.0以降では、IJServerの環境設定でチェックするように改善されました。

Webサービスを利用する場合は、JAXP1.2以上をサポートしているXMLパーサを使用するように設定してください。

なお、デフォルトではIJServerの環境設定の「コンテナのWebサービス機能」は“有効”に設定されています。Webサービスを利用しない場合は、この設定を“無効”にすることで、JAXP1.2をサポートしていないXMLパーサを使用することができます。

なお、V8.0互換モードのIJServerやInterstage V8以前で作成されたIJServerについては、従来通りの動作となります。

配備時に使用されるXMLパーサについては、“J2EE ユーザーズガイド”の“配備に必要なXMLパーサの設定”を参照してください。

■Javaバージョンについて

IJServer定義のJDKバージョンにインストールされていないJDKが指定されている場合、本製品のV7.0以前はインストールされているJDKでIJServerを起動していましたが、本製品のV8.0以降ではIJServer起動時にエラーとなります。

起動エラーとなった場合には、IJServerの環境設定で“Javaバージョン”を、インストールされているJavaのバージョンに変更してIJServerを起動してください。

■IJServerのファイル構成について

Interstage V7.0以降、J2EEアプリケーションの配備後の管理単位が、アプリケーション単位からモジュール単位に変更されました。

この変更に伴いIJServerのファイル構成が変更されました。

IJServerのファイル構成については“J2EE ユーザーズガイド”の“IJServerのファイル構成”を参照してください。

Interstage Application Server V6.0(Interstage V6.0)で作成したIJServerをバックアップ、およびリストアにより本バージョン・レベルへ移行した場合、既存のIJServer(IJServer配下を含む)のファイル構成は、Interstage V6.0相当となります。

■[IJServerディレクトリ]/commonについて

Interstage V7.0以降、EARファイル内のWARファイル、ejb-jarファイル、RARファイル以外の資源の展開先が、以下のように変更されました。

	Interstage V6.0以前	Interstage V7.0以降
仕様: EARファイル内のWARファイル、 ejb-jarファイル、RARファイル以外 の資源の展開先	[IJServerディレクトリ]/common以下	[IJServer]ディレクトリ]/apps/[アプリケーション名]以下 ※アプリケーション間の独立性を 高めたため
影響: common直下に保管したJARファイル の扱い	IJServerの起動時にクラスパスに 設定される	クラスパスに設定されない

Interstage V6.0以前にcommon直下に保管していたJARファイルを、本バージョン・レベルで使用する場合は、以下のいずれかの対応を行う必要があります。

- ・ マニフェストクラスパスを使用する
- ・ EARファイル内のSharedディレクトリにJARファイルを保管する
- ・ [IJServerディレクトリ]/SharedディレクトリにJARファイルを保管する
- ・ ワークユニットのクラスパスに設定する

ただし、J2EEの仕様に準拠したEARファイルの作成を行っている場合はマニフェストクラスパスが指定されていますので、上記の対応は不要です。

■IJServerに対するconnectorの配備について

Interstage V6.0ではIJServerにconnectorを配備した場合にも、システム全体で一意な資産として管理され、各IJServerで参照可能でした。

Interstage V7.0以降ではIJServerに配備したconnectorは、IJServer内で管理され、そのIJServer内でのみ参照可能となりました。

Interstage管理コンソールで、[リソース] > [connector]で配備されたconnectorとIJServerに配備されたconnectorが重複した場合は、IJServerのconnectorが優先されます。

■名前変換機能について

Interstage V6.0でのWebアプリケーション、およびEJBアプリケーションの名前変換は、IJServer単位での設定となっていました。

Interstage V7.0以降ではモジュール単位での設定に変更されました。設定はInterstage管理コンソールの各モジュールの画面から行うことが可能です。

これにともない、配備時にIJServer単位の名前変換定義の更新は行われなかったため、Interstage V6.0では出力されていた下記のメッセージは配備時に出力されなくなりますが、名前変換定義は有効になっています。

IS: 情報: is40902: 名前変換定義を更新しました

Interstage管理コンソールを使用しない場合の設定方法については、“J2EE ユーザーズガイド”の“名前変換機能”を参照してください。

■名前変換定義で参照名を重複して定義した場合について

名前変換定義は参照名を重複して定義することが可能ですが、Interstage V9.0.0以前では重複定義された場合にどの定義値を有効にするか明確にしていませんでした。

【重複定義の例】(jdbc/DataSourceが重複している)

```
<res-entry>
  <res-ref-name>jdbc/DataSource</res-ref-name>  ★ 参照名が重複している
  <datasource-name>DS1</datasource-name>
</res-entry>
<res-entry>
  <res-ref-name>jdbc/DataSource</res-ref-name>
  <datasource-name>DS2</datasource-name>
</res-entry>
<res-entry>
  <res-ref-name>jdbc/DataSource</res-ref-name>
  <datasource-name>DS3</datasource-name>
</res-entry>
```

Interstage V9.0.0以前は、以下のように各モジュールにより有効になる定義値が統一されていませんでした。

Interstage V9.0.1/V9.1以降では重複定義された場合、最後の定義値が有効になります(例ではDS3が有効)。

		V9.0.0以前	V9.0.1/V9.1以降
定義時	管理コンソール	最後の定義値が有効	最後の定義値が有効
実行時	J2EEクライアント	最初の定義値が有効	最後の定義値が有効
	Webアプリケーション	最初の定義値が有効	最後の定義値が有効
	EJBアプリケーション	最後の定義値が有効	最後の定義値が有効

名前変換を行うアプリケーションがJ2EEクライアントまたはWebアプリケーションの場合、有効になる定義値が変更になることがあります。従来通りに動作させたい場合は、以下のオプションを設定してください。

定義場所	JavaVMオプション
定義名	com.fujitsu.interstage.j2ee.jndi.EbeDef_CheckOrder_FromHead
定義値	true
記述例	-Dcom.fujitsu.interstage.j2ee.jndi.EbeDef_CheckOrder_FromHead=true

意味 J2EEクライアントアプリケーションまたはWebアプリケーションで名前変換定義に定義した値は先頭に定義した値が有効になります。

■Interstage Application Server V5.x以前から移行する場合

J2EEアプリケーションをInterstage Application Server V5.x(Interstage V5.x)以前のバージョンから移行する場合には、J2EEアプリケーションをIJServerに配備する必要があります。

Interstage管理コンソールの[ワークユニット] > [新規作成]タブ、または、isj2eeadminコマンドを使用して、IJServerワークユニットを作成してください。そのあと、“ワークユニット名” > [配備]タブで配備を行います。

Interstage管理コンソールの詳細については、Interstage管理コンソールのヘルプを参照してください。

isj2eeadminコマンドについては、“リファレンスマニュアル(コマンド編)”の“isj2eeadminコマンド”を参照してください。

■IJServerのクラスパス自動設定

IJServerのクラスパスは内部的に自動的に設定されます。

CORBAアプリケーションのライブラリとしてはブレインストール型Javaライブラリがクラスパスに常に設定され、Portable-ORB用のライブラリは使用できませんが、V5.1以前のServletサービスでPortable-ORB用のライブラリで動作させていたアプリケーションも、Interstage V6.0以降のIJServerでそのまま動作させることができます。

CORBAサービスの環境定義を変更する場合にはブレインストール型の設定を変更してください。

以下のクラスパスは、V9.0以降のIJServerでは自動設定されません。

Windows

JDKのインストールディレクトリ\lib\tools.jar

Solaris

Linux

JDKのインストールディレクトリ/lib/tools.jar

アプリケーションで必要な場合は、“J2EE ユーザーズガイド”の“クラスローダの構成”および“IJServerで使用するクラスの設定について”を参照して設定を行ってください。

■配備解除に失敗した場合について

Interstage Application Server/Interstage Web Server V9.0以降では、配備解除の際にファイルの削除に失敗した場合、手動でファイルの削除を行う必要があります。

配備対象モジュールのファイルのみ残った状態ではモジュールはロードされないため、IJServerの運用に影響を及ぼしますが、同一モジュールの再配備を行った場合に、モジュールの配備先に不明なファイルが存在することになり、再度削除を促すためのメッセージが出力され、配備処理がエラーとなります。このような場合、不要なファイルを削除してから、配備処理を行ってください。

■クラスのオートリロード機能

Interstage Application Server 8.0までは、クラスのオートリロード機能はモジュール(EAR/WAR/ejb-jar)単位での設定でした。

Interstage Application Server/Interstage Web Server V9.0以降では、設定はIJServer単位に変更されています。個々のモジュールごとに設定を行う必要はありません。

モジュールによりオートリロード機能の設定を切り替える必要がある場合には、モジュールを別のIJServerに配備し、各IJServerで必要であればオートリロードの設定を行ってください。

また、クラスローダの分離の設定に「分離しない」を選択している場合、Interstage 8.0までのIJServerではWebアプリケーション(WAR)のみオートリロード機能が使用可能でしたが、Interstage V9.0以降では、オートリロード機能は使用できません。

他の「クラスローダの分離」の設定とするか、またはモジュールを別のIJServerに配備し、各IJServerで必要であればオートリロードの設定を行ってください。

■クラスローダのトレース機能

Interstage Application Server 8.0までは、クラスローダのトレース機能の設定は、以下で設定可能でした。

- Interstage管理コンソール
[ワークユニット] > “ワークユニット名” > [環境設定]タブ > [共通定義] > [クラスローダのトレース情報の出力]
- isj2eeadminコマンドのIJServer定義ファイル
IJServerタグ/ClassLoaderタグ/Traceタグ

Interstage Application Server/Interstage Web Server V9.0以降でトレース情報を出力するには、以下でJavaVMオプションを設定します。

- Interstage管理コンソール
[ワークユニット] > “ワークユニット名” > [環境設定]タブ > [ワークユニット設定] > [JavaVMオプション]
- isj2eeadminコマンドのIJServer定義ファイル
IJServerタグ/ClassLoaderタグ/JavaCommandOptionsタグ
指定値: -Dcom.fujitsu.interstage.j2ee.ijservice.loader.trace=true

■ 配備時に指定するモジュール名について

Interstage 8.0までは、モジュール名の末尾にピリオドを指定した場合、配備資源の展開が不正なディレクトリに行われることがありました。Interstage V9.0以降では、モジュール名の末尾にピリオドを指定した場合、配備時にエラーとなり、DEP1833のメッセージが出力されます。Interstage V9.0以降で配備を行う場合、モジュール名の末尾には、ピリオド以外の文字を指定してください。モジュール名の詳細については、Interstage管理コンソールのヘルプの配備を参照してください。

なお、モジュール名の末尾にピリオドが指定されたモジュールが配備されているInterstage 8.0以前のIJServerをバックアップ、およびリストアにより本バージョンレベルに移行することは可能です。

6.2.2 コンパイル時・実行時にクラスパスに追加するjarについて

Interstage Application Server 8.0以前に、Servlet API/JavaServer Pages APIを使用したJavaアプリケーションをコンパイルまたは実行するために“isj2ee.jar”をクラスパスに指定していた場合は、“isj2ee.jar”のかわりに以下のjarをクラスパスに指定する必要があります。

Windows

C:\Interstage\F3FMjs5\common\lib\servlet-api.jar
C:\Interstage\F3FMjs5\common\lib\jsp-api.jar

Solaris

Linux

/opt/FJSVjs5/common/lib/servlet-api.jar
/opt/FJSVjs5/common/lib/jsp-api.jar

6.2.3 XMLパーサのXerces2サポート

Interstage Application Server V7.0では、IJServerで使用するXMLパーサのデフォルトはCrimsonでしたが、本バージョン・レベルでは、デフォルトでXerces2が使用されます。また、V9.0以降のIJServerではXMLパーサにCrimsonは選択できません。従来通りにCrimsonを使用する場合には、以下の対処を実施してください。Crimsonを使用するには、「コンテナのWebサービス機能」の定義を“無効にする”に設定してください。

V8.0互換モードのIJServerの場合

Interstage管理コンソールの[ワークユニット] > [IJServer名] > [環境設定]で、使用するXMLパーサの種別にCrimsonを選択してください。また、isj2eeadminコマンドでも設定できます。

V9.0以降のIJServerの場合

Interstage管理コンソールの[ワークユニット] > [IJServer名] > [環境設定]で、使用するXMLパーサの種別にその他を選択して、ApacheのホームページからダウンロードしたCrimsonのjarファイルを格納したディレクトリを指定してください。また、isj2eeadminコマンドでも設定できます。

■ Xerces2がサポートするエンコード

Xerces2では、以下のエンコードをサポートしています。使用するXMLパーサによってサポートするエンコードが異なるため、使用するXMLパーサのエンコードを確認してください。どのXMLパーサでも共通で使用可能なUTF-8を使用することをお勧めします。

UTF-8	ISO-8859-7	ebcdic-cp-no	ebcdic-cp-ar1	gb2312
ISO-8859-1	ISO-8859-8	ebcdic-cp-fi	ebcdic-cp-he	euc-jp
ISO-8859-2	ISO-8859-9	ebcdic-cp-se	ebcdic-cp-ch	iso-2020-jp
ISO-8859-3	ebcdic-cp-us	ebcdic-cp-it	ebcdic-cp-roece	Shift_JIS
ISO-8859-4	ebcdic-cp-ca	ebcdic-cp-es	ebcdic-cp-yu	Big5
ISO-8859-5	ebcdic-cp-nl	ebcdic-cp-gb	ebcdic-cp-is	euc-kr

ISO-8859-6	ebcdic-cp-dk	ebcdic-cp-fr	ebcdic-cp-ar2	koi8-r
------------	--------------	--------------	---------------	--------

■使用するJavaバージョンによる違い

Xerces2を使用する場合、使用するJavaバージョンによって以下の違いがあります。

	1.4	5.0
Xercesの種類	バージョン2.7.1のXerces	JDK5.0に内包されたXerces
Xercesの各実装クラスの パッケージ名	org.apache.xerces	com.sun.org.apache.xerces.internal

Javaバージョン1.4で運用したアプリケーションがJAXPのインタフェース(javax.xmlパッケージのインタフェース)でXMLファイルを扱っていれば問題ありません。仮にJavaバージョン1.4で直接Xercesの実装クラスを参照していたアプリケーションをJavaバージョン5.0に移行する場合には、アプリケーションを修正して上記のようにパッケージ名を変更してください。もしくは以下を指定することで動作するXMLパーサをJavaバージョン1.4と同様のXMLパーサに変更することもできます。

変更方法

IJServerの環境設定において使用するXMLパーサに“その他”を選択し、ディレクトリに以下を設定してください。

Windows

C:\Interstage\J2EE\lib\xerces

Solaris

Linux

/opt/FJSVj2ee/lib/xerces

6.2.4 Webサーバのバーチャルホストの表示変更について

本製品では、Interstage管理コンソールの下記項目にメインホストは表示せず、バーチャルホストしか表示しません。そのため、バーチャルホストが設定されていない場合には本項目自体表示しません。

ただし、Interstage Application Server V7.0/V6.0から本バージョン・レベルへのアップグレードを行った場合、既存のIJServerの下記項目でメインホストを選択していた場合には選択されたメインホストを表示します。

【V8.0互換モードのIJServerの場合】

- ・ [ワークユニット] > [新規作成] > [詳細設定] > [Webサーバコネクタ(コネクタ)設定] > [Webサーバのバーチャルホスト]
- ・ [ワークユニット] > “ワークユニット名” > [環境設定] > [詳細設定] > [Webサーバコネクタ(コネクタ)設定] > [Webサーバのバーチャルホスト]

【V9.0以降のIJServerの場合】

- ・ [ワークユニット] > [新規作成] > [詳細設定] > [Webサーバ/Webサーバコネクタ(コネクタ)設定] > [Webサーバのバーチャルホスト]
- ・ [ワークユニット] > “ワークユニット名” > [環境設定] > [詳細設定] > [Webサーバコネクタ(コネクタ)設定] > [Webサーバ/Webサーバのバーチャルホスト]

6.2.5 データベースについて

データベースの以下の変更内容について説明します。

- ・ **Oracleを使用する場合**
 - OCIドライバを使用する場合の変更
 - Oracleのコネクションプーリング
 - グローバルトランザクションについて
- ・ **Symfowareを使用する場合**
 - コネクションプーリングのデフォルト変更

- V4.0互換のJNDIサービスプロバイダについて
- SQL Serverを使用する場合
 - SQL Server 2000について
- PostgreSQLを使用する場合
 - JDBC 2.0 + Optional Packageについて
- データベース共通
 - V8.0互換モードのIJServer、またはInterstage 8.0以前の環境から移行したIJServerで利用できるデータソースについて
 - File System Service Providerについて
 - パスワードの省略について
 - isj2eeadminコマンドによる定義更新時のDatabaseKind変更不可
 - SQL文のキャッシュ機能について
 - データソースのキャッシュについて

■Oracleを使用する場合

◆OCIドライバを使用する場合の変更

Oracle OCIドライバを使用し、データソースの「File System Service Providerを使用する」の項目を“使用しない”(デフォルト)にした場合、PG60150の修正を適用していないInterstage V9.0では、ネットサービス名を指定してもSIDとして動作していました。V9.1以降では、“SID/ネットサービス名”に指定した値はOCIドライバの場合ネットサービス名として動作します。V9.0と同様にOCIドライバでSIDとして動作させたい場合は、J2EEプロパティファイルに以下の行を追加してください。

- 編集対象のファイル(J2EEプロパティファイル)

Windows

C:\¥Interstage¥J2EE¥etc¥isj2ee.properties

Solaris Linux

/opt/FJSVj2ee/etc/isj2ee.properties

- 追加する内容
oracle.oci.sid=yes

◆Oracleのコネクションプーリング

本製品では“Oracleのコネクションプーリングを使用する”ことができますが、Interstage V8.0以前で定義可能であった“Oracleでコネクションプーリングを行う”とは別機能になります。そのため本製品ではInterstage管理コンソールを使用してInterstage V8.0以前で定義可能であった“Oracleでコネクションプーリングを行う”を定義できません。

Oracle9i以前を使用して“Oracleでコネクションプーリングを行う”データソースを定義したい場合は、以下の方法を用いてください。

- “Oracleでコネクションプーリングを行う”を定義したデータソース資源をリストアする
- “File System Service Providerを使用する”をチェックし、“データソースを登録する(.bindingsファイルへの登録)”をチェックしない
データソース定義を作成し、データソースクラスにOracleConnectionCacheImplクラスを用いて.bindingsファイルを直接作成する

なお、Interstage V8.0以前の“Oracleでコネクションプーリングを行う”を定義したデータソースを使用することは、Interstage V9.0以降では非推奨です。必要な場合にのみ上記対処を行い、それ以外の場合は、データソース定義を削除し再作成を行ってください。Interstage管理コンソールにより参照した場合、データソースの種類はいずれも選択されていない状態で表示され、Interstage管理コンソールでは更新できません。

Oracle10g以降ではInterstage V8.0以前の“Oracleでコネクションプーリングを行う”で定義されたデータソース定義で使用するOracleConnectionCacheImplクラスが未サポートとなっているため、使用できません。

◆グローバルトランザクションについて

Interstage Application Server 8.0以前において定義可能であったグローバルトランザクションを使用する場合は、「データソースの種類」で「分散トランザクションを使用する」を選択してください。

またInterstage Application Server 8.0以前で「グローバルトランザクションを利用する」で定義されたデータソースを本製品へリストアした場合、「分散トランザクションを使用する」が選択された状態で表示されます。

■Symfowareを使用する場合

◆コネクションプーリングのデフォルト変更

Interstage V9.0より「データソースの種類」に「Interstageのコネクションプーリングを使用する」が選択可能となりました。

Interstage管理コンソールおよびisj2eeadminコマンドでSymfowareのJDBCデータソースを作成する場合、従来は「データソースの種類」のデフォルトが「Symfowareのコネクションプーリングを使用する」でしたが、Interstage V9.0より「Interstageのコネクションプーリングを使用する」がデフォルトとなります。

isj2eeadminコマンドで、旧バージョンの定義ファイルをそのまま使用し「Symfowareのコネクションプーリングを使用する」を登録する場合は、isj2eeadminコマンドに“-v 8.0”オプションを付加してください。

詳細は「リファレンスマニュアル(コマンド編)」の“isj2eeadmin”を参照してください。

◆V4.0互換のJNDIサービスプロバイダについて

V4.0以前のSymfowareは未サポートです。Interstage V8.0以前はInterstage管理コンソールを使用してSymfowareのデータソースを定義する時に、JNDIサービスプロバイダのクラス名にV4.0以前のSymfowareのクラス名(fujitsu.symfoware.jdbc2.jndisp.SYMContextFactory)を選択できましたが、Interstage V9.0以降では選択できません。旧バージョン・レベルの資産を利用し、データソース定義にJNDIサービスプロバイダのV4互換のクラス名が設定されている場合は、isj2eeadminコマンドを使用してInitialContextFactoryタグの値をサポートするクラス名に修正してください。

詳細は「リファレンスマニュアル(コマンド編)」の“isj2eeadmin”を参照してください。

■SQL Serverを使用する場合

◆SQL Server 2000について

SQL Server 2000はサポート対象外となりました。SQL Server 2005以降を利用してください。

■PostgreSQLを使用する場合

◆JDBC 2.0 + Optional Packageについて

PostgreSQLのJDBC 2.0 + Optional Package のJDBCドライバはサポート対象外となりました。JDBC 3.0のJDBCドライバを使用してください。

■データベース共通

◆V8.0互換モードのIJServer、またはInterstage 8.0以前の環境から移行したIJServerで使用できるデータソースについて

V8.0互換モードのIJServer、またはInterstage 8.0以前の環境から移行したIJServerでは、Interstage V9.0より追加・変更された以下のデータソースは使用できません。これら以外のデータソースを使用してください。

- ・ データベースタイプが“Oracle”かつ、“Oracleのコネクションプーリングを使用する”に設定されたデータソース
- ・ データベースタイプが“Oracle”かつ、“RACを使用する”に設定されたデータソース
- ・ データベースタイプが“Symfoware”かつ、“Interstageのコネクションプーリングを使用する”に設定されたデータソース
- ・ データベースタイプが汎用定義のデータソース
- ・ File System Service Providerを使用しないデータソース

◆File System Service Providerについて

Interstage V9.0より、File System Service Providerを使用せずにJDBCデータソースを登録することが可能となりました。デフォルトではFile System Service Providerを使用しない場合のデータソースが定義されます。

isj2eeadminコマンドにより、旧バージョン・レベルで抽出した定義ファイルでFile System Service Providerを使用するデータソースを登録する場合は、“-v 8.0”オプションを指定してください。詳細は「リファレンスマニュアル(コマンド編)」の“isj2eeadmin”を参照してください。

V8.0互換モードのIJServerまたは8.0以前に作成されたIJServerから、Oracle、SQL Server、PostgreSQLのデータソースを使用する場合は、従来通りFile System Service Providerを使用する必要があります。

◆パスワードの省略について

Interstage V9.0より、データソース定義の登録、更新時にパスワードが省略可能となります。このため、isj2eadminコマンドでパスワードを省略した場合、エラーとならず登録が完了します。

◆isj2eadminコマンドによる定義更新時のDatabaseKind変更不可

Interstage V9.0より、isj2eadminコマンドによりデータソース定義を更新する際、DatabaseKindは定義更新時に変更できなくなります。DatabaseKindを変更する場合は、一度削除してから再度作成してください。

◆SQL文のキャッシュ機能について

Interstage 8.0までサポートしていたCMP1.1の範囲で有効だったSQL文のキャッシュ機能はInterstage V9.0以降では未サポートとなりました。

Interstage V9.0より、データベースタイプが“Oracle”で、データソースの種類が“Oracleのコネクションプーリングを使用する”の場合とデータベースタイプが“Symfoware”で、データソースの種類が“Interstageのコネクションプーリングを使用する”の場合にStatementキャッシュ機能をサポートしましたので、“チューニングガイド”の“Statementキャッシュ機能”を参照し使用してください。

◆データソースのキャッシュについて

Interstage V9.0より、初回に参照したデータソース定義情報、データソースオブジェクトをキャッシュするようになりました。旧バージョン・レベルでは、IJServer起動後にデータソース定義を変更しても有効になる場合がありますでしたが、Interstage V9.0以降ではIJServer起動前にデータソースの定義登録を行ってください。

6.2.6 JDBCドライバ

Windows

Interstage JDBC Driverは非推奨機能のため未サポートとなりました。Microsoft(R) JDBCドライバを使用してください。移行は以下の手順により行います。

1. Microsoft(R) JDBCドライバをMicrosoft Corporationのホームページよりダウンロードしてください。JDBCドライバはSQL Server(TM)のバージョンに合わせて適切なものを使用してください。
2. 旧バージョンの資産をリストアした場合は、以下の点に注意し定義を更新してください。
 - ー 必須項目“データベース名”を入力する必要があります。
 - ー PROVIDER_URLがInterstage JDBCドライバに対応した記述形式になっていますので、詳細設定の“File System Service Providerを使用する”のチェックを解除しJDBCデータソース定義の.bindingsファイルへの登録を行わないようにしてください。もしくは管理コンソールのヘルプまたは“リファレンスマニュアル(コマンド編)”の“isj2eadmin”の“リソース定義ファイル”を参照して正しいPROVIDER_URLを設定してください。
3. J2EEプロパティまたはIJServerの環境設定などに指定されているInterstage JDBCDriverのクラスパスを、Microsoft(R) JDBCドライバのクラスパスへ変更してください。

6.2.7 自動再接続機能について

Interstage V9.0より、JDBCデータソースの自動再接続機能がデフォルトで有効となります。自動再接続機能が有効になることによる運用上の問題はありますが、データベース側のSQL文のログを参照した場合、接続可能かどうかを確認するために発行しているSQL文が多数出力されています。デバッグする場合などで自動再接続機能を無効にしたい場合は、Interstage管理コンソールまたはisj2eadminコマンドで以下の定義変更を行い、自動再接続が行われないようにしてください。

- Interstage管理コンソールの場合
以下の画面の「異常時の再接続」を「しない」に変更。
 - ー [ワークユニット] > [IJServer名] > [環境設定]
- isj2eadminコマンドの場合
IJServer定義ファイルの以下の項目をFALSEに指定して定義を変更。
 - ー IJServer > Datasources > Datasource > AbnormalReconnection

旧バージョン・レベルでバックアップした資産をリストアした場合、以下の違いがあります。

- 旧バージョン・レベルで、Interstage管理コンソールよりDBコネクション設定でデータソースのチェックボックスをオンに定義した場合デフォルトは異常時の再接続を「しない」になっていますので、リストア後も「しない」です。
- 旧バージョン・レベルで、isj2eeadminコマンドでDatasourceタグを指定しAbnormalReconnectionタグを省略した場合旧バージョン・レベルのデフォルトが定義されますので、リストア後も「FALSE」です。
- 旧バージョン・レベルで、Interstage管理コンソールのDBコネクション設定でデータソースのチェックボックスをオンにせず、isj2eeadminコマンドでもDatasourceタグを定義していない場合旧バージョン・レベルの定義がないためInterstage V9.0以降のデフォルトとなり、自動再接続機能が有効となります。
- V6で作成したIJServerの場合デフォルトは「しない」のままです。

また、旧バージョン・レベルのisj2eeadminコマンドの定義をそのまま利用してIJServerを作成した場合も上記の通りになります。V8.0互換モードのIJServerを作成した場合、デフォルトは「する」です。

6.2.8 SQL実行の通信待ち時間監視について

Interstage V9.0よりSQL実行処理の通信待ち時間をデフォルトで監視します(デフォルト監視時間:400秒)。そのため、400秒のSQL実行処理時間を許容範囲とする業務システムの場合、アプリケーションが正常動作しているにも関わらずIJServer21265の警告メッセージが出力される場合がありますが、ユーザは出力されるメッセージから問題がないことを確認することができます。

警告メッセージの出力を抑止したい場合は、Interstage管理コンソール、もしくはisj2eeadminコマンドで以下の定義変更を行い、SQL実行の通信待ち時間監視を行わないようにしてください。

- Interstage管理コンソールの場合
以下の画面の「通信待ち時間」を「0」に変更。
「ワークユニット」>「IJServer名」>「環境設定」
- isj2eeadminコマンドの場合
IJServer定義ファイルの以下の項目を「0」に指定して定義を変更。
IJServer > Datasources > Datasource > SqlWaitTimeout > Time

6.2.9 コネクション使用時間監視について

Interstage V9.0よりトランザクション開始前に獲得したコネクションがクローズされるまでをデフォルトで監視します(デフォルト監視時間:60分)。そのため、アプリケーションで獲得したコネクションを意図的にクローズせずに使用する業務システムの場合、アプリケーションが正常動作しているにも関わらずIJServer21263の警告メッセージが出力される場合がありますが、ユーザは出力されるメッセージから問題がないことを確認することができます。

警告メッセージの出力を抑止したい場合は、Interstage管理コンソール、もしくはisj2eeadminコマンドで以下の定義変更を行い、コネクション使用時間監視を行わないようにしてください。

- Interstage管理コンソールの場合
以下の画面の「コネクション使用監視時間」を「0」に変更。
「ワークユニット」>「IJServer名」>「環境設定」
- isj2eeadminコマンドの場合
IJServer定義ファイルの以下の項目を「0」に指定して定義を変更。
IJServer > Datasources > Datasource > ConnectionUseTimeout > Time

6.2.10 クラスローダの変更について

Interstage Application Server V7.0(Interstage V7.0)以降では、アプリケーションのクラスのロードを行うクラスローダ機能が提供されました。クラスローダ機能の提供により、クラスローダのデフォルト設定が従来と異なります。このため、クラスローダの分離方法をInterstage Application Server V6.0(Interstage V6.0)までの互換である“分離しない”以外を選択した場合は、以下の点に注意してください。

- 配備したモジュール間で参照関係がある場合、他のモジュールのクラスやEJBアプリケーションを正しく参照できないことがあります。

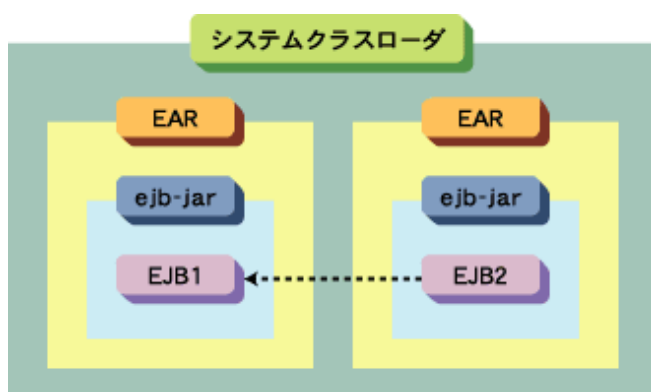
- ・異なるクラスローダで同一名のクラスをロードした場合、ClassCastExceptionが発生することがあります。
- ・環境変数CLASSPATHが有効となりません。
- ・メモリ使用量が変動する場合があります。“[■メモリ使用量について](#)”を参照して、再度見積りを実施してください。

Interstage V6.0との違いについては、以下を参照してください。また、クラスローダの詳細については“J2EE ユーザーズガイド”の“クラスローダ”を参照してください。

■IJServerのタイプが“WebアプリケーションとEJBアプリケーションを同一JavaVMで運用”である場合の、アプリケーション間の参照について

◆Interstage V6.0

EARファイルを配備すると、下図のように、EARファイルに含まれるEJBアプリケーションはシステムクラスローダでロードされました。そのため、同一プロセス内の他のEARファイルに含まれるEJBアプリケーションを参照できました。



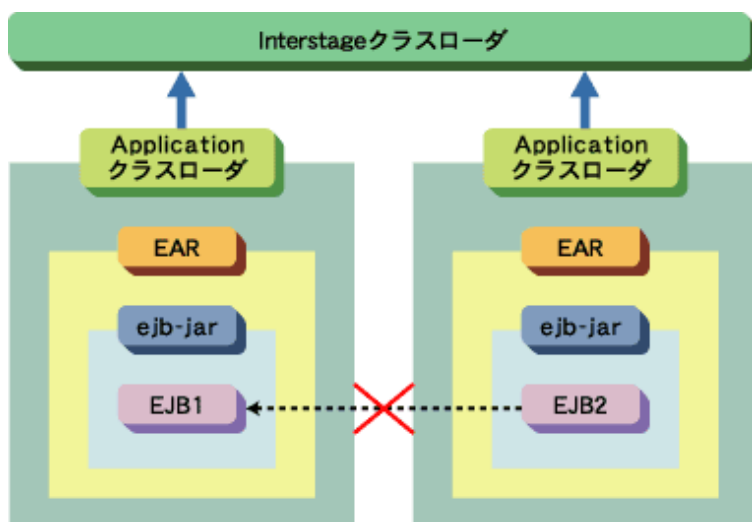
◆Interstage V7.0以降

“クラスローダの分離”の設定値やアプリケーションの配備方法によって、アプリケーション間の参照可否が異なります。“クラスローダの分離”には“EAR間で分離”(デフォルト設定)を設定することをお勧めしますが、以下を参考にして、設定内容を検討してください。

1) “クラスローダの分離”が“EAR間で分離”(デフォルト設定)の場合

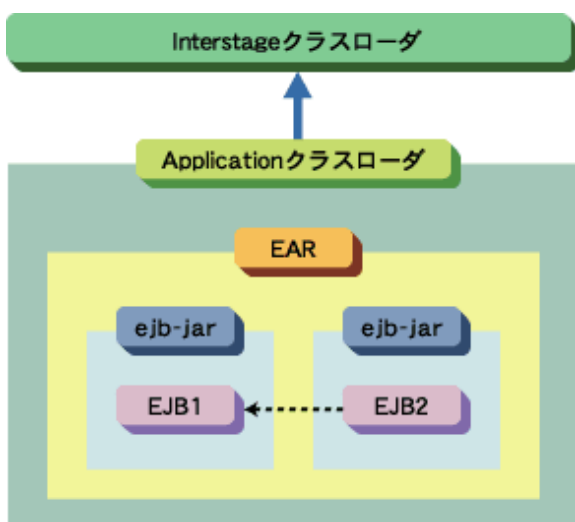
EARファイルを配備すると、下図のように、クラスローダはEARごとに割り当てられます。このようにクラスローダを割り当てることにより、アプリケーション間の独立性が高められました。しかし、その一方で、同一プロセス内の他のEARに含まれるEJBアプリケーションを参照できなくなりました。

注) ejb-jarを単体で配備した場合は、V6.0までと同様に、同一プロセス内の他の単体で配備したejb-jarに含まれるEJBアプリケーションを参照できます。



2) 同一プロセス内の他のEARに含まれるEJBアプリケーションを参照する方法(1)

下図のように、関連するEJBアプリケーションを1つのEARファイルに含めてIIServerに配備してください。

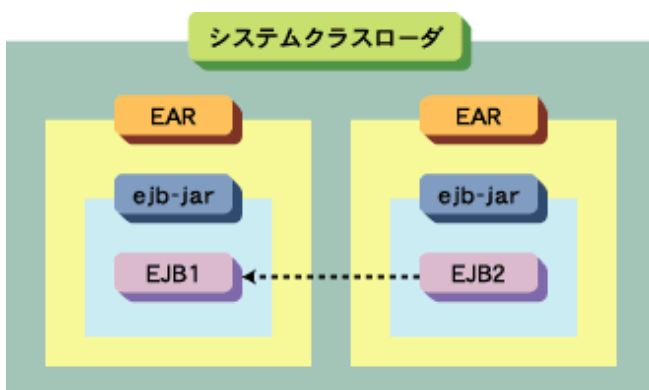


3) 同一プロセス内の他のEARに含まれるEJBアプリケーションを参照する方法(2)

関連するEJBアプリケーションを1つのEARファイルに含めることができない場合は、以下のどちらかの方法で、“クラスローダの分離”に“分離しない”を設定してください。

- ・ [ワークユニット] > [新規作成] > [詳細設定] > [共通定義] > [クラスローダの分離]
- ・ [ワークユニット] > [ワークユニット名] > [環境設定] > [詳細設定] > [共通定義] > [クラスローダの分離]

なお、上記は、isj2eadminコマンドで設定することもできます。詳細は、“リファレンスマニュアル(コマンド編)”を参照してください。クラスローダの分離については“J2EE ユーザーズガイド”の“クラスローダの分離”を参照してください。

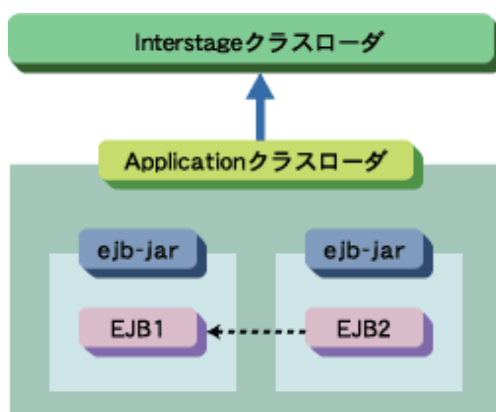


4) HotDeployを行う必要がある場合

“クラスローダの分離”に“分離しない”を設定した場合は、HotDeployができません。HotDeployを行う必要がある場合は、以下のどちらかの対処をしてください。

- ・ 関連するEJBアプリケーションを1つのEARファイルとしてIIServerに配備

- ・ 下図のようにEARファイル内の個々のejb-jarファイルをIJSERVERに配備
この場合、“クラスローダの分離”に“EAR間で分離”を設定します。



■CLASSPATHについて

Interstage V6.0では環境変数:CLASSPATHに設定された値はIJSERVER起動時に有効となっていました。

Interstage V7.0以降ではクラスローダの分離が“EAR間で分離”または“すべて分離”の場合、以下の理由により環境変数:CLASSPATHはIJSERVER起動時に有効となりません。

理由

環境変数:CLASSPATHには、Interstageだけではなくそのサーバ上で動作する他のシステムが使用するライブラリが指定されている場合や、無効なライブラリ(旧バージョンのライブラリなど)が指定されている可能性があります。そのため、環境変数:CLASSPATHをIJSERVER起動時に有効とすると、IJSERVERが誤動作する原因となります。

なお、クラスローダの分離が“分離しない”の場合は、Interstage V6.0と同様に環境変数:CLASSPATHはIJSERVER起動時に有効となります。この場合、環境変数:CLASSPATHはシステムクラスローダでロードされます。

IJSERVERのクラスの設定方法については、“J2EE ユーザーズガイド”の“J2EEアプリケーションの設計”―“クラスローダ”―“IJSERVERで使用するクラスの設定について”を参照してください。

■J2EEプロパティおよびワークユニットのクラスパスについて

クラスローダの分離が“EAR間で分離”または“すべて分離”の場合、J2EEプロパティまたはワークユニットのクラスパスに設定されたクラスはInterstageクラスローダでロードされます。

そのためJ2EEプロパティ、またはワークユニットのクラスパスとアプリケーション(EAR、ejb-jar、WAR、RAR)の両方に同じクラス名のクラスが存在する場合、そのクラスをアプリケーションから参照するとアプリケーションのクラスが使用されます。同じクラスをJ2EEプロパティ、またはワークユニットのクラスパスに設定されたクラスから参照すると、J2EEプロパティまたはワークユニットのクラスパスに設定されたクラスが使用されます。

これは何を意味するのかというと、J2EEプロパティまたはワークユニットのクラスパスに設定されたクラスとアプリケーションのクラスが物理的に同じファイルであっても、参照する場所によって別のクラスとして扱われ、ClassCastExceptionが発生する可能性を含んでいるということです。

これを回避するためには以下の手段が有効となります。

- ・ 同じクラス名のクラスが存在しないようにアプリケーションを設計してください。ワークユニットのクラスからアプリケーションのクラスを参照する必要がある場合は、カレントスレッドのコンテキストクラスローダを使用することでアプリケーションのクラスを動的にロードすることができます。
- ・ クラスローダの分離を“分離しない”としてください。この場合はワークユニットのクラスパスに設定されたクラス、アプリケーションのクラスの両方でワークユニットのクラスパスに設定されたクラスが参照されます。

■メモリ使用量について

クラスローダの分離方法に“EAR間で分離”(デフォルト設定)または“すべて分離”を選択した場合、HotDeploy機能などのクラスローダ機能の新機能を使用することができますが、EJBアプリケーションの規模に応じてメモリ使用量が変動します。

したがって、クラスローダの分離方法を変更(“分離しない”以外を選択)した場合は、メモリ使用量について再度、見積りを実施してください。

メモリ使用量の概算値は、以下の式より算出してください。

[EJBアプリケーションクラスファイルの合計サイズ]×2.0[安全係数]

■Interstage V6.0とクラスローダの構成が同じV6互換モードについて

HotDeployなどのクラスローダ機能のエンハンス機能が必要ない場合、本バージョン・レベルで作成したIJServerに、定義項目“クラスローダの分離”に“分離しない”を指定することで、クラスローダの構成をV6互換モードで動作させることができます。

■JNIを使用する場合

同じNativeモジュールは同じクラスローダ上からのみ利用可能です。別々のクラスローダで同じNativeモジュールをロードしようとした場合、java.lang.UnsatisfiedLinkErrorがスローされます。

Interstage V6.0まではEJBアプリケーションはすべてシステムクラスローダでロードされていましたが、本バージョン・レベルではクラスローダの分離の定義によってEJBアプリケーションをロードするクラスローダが分離されます。そのため、別々のEJBアプリケーションから同じNativeモジュールをロードしようとした場合、別々のクラスローダで同じNativeモジュールがロードされることになり、java.lang.UnsatisfiedLinkErrorがスローされる可能性があります。

“J2EE ユーザーズガイド”の“クラスローダ”の“クラスローダ使用時の注意事項”を参照して対処を行ってください。

6.2.11 返却される例外の詳細文字列について

Interstage Application Server V7.0(Interstage V7.0)以降では、J2EEアプリケーション運用中に開始したトランザクションのコミット処理の延長で、DBMSの更新処理中に例外が発生した場合、DBMSで発生した例外情報をJ2EEアプリケーションに返却します。そのため、トランザクションコミット時に発生したデッドロックを含め、DBMSで発生した例外の検知ができます。

ただし、返却する例外の詳細文字列が変更となるため、詳細文字列を参照しているInterstage Application Server V6.0(Interstage V6.0)以前の製品で運用していたアプリケーションを、本バージョン・レベルに移行したときに問題が発生する可能性があります。Interstage V6.0以前と同様の詳細文字列を返却するには、IJServerのワークユニット設定に以下のように指定します。なお、以下は、isj2eeadminコマンドで設定することもできます。詳細は、“リファレンスマニュアル(コマンド編)”を参照してください。

設定方法

パラメタ	値
Java VMオプション(Java Command Option)	-DCommitExSQLMsg=off

互換メッセージ内容

- javax.transaction.HeuristicRollbackException例外の詳細文字列
“STATUS_ROLLEDBACK returned from commit method of UserTransaction.”
- javax.transaction.HeuristicMixedException例外の詳細文字列
従来どおりメッセージを含めない。

6.2.12 JNDIから返却される例外について

Interstage V6.0より、J2EEアプリケーションクライアント、もしくはWebアプリケーションで行うlookup処理で、該当するオブジェクトが見つからなかった場合に返却される例外が異なります。

- Interstage V5.x以前の場合
javax.naming.NameNotFoundException例外を返却します。
- Interstage V6.0以降の場合
javax.naming.NamingException例外を返却します。

■注意

以下のようにサブコンテキストを一度獲得し、サブコンテキストに対してlookup処理を実行した場合は従来通りjavax.naming.NameNotFoundException例外が返却されます。

```
// JDBCデータソース”DB1”のlookup例
javax.sql.DataSource dataSource = null;
try {
    javax.naming.Context nctx = new javax.naming.InitialContext();
```



```
javax.naming.Context sctx = (javax.naming.Context)nctx.lookup("java:comp/env/");
dataSource = (javax.sql.DataSource)sctx.lookup("jdbc/DB1");
} catch(javax.naming.NamingException ex) { }
```

javax.naming.NameNotFoundException 例外は javax.naming.NamingException 例外のサブクラスのため lookup 処理の catch 句は javax.naming.NamingException 例外で行うことを推奨します。

6.2.13 J2EEアプリケーションの移行時のその他の注意事項

■ 配備時向け名前変換定義ファイルについて

Interstage Application Server V5.xで配備時向け名前変換定義ファイル(interstage.xml)を作成していた場合、その移行が必要になることがあります。詳細については“[4.2 Servletサービス\(Tomcat 5.5ベースのサーブレット実行環境\)への移行](#)”を参照してください。

■ Interstage Application Server V5.x以前に作成したJDBCデータソースについて

Interstage Application Server V5.x以前で作成したJDBCデータソースをバックアップ、リストア(または移入)した場合、必須項目が不足しているためInterstage管理コンソールなどで参照した定義をそのまま更新できません。
定義更新が必要な場合は、必須項目を入力してください。

■ マルチシステム機能について

拡張システムでのJ2EEアプリケーションクライアントの運用は未サポートです。また、拡張システムでfjj2eeadminコマンド、j2eejndisetupコマンドは使用できません。デフォルトシステムを使用してください。

6.3 INTERSTAGE V3.xからJ2EEアプリケーションへの移行

Interstage Application Server V3.x(Interstage V3.x)からのJ2EEアプリケーションの移行について、以下を説明します。

- ・ [運用方法の違い](#)
- ・ [J2EEアプリケーションへの移行方法](#)

■ 運用方法の違い

Interstage V3.xでの運用方法と、J2EEとしての運用方法には、以下の違いがあります。

- ・ [パッケージ化](#)
- ・ [JNDI](#)
- ・ [セキュリティ](#)

パッケージ化

Interstage V3.x

以下のように、アプリケーションごとにそれぞれ1つのアーカイブファイルとして流通することができます。

アプリケーション	アーカイブファイル
EJBアプリケーション	EJB JARファイル
Webアプリケーション	WARファイル

J2EE

以下に示すようなアプリケーションごとに作成されたアーカイブファイルを、さらにEnterprise ARchive (EAR)ファイルとしてパッケージ化することによって、運用で使用するアプリケーションすべてを1つのアーカイブファイルとして流通することができます。

アプリケーション	アーカイブファイル
EJBアプリケーション	EJB JARファイル
Webアプリケーション	WARファイル
J2EEアプリケーションクライアント	クライアントJARファイル
resource adapter	RARファイル

JNDI

Interstage V3.x

JavaアプリケーションやWebアプリケーションから、JNDIを使用してEJBやJDBCにアクセスする場合、以下のようにJNDIのサービスプロバイダを使い分ける必要があります。

- **EJBにアクセスする場合**
com.fujitsu.interstage.ejb.jndi.FJCNCtxFactoryForClient
- **JDBCにアクセスする場合**
 - Symfowareでは
com.fujitsu.symfoware.jdbc2.jndisp.SYMContextFactory
 - Oracleでは
com.sun.jndi.fscontext.RefFSContextFactory

J2EE

J2EEアプリケーションクライアントやWebアプリケーションから、JNDIを使用してEJBやJDBCにアクセスする場合、以下の設定をすることによってJNDIのサービスプロバイダを使い分ける必要はありません。

- **J2EEアプリケーションクライアントの場合**
Javaに次のパラメタを指定します。
java.naming.factory.initial=com.fujitsu.interstage.j2ee.jndi.InitialContextFactoryForClient

セキュリティ

Interstage V3.x

認証や承認を実装するには、ユーザがおのおのの手法でセキュリティの機構を実装する必要があります。

J2EE

Interstage ディレクトリサービスを使用したセキュリティ機構によって、J2EEアプリケーションのユーザを一元管理し、J2EE全体で統一された認証や承認の仕組みを実装することができます。

■J2EEアプリケーションへの移行方法

以下の場合について、Interstage V3.xからJ2EEアプリケーションに移行する方法について説明します。

- [サーブレット→EJB連携の移行](#)
- [J2EEアプリケーションクライアント→EJB連携の移行](#)

6.3.1 サーブレット→EJB連携の移行

J2EE環境に移行するための以下の手順を説明します。

- [サーブレットのソース修正](#)
- [定義の変更](#)
- [セキュリティ機能を使用する場合](#)



注意

EJBの場合、Interstage V3.xとは動作に違いがある機能が存在しますので、Enterprise Beanを開発する際には注意が必要です。詳しくは“6.5 EJBサービスの移行”を参照してください。

サープレットのソース修正

EjbClient.javaのlookup処理で指定する、参照するEJBアプリケーション名をEnterprise Bean Environment形式に変更します。

【Interstage V3.x形式のソース】

```
...
javax.naming.Context ic = new javax.naming.InitialContext();
java.lang.Object obj = (Object)ic.lookup("EjbServer");
home = (EjbServerHome)javax.rmi.PortableRemoteObject.narrow(obj, EjbServerHome.class);
...
```

【J2EE形式のソース】

```
...
javax.naming.Context ic = new javax.naming.InitialContext();
java.lang.Object obj = (Object)ic.lookup("java:comp/env/ejb/EjbServer");
home = (EjbServerHome)javax.rmi.PortableRemoteObject.narrow(obj, EjbServerHome.class);
...
```

EJBアプリケーションのlookup処理については、“J2EE ユーザーズガイド”の“JNDI”―“オブジェクトの参照方法”を参照してください。

定義の変更

Webアプリケーション環境定義ファイルを設定します。

EJBオブジェクト参照の定義として、ejb-refタグまたはejb-local-refタグを追加します。

```
...
<ejb-ref>
  <ejb-name>ejb/EjbServer</ejb-name>
  ...
</ejb-ref>
...
```

また、名前変換機能を使用する場合には“FJebeProperties.xmlファイル”に必要な名前変換の定義を指定する必要があります。オブジェクト参照機能の詳細は“J2EE ユーザーズガイド”の“JNDI”を参照してください。

セキュリティ機能を使用する場合

セキュリティ機能を使用する場合は、Webアプリケーション環境定義ファイルにセキュリティの定義を追加します。

認証方式:HTTP Basic認証

アクセスが許されるセキュリティロール:Administrator

認証の範囲:sample以下のすべてのurl

```
...
<servlet>
  ...
  <security-role-ref>
    <role-name>ADM</role-name>
    <role-link>Administrator</role-link>
  </security-role-ref>
  ...
</servlet>
```

```

. . .
<security-constraint>
  <web-resource-collection>
    <web-resource-name>sample</web-resource-name>
    <url-pattern>*/</url-pattern>
  </web-resource-collection>
  <auth-constraint>
    <role-name>Administrator</role-name>
  </auth-constraint>
</security-constraint>
. . .
<login-config>
  <auth-method>BASIC</auth-method>
  <realm-name>sample</realm-name>
</login-config>
. . .
<security-role>
  <role-name>Administrator</role-name>
</security-role>
. . .

```

セキュリティ機能の詳細は、“J2EE ユーザーズガイド”の“J2EEアプリケーションのセキュリティ”を参照してください。

6.3.2 J2EEアプリケーションクライアントーEJB連携の移行

Enterprise Bean開発を例に、J2EEに移行する場合の手順の相違点を以下に説明します。



注意

EJBの場合、Interstage V3.xとは動作に違いがある機能が存在しますので、Enterprise Beanを開発する際には注意が必要です。詳しくは“6.5 EJBサービスの移行”を参照してください。

Step1:Enterprise Beanを開発する

J2EEのセキュリティ機能を使用する場合には、セキュリティロール定義とメソッドパーミッション定義が必要です。メソッドパーミッション機能を使用することにより、J2EEアプリケーションクライアントやWebアプリケーションの認証機能と連携して、EJBアプリケーションの不当アクセスを制限することができます。J2EEのセキュリティ機能を使用する場合の運用環境の設定は、“J2EE ユーザーズガイド”の“J2EEアプリケーションのセキュリティ”を参照してください。

ここでは、データベースの変更(レコードの追加/削除)を、管理者のみが行うことができるように制御する例を説明します。Session Beanの開発でdeployment descriptorを編集する際に以下のように定義を追加します。

1. Interstage StudioのEJB Deployment Descriptorエディタを起動します。
2. タグの中から“セキュリティロール”を選択して、セキュリティロールを追加します。ここでは、管理者権限を持つセキュリティロール“Admin”を追加します。

セキュリティロール名	説明
Admin	管理者

3. リストの中からアクセスを制御する“TutorialSession”を選択し、“メソッドパーミッション”タブをクリックします。レコードを更新する“insertRecord”メソッドと、レコードを削除する“deleteRecord”メソッドに対して次のように定義を追加します。

セキュリティロール名	メソッド名	インタフェース	パラメタリスト	説明
Admin	insertRecord	Remote	int,java.lang.String	更新
Admin	deleteRecord	Remote	int	削除

上記の設定を行うことによって、メソッドに設定されたセキュリティロールに属するユーザ以外がメソッドを実行した場合には例外が返却されます。

Step2:Enterprise Beanをパッケージ化する～ejb-jarファイルの作成～

この手順には特に相違はありません。

Step3:Enterprise Beanを実行可能な状態にする

EJBアプリケーションの展開～配備についてはInterstage管理コンソールで行うことができます。
また、データソースの登録についても、Interstage管理コンソールで行います。
詳細についてはInterstage管理コンソールのヘルプを参照してください。

Step4:クライアントアプリケーションを開発する

クライアントアプリケーションのlookup処理の記述形式を変更する必要があります。また、J2EEアプリケーションクライアントの機能(リソースアクセスなど)を使用するためにdeployment descriptorファイルの作成が必要です。

1. クライアントアプリケーションの修正

Tutorial.javaのlookup処理で指定する、参照するEJBアプリケーション名をEnterprise Bean Environment形式に変更します。

【Interstage V3.x形式のソース】

```
try{
    InitialContext ic = new InitialContext();
    java.lang.Object obj = ic.lookup("TutorialSession");
    home = (TutorialSessionHome)PortableRemoteObject.narrow
        (obj, TutorialSessionHome.class);
} catch (Exception ex) {
    . . .
}
```

【J2EE形式のソース】

```
try{
    InitialContext ic = new InitialContext();
    java.lang.Object obj = ic.lookup("java:com/env/ejb/TutorialSession");
    home = (TutorialSessionHome)PortableRemoteObject.narrow
        (obj, TutorialSessionHome.class);
} catch (Exception ex) {
    . . .
}
```



ポイント

Interstage V3.xまでの記述でも動作可能ですが、J2EEアプリケーションクライアントではEnterprise Bean Environment形式でlookup処理を行うことを推奨しています。

ただし、リソース接続者管理機能を使用する場合は、必ずEnterprise Bean Environment形式で記述してください。

2. deployment descriptorの作成

リソースアクセスや名前変換機能を使用する場合にはdeployment descriptorファイルを作成する必要があります。Pure Javaアプリケーションのプロジェクトを作成後、以下のようにdeployment descriptorファイルを作成してください。

1. [ファイル]メニューの[新規作成]を選択します。
2. [新規作成]ダイアログの“deployment descriptorファイル”を選択します。
3. [deployment descriptorファイル編集]画面で必要な情報を設定し保存して終了します。

deployment descriptorの定義項目についての詳細は“J2EE ユーザーズガイド”の“J2EEアプリケーションクライアントの設定”を参照してください。

【deployment descriptor ファイルの例】

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE application-client PUBLIC "-//Sun Microsystems, Inc.//DTD J2EE Application Client1.2//EN"
'http://java.sun.com/j2ee/dtds/application-client_1_2.dtd'>

<application-client>
  <icon>
    <small-icon>smallicon.jpg</small-icon>
    <large-icon>largeicon.jpg</large-icon>
  </icon>
  <display-name>display</display-name>
  <description>Tutorial</description>
  <ejb-ref>
    <description>EJB Information</description>
    <ejb-ref-name>ejb/TutorialSession</ejb-ref-name>
    <ejb-ref-type>Session</ejb-ref-type>
    <home>TutorialSessionHome</home>
    <remote>TutorialSessionRemote</remote>
    <ejb-link>TutorialSession</ejb-link>
  </ejb-ref>
</application-client>
```

Step5:Enterprise Beanを運用する

J2EEアプリケーションクライアントの場合には、環境変数の設定、FJjndi.propertiesファイルの設定およびコマンドラインの変更が必要です。

(1)環境変数の設定

環境変数CLASSPATHに、次の値が設定されていない場合は設定してください。

Windows

```
C:¥Interstage¥J2EE¥lib¥isj2ee.jar
%ProgramFiles%¥Common Files¥FujitsuXML¥xmlpro.jar
%ProgramFiles%¥Common Files¥FujitsuXML¥xmltrans.jar
```

Solaris

Linux

```
/opt/FJSVj2ee/lib/isj2ee.jar
/opt/FJSVj2ee/lib/isj2ee.jar
/opt/FJSVxmlpc/lib/xmltrans.jar
```

(2)FJjndi.propertiesファイルの設定

名前変換機能やセキュリティ機能を使用する場合には“FJjndi.propertiesファイル”に必要な環境プロパティを指定する必要があります。このファイルに設定するユーザIDとパスワードはInterstage ディレクトリサービスに設定する必要があります。

【FJjndi.propertiesファイルの記述例】 Windows

```
FJUserID=j2ee
FJPassword=j2ee
com.fujitsu.interstage.j2ee.DeploymentDescriptorClient=C:¥work¥TutorialClient¥application-client.xml
```

【FJjndi.propertiesファイルの記述例】 Solaris Linux

```
FJUserID=j2ee
```

```
FJPassword=j2ee
com.fujitsu.interstage.j2ee.DeploymentDescriptorClient=/export/home/j2eeapl/TutorialClient/application-client.xml
```

FJjndi.propertiesファイルの詳細については、“J2EE ユーザーズガイド”の“J2EEアプリケーションクライアントの設定”を参照してください。
ユーザIDとパスワードのInterstage ディレクトリサービスへの設定方法については、Interstage ディレクトリサービスのリポジトリを構築したサーバのInterstage Application Serverのマニュアルの“ディレクトリサービス運用ガイド”を参照してください。

(3)コマンドラインの変更

クライアントアプリケーションを実行するコマンドラインを以下のように変更します。

【Interstage V3.xの場合】

```
java -Djava.naming.factory.initial=com.fujitsu.interstage.ejb.jndi.FJCNCtxFactoryForClient Tutorial
```

【J2EEの場合】

```
java -Djava.naming.factory.initial=com.fujitsu.interstage.j2ee.jndi.InitialContextFactoryForClient Tutorial
```

6.4 Servletサービスの移行

ここでは、Servletサービスの移行について説明します。

■リクエストのURIに¥を含む場合

リクエストのURIに¥(エンコードされている場合も該当)を含む場合、以下の製品では呼び出し可能であったコンテンツが呼び出しできなくなることがあります。リクエストのURIに¥を含まないようにアプリケーションを修正してください。

- Windows Server(R) 2003 for Itanium-based Systems以外のWindows製品上で動作するInterstage Application Server/Interstage Web Server V9.0

■セッションIDの長さについて

Interstage Application Server 8.0以降では、セッションIDの長さが長くなっています。

セッションIDの一意性の範囲、桁数はServletサービスの実装により異なる可能性があります。セッションIDをセッション識別以外の目的で使わないようにしてください。

■サーブレット実行環境の変更

Interstage Application Server/Interstage Web Server V9.0から、Tomcat 5.5ベースのサーブレット実行環境(Servletサービス)を提供しています。

本製品のV6.0から提供しているTomcat 4.1ベースのサーブレット実行環境(Servletサービス)は互換用として本製品でも提供しています。

■isj2eeadminコマンドを使用して環境を構築する場合

Interstage Application Server/Interstage Web Server V9.0から、IJServer定義ファイルとWebサーバコネクタ定義ファイルにWebサーバ名の指定が必要になりました。

Interstage Application Server 8.0から抽出したファイルを使用するには、これらのファイルにWebサーバ名の指定を行ってからisj2eeadminコマンドで適用してください。Webサーバ名の指定については、“リファレンスマニュアル(コマンド編)”の、“IJServer定義ファイル”および、“Webサーバコネクタ定義ファイル”を参照してください。

■以前のバージョンで発生した障害修正の追加定義

本ソフトウェアでは、互換性維持のために追加定義により修正前後の動作を変更可能な障害修正が含まれています。

ただし、以降に示す障害修正については修正後の動作がデフォルトになっているため、以前のバージョンで追加定義を設定していない場合は動作が変わります。

旧バージョンから移行した環境でも修正前の動作で運用する場合、Interstage管理コンソールまたはisj2eeadminコマンドで追加定義を設定してください。

追加定義は以下で設定します。

Interstage管理コンソール:

[ワークユニット] > “ワークユニット名” > [環境設定] タブ > [詳細設定] > [ワークユニット設定] > [JavaVMオプション]

isj2eeadminコマンド:

IJServer定義ファイルの下記タグの値

/Isj2eeIjserverDefinition/IJServer/Common/JavaCommandOptions

設定方法の詳細については「Interstage管理コンソールヘルプ」または「リファレンスマニュアル(コマンド編)」を参照してください。

以下は、修正後の動作がデフォルトになっている障害修正の追加定義です。

Tomcat5.5ベースのServletサービス

PG62294:

書式:

-Dcom.fujitsu.interstage.jservlet.api.getRemoteHost.PG62294=true|false
(省略値は"true")

true : WebサーバでDNS逆引きに成功した時、getRemoteHostはクライアント端末のリモートホスト名を返却します。

false : getRemoteHostは常にクライアント端末のIPアドレスを返却します。(修正前の動作)

Tomcat4.1ベースのServletサービス

PG58017:

書式:

-Dcom.fujitsu.interstage.j2ee.ijserver.request.rd.compliant=true|false
(省略値は"true")

true : getRequestDispatcherメソッドの引数を、リクエストURLからの相対パスと見なして処理します。

false : getRequestDispatcherメソッドの引数を、ServletPathからの相対パスと見なして処理します。(修正前の動作)

6.5 EJBサービスの移行

ここでは、EJBサービスについて、以下を説明します。

- [Interstage Application Server V9.0.1/V9.1での変更内容](#)
- [Interstage Application Server V9.0での変更内容](#)
- [Interstage Application Server 8.0での変更内容](#)
- [Interstage Application Server V7.0での変更内容](#)
- [Interstage Application Server V6.0での変更内容](#)
- [EJBアプリケーション移行時の注意点](#)
- [Interstage Application Server V5.x以前からの移行](#)

6.5.1 Interstage Application Server V9.0.1/V9.1での変更内容

Interstage Application Server V9.0.1(以降、Interstage V9.0.1)(RHEL-AS4(x86)、RHEL-AS4(EM64T)、RHEL5(x86)、RHEL5(Intel64)の場合)、Interstage Application Server V9.1(以降、Interstage V9.1)(Windows、Solaris、RHEL-AS4(IPF)、RHEL5(IPF)の場合)の変更内容を説明します。

- [\[アプリケーション環境定義\]の定義項目について](#)
- [Message-driven Beanの初期起動インスタンス数について](#)

■[アプリケーション環境定義]の定義項目について

Interstage V9.0.1/V9.1では、EJB2.1に準拠しました。

EJB2.1に準拠したEJBアプリケーションの場合には、[システム] > [ワークユニット] > [IJSERVER名] > [EJBモジュール名] > [EJBアプリケーション名] > [アプリケーション環境定義]で表示していた以下の定義については表示されません。

内容を参照したい場合は、[システム] > [ワークユニット] > [IJSERVER名] > [EJBモジュール名]で表示されるdeployment descriptorの内容を確認してください。また、定義値を変更したい場合は、開発環境で生成されるejb-jar.xmlの内容を変更してください。

非表示となる定義情報	deployment descriptorファイルの対応するタグ
Enterprise Bean名	ejb-name
Homeインタフェース名	home
Remoteインタフェース名	remote
LocalHomeインタフェース名	local-home
Localインタフェース名	local
Enterprise Bean クラス名	ejb-class
セッション種別	session-type
トランザクション管理種別	transaction-type
Persistence タイプ	persistence-type
抽象スキーマ名	abstract-schema-name
Primary Keyクラス名	prim-key-class
Primary Keyフィールド名	primkey-field
リエントラント	reentrant
メッセージセレクトラ	message-selector
Destinationタイプ/サブスクライバの永続性	message-driven-destination
環境プロパティ	env-entry
参照EJB	ejb-ref
参照LocalEJB	ejb-local-ref
参照リソース	resource-ref
参照環境リソース	resource-env-ref
トランザクション	container-transaction
参照セキュリティロール	security-role-ref
メソッドパーミッション	method-permission
クエリ定義	query

また、以下の説明についても非表示となりました。

内容を参照したい場合は、対応するdeployment descriptorファイルのタグを参照してください。

非表示となる定義情報	deployment descriptorファイルの対応するタグ
セキュリティアイデンティティの説明	security-identityタグのdescriptionタグ



ポイント

従来のJ2EE1.3ベースで構築されたアプリケーションは従来どおりに運用可能です。J2EE1.3とJ2EE1.4のアプリケーションを同一のIJSERVER上で混在して運用することも可能です。

■Message-driven Beanの初期起動インスタンス数について

Message-driven Beanの初期起動インスタンス数のデフォルト値が8となります。Interstage Application Server V9.0.0以前は1でした。初期起動インスタンス数は同時に処理するメッセージ数です。デフォルト値が変更されたことで、初期設定でのメッセージ処理性能が向

上します。

初期起動インスタンス数を変更する場合にはInterstage管理コンソールを使用してEJBアプリケーションのアプリケーション環境設定で変更してください。また、ejbdefimportコマンドでも変更できます。

6.5.2 Interstage Application Server V9.0での変更内容

Interstage Application Server V9.0(以降、Interstage V9.0)での変更内容を説明します。

■Message-driven Bean使用時のイベントチャネルの停止について

Message-driven Beanを使用し、JMSメッセージングモデルとトランザクション管理種別が下記表の○の場合、従来はイベントチャネルが停止するとIJSerVerプロセスを停止していました。しかし、IJSerVerを停止すると他の業務にも影響があるため、Interstage V9.0以降ではメッセージIJSerVer21457を出力し、IJSerVerプロセスの停止は行いません。

JMSメッセージングモデル	トランザクション管理種別		
	Bean	Container	
		トランザクション属性	
		Required	NotSupported
Point-To-Pointモデル	—	—	—
Publish/Subscribeモデル	○	○	—

6.5.3 Interstage Application Server 8.0での変更内容

Interstage Application Server 8.0(以降、Interstage 8.0)での変更内容を説明します。

- [EJBのプロセスディスパッチ方式の変更](#)
- [STATELESS Session Beanのインスタンス生成モード](#)
- [Message-driven Beanのスレッドプール](#)
- [EJBサービス運用コマンド](#)
- [Message-driven Bean機能拡張](#)

■EJBのプロセスディスパッチ方式の変更

Interstage Application Server 8.0でEJBコンテナのディスパッチ方式が改善され、ラウンドロビン方式で動作します。Interstage Application Server V7.0はLRU方式で動作しました。

以下に各プロセスディスパッチ方式について説明します。ラウンドロビン方式の場合には各プロセスに処理が均等に割り振られるため、処理が集中した場合の性能改善効果や危険分散効果があります。

プロセスディスパッチ方式	ディスパッチ論理
ラウンドロビン方式 (8.0のディスパッチ方式)	要求待ちのサーバアプリケーションプロセスの中で、最初に要求待ちとなったプロセスに、クライアントからの要求メッセージを振り分けます。 クライアントからの要求数が少ない場合は、各プロセスに均等に要求メッセージが振り分けられます。
LRU方式 (7.0までのディスパッチ方式)	要求待ちのサーバアプリケーションプロセスの中で、最後に要求待ちとなったプロセスに、クライアントからの要求を振り分けます。 クライアントからの要求数が少ない場合は、特定のプロセスのみに要求メッセージが振り分けられます。

■STATELESS Session Beanのインスタンス生成モード

以下の条件を満たす場合、STATELESS Session Beanのインスタンス生成タイミングのチューニングが行えません。

- Interstage Application Server V6.0またはV7.0を使用する

- Enterprise Bean定義情報のSTATELESS他Beanアクセス参照機能に“**At Start-Up**”を指定しているアプリケーションを、8.0に移行する

この場合、**At Start-Up**定義が有効となり、インスタンス生成タイミングのチューニングが行えません。

インスタンス生成タイミングのチューニングを行いたい場合は、**ejbdefexport/ejbdefimport**コマンドを使用し、<stateless-instance-create-type?>タグを削除するか、または、タグの値を“**At First Access**”に変更して下さい。

ejbdefexport/ejbdefimportコマンドについては、“J2EE ユーザーズガイド”の“Enterprise Bean定義情報の**export**と**import**”、および“リファレンスマニュアル(コマンド編)”の“EJBサービス運用コマンド”を参照してください。

■Message-driven Beanのスレッドプール

Interstage Application Server 8.0では、Message-driven Beanはスレッドをプーリングすることが可能となるため、以下を見直してください。

- Message-driven Beanの同時処理数の最大値が、デフォルトで64となります。
デフォルトで65以上のメッセージがIJServerプロセスに配信されると、使用中のスレッドがプールに返却されるまで待機します。
プールに返却されるまで待機させない場合には、各Message-driven Beanの初期起動インスタンス数の合計値を、Interstage管理コンソールまたはisj2eeadminコマンドで、Message-driven Beanの同時処理数の最大値に指定してください。
- Message-driven Bean、またはMessage-driven Beanから呼び出すアプリケーションでスレッド変数などを使用していた場合には、動作が変わる可能性があります。
スレッド変数を使用したアプリケーションを使用している場合には、スレッド変数を使用しないように変更するか、アプリケーション開始または終了時にスレッド変数をクリアするように、アプリケーションを変更してください。
アプリケーションの変更ができない場合には、以下のオプションを設定して、スレッドプールを使用しないように設定してください。
FJJBconfig.propertiesを編集した場合は、設定を有効にするためIJServerを再起動してください。



スレッドプールを使用しないよう設定した場合、Interstage管理コンソールのIJServerモニタ画面の「Message-driven Beanスレッド情報」に表示される値は更新されません。

項目	設定内容
定義ファイル格納ディレクトリ	Windows Interstage インストールディレクトリ¥ejb¥etc Solaris Linux /opt/FJSVejb/etc
定義ファイル名	FJJBconfig.properties
指定するキー	"mdb.thread.pool"
指定する値	- ON(デフォルト) Message-driven Beanでスレッドプールを使用する。 - OFF Message-driven Beanでスレッドプールを使用しない。 大文字・小文字は区別しません。

■EJBサービス運用コマンド Solaris Linux

Interstage Application Server 8.0では、アプリケーションファイルの権限を指定する機能がサポートされたため、以下のコマンドはアプリケーションファイル保護レベルで指定したユーザで使用してください。

アプリケーションファイル保護レベルについては、“J2EE ユーザーズガイド”の“J2EEアプリケーションの配備と設定”を参照してください。

- ejbdefexport
- ejbdefimport

■Message-driven Bean機能拡張

◆異常時のメッセージ退避機能

以下の条件を満たす場合、リトライ回数を超過してもメッセージ受信を繰り返す可能性があります。

- Point-To-Pointメッセージングモデル(1受信者だけにメッセージを配信するモデル)
- 不揮発化チャネルを利用

イベントチャネルのイベントデータをメモリにキャッシュする数を、イベントチャネルに蓄積できるイベントデータの最大値よりも大きく設定してください。

設定はイベントサービス運用コマンドを使用して行います。詳細は“リファレンスマニュアル(コマンド編)”の“esetcnf”および“esetcnfchnl”を参照してください。

Point-To-Pointメッセージングモデルを利用する場合、以下の非互換があります。

- V6.0以前、またはWindows(R)のV7.0のInterstage JMSサービスを使用してノーティフィケーションサービスのイベントチャネルに送信されたメッセージは、リトライ回数を超過しても退避されずにメッセージ受信を繰り返します。
- setRollbackOnlyメソッドを実行してトランザクションをロールバックした場合、V6.0(Windows(R)の場合はV7.0)以前のInterstageではメッセージが退避されませんでしたが、今版よりsetRollbackOnlyメソッドを実行してトランザクションをロールバックした場合もメッセージが退避されます。

上記に該当する場合は、以下の互換オプションを設定して運用してください。ただし、互換オプションを設定した場合、JMSが受信したメッセージは同時に処理されません。

設定ファイルの格納先

Windows

C:\Interstage¥EJB¥etc¥FJEJBconfig.properties

Solaris

Linux

/opt/FJSVejb/etc/FJEJBconfig.properties

設定内容

MDB_RecvType=reciver

6.5.4 Interstage Application Server V7.0での変更内容

Interstage Application Server V7.0(以降、Interstage V7.0)での変更内容を説明します。

- 規約非準拠のEJBアプリケーションの配備について
- STATEFUL Session Beanの無通信監視時間
- finderメソッドの復帰値collectionの各種APIサポート
- IPCOMとの連携
- EJB QL拡張
- セッションタイムアウト機能・トランザクションタイムアウト機能
- 同一EJBアプリケーション名の配備

■規約非準拠のEJBアプリケーションの配備について

Interstage Application Server V7.0以降では、EJBアプリケーションを配備する際、EJB規約に準拠しているかのチェックが緩和されています。

以下が、V7.0以降で運用可能となるEJB規約の制限です。

No.	Bean種別	チェック対象	チェック内容
1	Enterprise Beanクラス	すべて	ビジネスメソッドがstaticで定義されている。

No.	Bean種別	チェック対象	チェック内容
2	Enterprise Beanクラス	Entity Bean	Enterprise Beanクラスに定義されているejbCreateメソッドの復帰値とdeployment descriptorに設定されている<prim-key-class>は一致しているが、Enterprise Beanクラスの親のクラスに定義されているejbCreateメソッドの復帰値が<prim-key-class>と一致しない。
3	Enterprise Beanクラス	BMP Entity Bean	Enterprise Beanクラスに定義されているejbFind<~>メソッドのthrows句にある例外が、Homeインタフェースに定義されているfind<~>のthrows句にない。
4	Enterprise Beanクラス	Message-driven Bean	ejbCreateメソッドのthrows句にjavax.ejb.CreateExceptionがある。
5	Enterprise Beanクラス	CMP2.0 Entity Bean	ejbCreateメソッドのthrows句にjavax.ejb.CreateExceptionがない。
6	Primary Key クラス	CMP Entity Bean	deployment descriptorに<primkey-field>が指定されておらず、Primary Keyクラスにpublicではないフィールドがある。
7	Primary Key クラス	CMP Entity Bean	deployment descriptorに<primkey-field>が指定されておらず、<cmp-field>にないフィールドがPrimary Keyクラスに定義されている。
8	Homeインタフェース	Session Bean Entity Bean	Createメソッドのthrows句にjavax.ejb.CreateExceptionがない。
9	deployment descriptor	CMP2.0 Entity Bean	<multiplicity>に“one”または“many”を指定している

■STATEFUL Session Beanの無通信監視時間

無通信監視時間のデフォルト値が、30分に変更になりました。30分以上、STATEFUL Session Beanにアクセスしないアプリケーション構成の場合は、以下のエラーが発生する可能性があるため、無通信監視時間を大きく設定してください。

```
java.rmi.NoSuchObjectException: CORBA OBJECT_NOT_EXIST 0 No; nested
exception is: org.omg.CORBA.OBJECT_NOT_EXIST: CORBA_Request_get_response
minor code: 0 completed: No
```

無通信監視時間は、Interstage管理コンソールの[ワークユニット] > [IJServer名] > [EJBアプリケーション]からSTATEFUL Session Beanを選択して、Interstage拡張情報の“無通信監視時間”に設定してください。

■finderメソッドの復帰値collectionの各種APIサポート

Interstage Application Server V7.0より、Entity Beanの復帰値Collection型finderメソッドを実行して取得したjava.util.Collectionオブジェクトに対して、java.util.Collectionインタフェースに定義されているすべてのメソッドが使用可能となりました。



CMP1.1およびBMPで使用する場合には、以下の条件があります。以下の条件で使用しない場合は、V6.0以前の範囲になります。CMP2.0では無条件に使用できます。

- IJServerのタイプが、WebアプリケーションとEJBアプリケーションを同一JavaVMで運用
トランザクション属性に“Mandatory”以外のトランザクション属性を指定して下さい。
- IJServerのタイプが、WebアプリケーションとEJBアプリケーションを別JavaVMで運用 または、EJBアプリケーションのみ運用
トランザクション属性にMandatory以外のトランザクション属性を指定するか、トランザクション属性に“Mandatory”を指定する場合は、Interstage管理コンソールの[アプリケーション環境]より、「ローカル呼出し」に“しない”を設定して下さい。

■IPCOMとの連携

Interstage Application ServerでIPCOMと連携する場合、V6.0ではEJBアプリケーションを配備した後に、配備したアプリケーションごとにオブジェクトリファレンスの再登録(オブジェクトリファレンスの確認/削除/再登録)が必要でした。

V7.0以降ではIJServerに負荷分散ホスト名を指定するだけで、EJBアプリケーションの配備時に自動的に環境設定を行いますので、EJBアプリケーションごとの操作は不要です。また、配備済みのEJBアプリケーションに対して、「IJServerの環境設定」の内容を更新するだけで環境設定ができます。

IJServer単位で設定した場合には、非互換はありません。

EJBアプリケーションごとにIPCOMと連携する場合、V6.0と同様にCORBAサービスのコマンドを使用して、アプリケーションの配備後にオブジェクトリファレンスを再登録する手順で設定を行ってください。

1. `odlistns -l`コマンドを実行してオブジェクトリファレンス情報を表示して、負荷分散対象のEJBアプリケーションのインタフェースリポジトリ名とインプリメンテーションリポジトリ名を確認します。
以下にオブジェクトリファレンスの詳細情報の見方を示します。

Name (Type)	Object information (detail)
	Default object information (detail)
EasyBean (o)	RMI:pkgTest.EasyBeanHome:0000000000000000,
(a)	(b)
	IDL:com.fujitsu.interstage.j2ee.ijserver/IJServer001:1.0,
	(march:8002:1.1:UNICODE (UCS2))
	(c)

- a) EJBアプリケーション名
- b) インタフェースリポジトリ名
- c) インプリメンテーションリポジトリ名

2. ネーミングサービスに登録されたオブジェクトリファレンスを削除します。
以下のコマンドを実行して負荷分散したいEJBアプリケーションのオブジェクトリファレンスを削除してください。

```
OD_or_adm -d -n [EJBアプリケーション名]
```

3. オブジェクトリファレンスを登録します。
以下のように仮想IPアドレスのホスト名を指定して、1.で確認した情報を指定してオブジェクトリファレンスを再登録します。

```
OD_or_adm -c [インタフェースリポジトリID] -a [インプリメンテーションリポジトリID] -h [仮想IPアドレスのホスト名] -n [アプリケーション名] -p [ポート番号]
```



例

```
OD_or_adm -c RMI:pkgTest.EasyBeanHome:0000000000000000 -a  
IDL:com.fujitsu.interstage.j2ee.ijserver/IJServer001:1.0 -h vhost -p 8002 -n EJB001  
(vhostは、仮想IPアドレス)
```



注意

オブジェクトリファレンスの再登録後、Interstage管理コンソールの[ワークユニット] > [IJServer名] > [環境設定]からIJServerの更新を行った場合には、再度オブジェクトリファレンスの再登録を行ってください。

■EJB QL拡張

Interstage Application Server V7.0以降でEJB QLを使用する場合、デフォルトでV6.0のEJB QLの機能を拡張したEJB QL拡張が設定されています。

【EJB QL拡張を使用しない場合】

拡張機能を使用しない(EJB2.0規約の範囲内で運用する)場合は、Interstage管理コンソールの[システム] > [環境設定] の 詳細設定の下にある[EJBサービス詳細設定]を参照し、“CMP2.0のEJB QL拡張”の定義項目を“使用しない”に変更してください。

EJB QL拡張を使用しない場合、以下の拡張機能が制限されます。

- 比較述語、BETWEEN述語の値式にjava.lang.Stringが指定できる
- LIKE述語のパターンに変数が指定できる
- LIKE述語のエスケープ文字に変数が指定できる
- NULLに変数が指定できる
- INに変数と数値が指定できる
- ORDER BY句が指定できる
- 演算子MODが指定できる
- 集合関数(MIN、MAX、AVG、SUM、COUNT)が指定できる

■セッションタイムアウト機能・トランザクションタイムアウト機能

Interstage V7.0以降では、セッションタイムアウト機能、トランザクションタイムアウト機能は使用できません。

これらの機能は、Interstage V6.0以前では非推奨機能であり旧資産との互換は保証されているため、代替機能への移行を推奨します。代替機能への移行方法については、[■非推奨機能](#)を参照してください。

■同一EJBアプリケーション名の配備

1つのIJSERVERに同じJNDI名のEJBアプリケーションが配備されないように、EJBアプリケーション名の重複チェックを行います。

IJSERVERのタイプが「WebアプリケーションとEJBアプリケーションを同一JavaVMで運用」の場合、V6.0では異なるモジュールでも上書きしていましたが、V7.0以降ではモジュールが異なる場合に上書きを行わないようチェックを行います。

6.5.5 Interstage Application Server V6.0での変更内容

Interstage Application Server V6.0(以降、Interstage V6.0)での変更内容を説明します。

- [デフォルト設定の追加](#)
- [非推奨機能](#)
- [同時処理数について](#)
- [性能オプションについて](#)
- [STATELESS Session Beanのインスタンス数について](#)
- [ログ出力について](#)
- [EJBアプリケーション名について](#)

■デフォルト設定の追加

旧バージョンでMessage-driven Beanを運用する場合、EJBアプリケーション環境定義にJMSコネクションファクトリ名とDestination名の定義が必須でした。

8.0では、JMSコネクションファクトリ名とDestination名を定義しなかった場合、以下のデフォルト設定で動作します。

- JMSコネクションファクトリ名
 - Topicの場合:TopicCF001
 - Queueの場合:QueueCF001
- Destination名
 - EJBアプリケーション名をデフォルトのDestination名とします。

また、旧バージョンにおいて、トランザクション管理種別に“Container”が指定されていてトランザクション属性が設定されていないEJBアプリケーションは、運用することができませんでした。

8.0では、コンテナが自動的に“Required”が指定されたものとしてトランザクションの制御を行います。

■非推奨機能

以下の機能群は、V6より非推奨機能となりました。いずれも旧資産との互換は保証されていますが、次期バージョンでは提供されない可能性があるため、代替機能への移行を推奨します。

なお、これらの機能に対する定義操作は、Interstage管理コンソールではサポートされていません。

- セッションタイムアウト機能

STATEFUL Session Beanの無通信監視機能をサポートしたことにより、セッションタイムアウト機能は非推奨機能となっています。

Interstage管理コンソールからは定義できません。

以前のバージョン・レベルでセッションタイムアウト機能を使用し、本バージョン・レベルで引き続き使用する場合、以前のバージョン・レベルにてセッションタイムアウト機能を使用していたEJBアプリケーション実行環境をejbdefexportコマンドを使用して移出し、本バージョン・レベルへejbdefimportコマンドで移入してください。

詳細は“J2EE ユーザーズガイド”の“運用コマンドを使用してカスタマイズする方法”を参照してください。

下位互換性のために、旧資産をそのまま使用する場合には引き続きサポートされます。

代替機能である“STATEFUL Session Beanの無通信監視機能”については、“J2EE ユーザーズガイド”の“STATEFUL Session Beanの無通信監視機能”を参照してください。



セッションタイムアウト機能を使用する場合は、STATEFUL Session Beanの無通信監視機能を使用することはできません。

- トランザクションタイムアウト機能

EJBのトランザクションタイムアウト機能は非推奨機能となっています。

Interstage管理コンソールからは定義できません。

以前のバージョン・レベルでトランザクションタイムアウト機能を使用し、本バージョン・レベルで引き続き使用場合は、以前のバージョン・レベルにてトランザクションタイムアウト機能を使用していたEJBアプリケーション実行環境をejbdefexportコマンドを使用して移出し、本バージョン・レベルへejbdefimportコマンドで移入してください。

詳細は“J2EE ユーザーズガイド”の“運用コマンドを使用してカスタマイズする方法”を参照してください。

トランザクションタイムアウトの代わりにワークユニットの時間監視機能を使用してください。

ワークユニットの時間監視をIJSERVERワークユニット作成時に設定する場合は、Interstage管理コンソールを使用する場合、[詳細設定] > [ワークユニット設定]で行います。また作成済みのIJSERVERワークユニットに対して設定する場合は、[ワークユニット] > [環境設定] > [ワークユニット設定]から“アプリケーション最大処理時間”を設定してください。

Interstage管理コンソールの詳細については、Interstage管理コンソールのヘルプを参照してください。

isj2eadminコマンドを使用する場合は、“リファレンスマニュアル(コマンド編)”の“isj2eadminコマンド”を参照して設定してください。

- 最大メモリ量の設定

IJSERVERの最大メモリ量をIJSERVERワークユニット作成時に設定する場合は、Interstage管理コンソールを使用する場合、[詳細設定] > [ワークユニット設定]で行います。また作成済みのIJSERVERワークユニットに対して設定する場合は、[ワークユニット] > [環境設定] > [ワークユニット設定]から“JavaVMオプション”において以下を指定してください。

Interstage管理コンソールの詳細については、Interstage管理コンソールのヘルプを参照してください。

isj2eadminコマンドを使用する場合は、“リファレンスマニュアル(コマンド編)”の“isj2eadminコマンド”を参照して設定してください。

-Xmx[最大メモリ量]m

V5で指定可能であったワークユニット定義項目のEJBアプリケーション最大メモリ量(Maximum Memory for EJB Application)は、ワークユニットの設定項目に表示されません。

■同時処理数について

V5で提供されたIJSERVERでは同時処理数のデフォルトは40でしたが、8.0で提供されたIJSERVERでは同時処理数の最大値が64/最小値が16で動作します。

同時処理数の最大値/最小値は、以下で行います。

- Interstage管理コンソールの[ワークユニット] > [EJBコンテナ設定]の“同時処理数”
- isj2eadminコマンドの“ejb”タグ
isj2eadminコマンドを使用する場合は、“リファレンスマニュアル(コマンド編)”の“isj2eadminコマンド”を参照して設定してください。

■性能オプションについて

V5でサポートしていた以下の性能オプションについては、デフォルトで動作します。
Interstage管理コンソールで指定する必要はありません。

- EJBオブジェクトの共用
- 複数レコードの一括更新

■STATELESS Session Beanのインスタンス数について

STATELESS Session Beanのインスタンス数を定義する必要がなくなりました。

STATELESS Session Beanのインスタンスは、STATELESS Session Beanへの初回アクセス時に“同時処理数の最大値”に指定された値の数だけ作成されます。

■ログ出力について

V5で各ファイルに出力されていた以下の情報については、IJServerのログ(Windows(R)システムの場合、J2EE共通ディレクトリ¥ijserver¥[IJServer名]¥log配下のファイル、Solaris/Linuxシステムの場合、J2EE共通ディレクトリ/ijserver/[IJServer名]/log配下のファイル)に出力されます。

なお、従来から出力しているイベントログのメッセージについては、従来通りイベントログにもメッセージが出力されます。

- アプリケーションの標準出力、標準エラー出力
- EJBコンテナログ
- EJBコンテナのエラーメッセージ
- EJBのスナップ出力
- connectorのログ出力 (注1)

注1) connectorのログ出力先を、従来と同じ場所に出力する場合は、以下を設定してください。

定義ファイル格納ディレクトリ	Windows Interstageインストールディレクトリ¥J2EE¥etc¥jca Solaris Linux /opt/FJSVj2ee/etc/jca
定義ファイル名	jca.properties
指定するキー	“log.file.option”
指定する値	“V5”を指定します。



例

log.file.option=V5

■EJBアプリケーション名について

EJBアプリケーション名に“:”を使用することはできません。V5で“:”を使用していた場合には、配備時にEJBアプリケーション名を変更してください。

なお、名前を変更したEJBアプリケーション名を、アプリケーション名を変更せずにlookupする場合には、名前変換機能を使用してください。

6.5.6 EJBアプリケーション移行時の注意点

■本バージョン・レベルで提供する追加機能を使用する場合

旧バージョン・レベルで開発したEJBアプリケーションを、動作させるためには、以下の作業が必要となります。

- Interstage Application Server V4.x以前からの移行時
EJB2.0規約またはEJB2.1規約に準拠したEJBアプリケーションに変更する場合は、deployment descriptorをEJB2.0またはEJB2.1の形式にする必要があります。

6.5.7 Interstage Application Server V5.x以前からの移行

Interstage Application Server V5.x以前からの移行については、“[4.4 EJBサービス\(IJServer\)への移行方法](#)”を参照してください。

6.6 Interstage JMSの移行

Interstage JMSは、以下の製品で利用可能です。

- Interstage Application Server Enterprise Edition
- Interstage Application Server Standard-J Edition

ここでは、Interstage JMSについて、以下を説明します。

- [Interstage Application Server V9.0での変更内容](#)
- [Interstage Application Server V7.0での変更内容](#)



注意

Interstage Application Server V5.x以前からInterstage JMSを移行する場合、イベントサービス運用コマンドの使用について注意すべき点があります。“[7.5 イベントサービスの移行](#)”を参照して確認してください。

6.6.1 Interstage Application Server V9.0での変更内容

Interstage Application Server V9.0(以降、Interstage V9.0)での変更内容を説明します。

■JMS1.0.2規約からJMS1.1規約への変更について

Interstage 8.0以前までJava Message Service 1.0.2規約をサポートしていましたが、Interstage V9.0以降においてJava Message Service 1.1規約をサポートするように変更となりました。

Java Message Service 1.1規約では、Java Message Service 1.0.2規約に従ったアプリケーションも正常に動作することを保証していますが、Interstage V9.0以降でJMS1.0.2規約のアプリケーションを実行すると、Interstage 8.0以前の環境で実行した場合とエラーチェックおよび例外情報が異なることがあります。

Interstage 8.0以前の環境で作成したアプリケーションを、Interstage V9.0以降の環境で同様の動作で実行する場合は、アプリケーション実行時にシステムプロパティに以下のように指定してください。

[システムプロパティ名]

com.fujitsu.interstage.jms.exception_version_lower

[指定方法]

“yes”を設定します。

```
-Dcom.fujitsu.interstage.jms.exception_version_lower=yes
```

6.6.2 Interstage Application Server V7.0での変更内容

Interstage Application Server V7.0(以降、Interstage V7.0)での変更内容を説明します。

- [マルチサーバ管理機能を使用する場合](#)
- [V7.0以降のアプリケーションとV6.0以前のアプリケーションを混在して運用する場合](#)

- [Interstage Application Server Standard-J Editionのインストール直後のセットアップ状態について](#)
- [EJBのMessage-driven Bean使用時の接続コンシューマ数について](#)
- [メッセージの出力について](#)

■マルチサーバ管理機能を使用する場合

Interstage V7.0以降においてマルチサーバ管理機能を使用する場合、Interstage JMSの構成情報、保存先およびイベントチャネルの操作は、スタンドアロンの操作と異なります。マルチサーバ管理機能を使用する場合の操作方法については、“運用ガイド(基本編)”の以下を参照してください。

- Interstageの環境構築を行う場合
“マルチサーバ管理機能” – “Interstageの環境構築” – “J2EEリソース(JDBC、connector、JMS、JavaMail)の作成”
- Interstageの運用を行う場合
“マルチサーバ管理機能” – “Interstageの運用” – “J2EEリソース(JDBC、connector、JMS、JavaMail)の運用”
- Interstageの構成変更を行う場合
“マルチサーバ管理機能” – “Interstageの構成変更” – “J2EEリソース(JDBC、connector、JMS、JavaMail)の構成変更”

■V7.0以降のアプリケーションとV6.0以前のアプリケーションを混在して運用する場合

V7.0以降の送信用アプリケーションからV6.0以前の受信用アプリケーションに対してメッセージを送信することはできません。本運用を行った場合、V6.0以前のアプリケーションにおいてエラーメッセージjms2510が出力されて、異常終了してしまいます。

V7.0以降の送信用アプリケーションからV6.0以前の受信用アプリケーションに対してメッセージを送信する場合は、V7.0以降のシステムで環境変数JMS_RECEIVER_VERSION_LOWERに“ON”を設定してください。

■Interstage Application Server Standard-J Editionのインストール直後のセットアップ状態について **Windows**

Interstage 8.0では、Interstage Application Server Standard-J Editionのインストール直後のセットアップ状態が変更されています。

- V6.0(変更前)
Interstage管理コンソールの[システム] > [環境設定]で[JMS詳細設定]に“使用する”が指定され、システム構成サービスにイベントサービスが登録されていました。
- 8.0(変更後)
Interstage管理コンソールの[システム] > [環境設定]で[JMS詳細設定]に“使用しない”が指定され、システム構成サービスにイベントサービスが登録されていません。
したがって、Interstage JMSを使用する場合は、[システム] > [環境設定]で[JMS詳細設定]を“使用する”に変更し、システム構成サービスにイベントサービスを登録してください。

■EJBのMessage-driven Bean使用時の接続コンシューマ数について **Solaris** **Linux**

Interstage V7.0以降では、EJBのMessage-driven Beanを使用している場合、イベントチャネルの接続コンシューマ数に以下の値が加算されます。

- Point-To-Pointメッセージングモデル
通信状態により、接続コンシューマ数が初期値から最大値まで増加します。

接続コンシューマ数(初期値) = ワークユニット(IJServer)の[プロセス多重度] ×
Message-driven Beanの[初期起動インスタンス数(同時実行スレッド数)]

接続コンシューマ数(最大値) = ワークユニット(IJServer)の[プロセス多重度] ×
Message-driven Beanの[初期起動インスタンス数(同時実行スレッド数)] × 2

- Publish/Subscribeメッセージングモデル

接続コンシューマ数 = 1

■メッセージの出力について

Solaris

Interstage V7.0以降において、日本語環境でシステムログに出力されるメッセージを日本語から英語に変更しています。

Linux

Interstage V6.0以降において、日本語環境でシステムログに出力されるメッセージを日本語から英語に変更しています。

以前のバージョン・レベルと同様、日本語のメッセージを出力する場合は、以下のJava(TM)VMのシステムプロパティに設定を行ってください。

プロパティ名	プロパティ値
com.fujitsu.interstage.jms.message_format	“V5”を指定します。 V5を指定した場合、日本語のメッセージが出力されます。 注 UTF-8環境の場合、“V5”は指定できません。

プロパティ値が誤っている場合は、変更後の形式でメッセージが出力されます。

6.7 Interstage Webサービスの移行

■Interstage Application Server 8.0から移行する場合の注意事項

◆非ascii文字の通信形式

Interstage Application Server 8.0のInterstage Webサービスでは、通信における文字列データ中の非ascii文字は、XMLの文字参照形式に変換されてascii文字を用いた形式で送信されていました。

Interstage Application Server V9.0からは、非ascii文字も文字参照形式に変換せずそのまま送信されます。

XML規約に従っている限り、受信側でこの変更によるデータの違いは発生しません。

Interstage Application Server 8.0と同様に、非ascii文字をXMLの文字参照形式に変換してascii文字を用いた形式で送信する必要がある場合は、“J2EE ユーザーズガイド”の“Webサービス設定ファイル”を参照してください。

第7章 OLTP機能の移行

ここでは、以下について説明します。

- [ワークユニットの移行](#)
Interstage Application Server V9.1以前から移行する場合に参照してください。
- [CORBAワークユニットのサポートに伴うアプリケーションの移行](#)
INTERSTAGE Application Server V4.x以前から移行する場合に参照してください。
- [CORBAワークユニットの移行](#)
Interstage Application Server V5.x以前から移行する場合に参照してください。
- [CORBAサービスの移行](#)
Interstage Application Server 9.1以前から移行する場合に参照してください。
- [イベントサービスの移行](#)
Interstage Application Server 8.0以前から移行する場合に参照してください。
- [コンポーネントランザクションサービスの移行](#)
Interstage Application Server V5.x以前から移行する場合に参照してください。
- [データベース連携サービスの移行](#)
Interstage Application Server 8.0以前から移行する場合に参照してください。
- [Portable-ORBの移行](#)
Interstage Application Server V9.0以前から移行する場合に参照してください。

7.1 ワークユニットの移行

7.1.1 Interstage Application Server / Interstage Web Server V9.2での変更内容

Interstage Application Server / Interstage Web Server V9.2(Interstage V9.2)での変更内容を説明します。

■ワークユニット定義“プロセス強制停止時間”の機能範囲を拡張

ワークユニット定義の“プロセス強制停止時間”がワークユニット通常停止と同期停止に対しても有効になりました。(Interstage Application Server V9.1以前は、ワークユニット強制停止のみ有効でした。)

定義の詳細は、“OLTPサーバ運用ガイド(「ワークユニット定義」「Control Optionセクション」の「Shutdown Time: プロセス強制停止時間)」、または、“リファレンスマニュアル(コマンド編)(「isj2eeadmin」「IJServer定義ファイル」の「ShutdownMonitorTime)」を参照してください。



注意

旧バージョンから移行の際、注意が必要です。

ワークユニット通常停止と同期停止でもプロセス強制停止時間が有効になったため、旧バージョンから移行する際は、プロセス強制停止時間の値を見直す必要があります。

プロセス強制停止時間の見積もり方法については、“チューニングガイド(「ワークユニットのチューニング」「プロセス強制停止時間のチューニング)」”を参照してください。

互換モード

Interstage Application Server V9.1以前は、ワークユニット強制停止のみ、プロセス強制停止時間は有効でした。ワークユニット通常停止と同期停止では有効ではありません。互換モードは、Interstage Application Server V9.1以前の仕様と同様の運用モードです。

[Interstage Application Server V9.1以前との共通点]

互換モードは、ワークユニット通常停止と同期停止において、プロセス停止処理がプロセス強制停止時間を超過した場合、プロセスは強制停止しません。(ワークユニット強制停止のみプロセスを強制停止します。)

[Interstage Application Server V9.1以前との差異]

互換モードでは、ワークユニット通常停止と同期停止において、プロセス停止処理がプロセス強制停止時間を超過した場合、以下のメッセージを出力します。

【CORBAワークユニットの場合】

- EXTP4381

【IJServerワークユニットの場合】

- EXTP4362

また、スタックトレース、または、スレッドダンプを採取します。

【CORBAワークユニットの場合】 **Solaris**

- ワークユニット定義の“タイムアウト検出時のスタックトレース取得の有無(Output of Stack Trace)”の設定が、“YES、または、省略”の場合、スタックトレースを出力します。

【IJServerワークユニットの場合】

- スレッドダンプを採取します。

ワークユニット同期停止は、アプリケーション処理中に実行すると、処理中の要求が完了するのを待ってからプロセスを停止します。プロセス強制停止時間が設定されていた場合、処理中の要求が遅延すると、処理中の要求が途中で強制終了される場合があります。このような場合、プロセス強制停止時間の見積もりを見直すか、処理中の要求を待ち続けるために、互換モードを設定してください。

注 IJServerワークユニットの通常停止は同期停止と同じ動きとなります。

互換モードは、ワークユニット定義の環境変数で設定します。設定の詳細は、“OLTPサーバ運用ガイド”、または、“リファレンスマニュアル(コマンド編)isj2eeadmin”、または、“Interstage管理コンソールのヘルプ”を参照してください。

【CORBAワークユニットの場合】

- OLTPサーバ運用ガイド
「ワークユニット定義」「Control Optionセクション」の「Shutdown Time: プロセス強制停止時間」
- Interstage管理コンソールのヘルプ
CORBAワークユニットの操作 > 環境設定 > プロセス強制停止時間

【IJServerワークユニットの場合】

- Interstage管理コンソールのヘルプ
IJServerワークユニットの操作 > 環境設定 > プロセス強制停止時間
- リファレンスマニュアル(コマンド編) isj2eeadmin
「IJServer定義ファイル」の「ShutdownMonitorTime」

■ワークユニット定義のデフォルト値変更 **Solaris**

ワークユニット定義の“タイムアウト検出時のスタックトレース取得の有無(Output of Stack Trace)”のデフォルト値を“NO(取得しない)”から“YES(取得する)”に変更しました。定義の詳細は、“OLTPサーバ運用ガイド(「ワークユニット定義」「Control Optionセクション」の「Output of Stack Trace: タイムアウト検出時のスタックトレース取得の有無」)”を参照してください。



注意

旧バージョンから移行の際、注意が必要です。

旧バージョンの設定が移行後も有効ですが、定義が省略されていた場合、移行前後で動作が異なります。
旧バージョンでは、定義を省略した場合、スタックトレースは取得されませんが、移行後は、スタックトレースは取得されます。
移行後もスタックトレースを取得しない場合は、ワークユニット定義の“Output of Stack Trace”を“NO”に設定してください。
定義の変更は“OLTPサーバ運用ガイド”または“Interstage管理コンソール ヘルプ”を参照してください。
定義が省略されているかは、isinfwudefコマンドで確認できます。isinfwudefコマンドについては、“リファレンスマニュアル(コマンド編)”の「isinfwudef」を参照してください。



例

isinfwudefコマンドでワークユニット名を指定して定義内容を表示します。

```
isinfwudef ISSAMPLE1
```

“YES(取得する)”が設定されている場合

```
[Control Option]
Output of Stack Trace:YES
```

“NO(取得しない)”が設定されている場合

```
[Control Option]
Output of Stack Trace:NO
```

省略されている場合

```
[Control Option]
```



ポイント

なお、Interstage管理コンソールは省略できません。Interstage管理コンソールでワークユニット作成、または、更新した場合は、初期値としてデフォルト値が選択されるため、必ず“YES(取得する)”か“NO(取得しない)”のどちらかが設定されています。

旧バージョンのInterstage管理コンソールのデフォルト値は“NO(取得しない)”です。スタックトレースの取得を推奨しているため、移行後は、“YES(取得する)”に変更してください。

7.2 CORBAワークユニットのサポートに伴うアプリケーションの移行

CORBAワークユニットは、以下の製品で利用可能です。

- Interstage Application Server Enterprise Edition

INTERSTAGE Application Server V4.x(INTERSTAGE V4.x)以前の環境で動作していたアプリケーションを、CORBAワークユニットとして運用させるための移行手順について説明します。

■CORBAアプリケーションをワークユニット上で動作させる場合

CORBAアプリケーションをワークユニットで動作させるためには、ワークユニット定義の作成、登録が必要です。
ワークユニット定義/登録方法については、“アプリケーション作成ガイド(CORBAサービス編)”を参照してください。

■トランザクションアプリケーションからCORBAアプリケーションへ移行

INTERSTAGE V4.x以前のワークユニット(トランザクションアプリケーション)をCORBAワークユニットに移行するには、アプリケーションの書換え(使用APIの変更と、main処理の追加)が必要です。

アプリケーションの書換え方法については、“アプリケーション作成ガイド(CORBAサービス編)”を参照してください。

7.3 CORBAワークユニットの移行

Interstage Application Server V6.0(Interstage V6.0)以降では、CORBAワークユニットを、Interstage管理コンソールを使用して運用することで、以下の操作を一元管理することが可能になりました。Interstage管理コンソールを使用した運用についての詳細は、“OLTPサーバ運用ガイド”を参照してください。

- ・ インプリメンテーションリポジトリ定義の登録/削除(OD_impl_instコマンド)
- ・ ネーミングサービスへの登録/削除(OD_or_admコマンド)
- ・ CORBAワークユニットの作成/更新/参照/削除(isaddwudef/isinfwudef/isdelwudefコマンド)
- ・ CORBAワークユニットの起動/停止(isstartwu/isstopwuコマンド)
- ・ CORBAワークユニットの状態参照(islistwu/islistwudefコマンド)
- ・ CORBAアプリケーションの閉塞/閉塞解除(isinhibitque/ispermitqueコマンド)

以下はコマンドラインでの操作とInterstage管理コンソールでの操作の対応表です。

操作内容	コマンドラインでの操作	Interstage管理コンソールを使用した場合の操作画面
CORBAワークユニットの作成	isaddwudefコマンド	ワークユニット:新規作成
インプリメンテーションリポジトリ定義の登録	OD_impl_instコマンド	CORBAアプリケーション:配備
ネーミングサービスへの登録	OD_or_admコマンド	
ワークユニットの起動/停止/状態参照	isstartwuコマンド isstopwuコマンド isinfwudefコマンド	ワークユニット:状態 CORBAワークユニット:操作
CORBAアプリケーションの閉塞/閉塞解除	isinhibitqueコマンド ispermitqueコマンド	CORBAアプリケーション:状態
ワークユニットの削除	isdelwudefコマンド	ワークユニット:状態
インプリメンテーションリポジトリ定義の削除	OD_impl_inst -d コマンド	CORBAアプリケーション:配備解除
ネーミングサービスからの削除	OD_or_adm -d コマンド	
CORBAワークユニットの更新	isaddwudef -o コマンド	CORBAアプリケーション:環境設定 CORBAワークユニット:環境設定

7.4 CORBAサービスの移行

7.4.1 Interstage Application Server V9.2での変更内容

Interstage Application Server V9.2(Interstage V9.2)での変更内容を説明します。

- ・ [CORBAワークユニットの標準出力・標準エラー出力について](#)
- ・ [APIの動作について](#)
- ・ [JDK/JRE 6について](#)
- ・ [インタフェース情報不一致検出機能について](#)

■CORBAワークユニットの標準出力・標準エラー出力について Windows

COBOL言語で作成したCORBAワークユニットアプリケーションについて、翻訳時に翻訳オプション“SSOUT(環境変数情報名)”を指定しない場合に、stdoutファイルおよびstderrファイルにデータが出力されるようになります。

■APIの動作について

以下のAPIについて、接続先URLのリスト(m)に<address>を省略した不正なURLを指定した場合の動作が変更されました。

言語	API名
C言語	CORBA_ORB_resolve_initial_references_remote()
C++言語	CORBA::ORB::resolve_initial_references_remote()
Java	org.omg.CORBA.ORB.resolve_initial_references_remote()
COBOL	CORBA-ORB-RESOLVE-INITIAL-REFERENCES-REMOTE
OOCOBOL	CORBA-ORB-RESOLVE_INITIAL_REFERENCES_REMOTE

・ 変更前

APIは例外を返却しません。

・ 変更後

APIは指定されたオブジェクトが見つからない旨を通知する例外を返却します。

■JDK/JRE 6について

JDK/JRE 6を使用する場合、Interstage Application Serverのインストール時には、JDK/JRE 6.0のインストールディレクトリにorb.propertiesファイルが格納されません。JDK/JRE 6を使用してCORBA-Javaアプリケーションを実行する場合は、必ずORB (Object Request Broker)を指定してください。ORB (Object Request Broker)の指定方法については、“アプリケーション作成ガイド (CORBAサービス編)”の“アプリケーションの開発 (Java言語)” – “CORBAアプリケーションの実行” – “ORB(Object Request Broker)の指定”を参照してください。

■インタフェース情報不一致検出機能について Linux

RHEL-AS4(IPF) または RHEL5(IPF) において、IDLcコマンドに“-f”オプションを指定してインタフェース情報不一致検出機能を有効にしている場合にスタブ・スケルトンファイルに出力されるインタフェース情報一致判定用のダイジェスト値が他プラットフォームと同じ値になるよう修正されました。

そのため、IDLcコマンドに“-f”オプションを指定してV9.1以前で生成したスタブ・スケルトンファイルを使用したCORBAアプリケーションとV9.2以降で生成したスタブ・スケルトンファイルを使用したCORBAアプリケーションの疎通を行った場合に、インタフェース情報が不一致と判定される場合があります。その場合、スタブ・スケルトンファイルを生成するVLをV9.1以前またはV9.2以降のどちらかに統一してください。

7.4.2 Interstage Application Server / Interstage Web Server V9.1での変更内容

Interstage Application Server / Interstage Web Server V9.1(Interstage V9.1)での変更内容を説明します。

- ・ [コード系について](#)
- ・ [C++アプリケーションの動作について](#)

■コード系について

ADJUSTまたはInterstage Charset Managerがインストールされている場合の、CORBAサービスが使用する標準コード変換機能のコード系が変更されました。

そのため、ADJUSTまたはInterstage Charset Managerの標準コード変換機能(外字変換など)においてUCS2/UTF8を使用していた場

合は、UTF16BE/UTF8_4に変更する必要があります。標準コード変換機能の設定方法の詳細については、ADJUSTまたはInterstage Charset Managerのマニュアルを参照してください。

CORBAのコード系	標準コード変換機能のコード系	
	変更前	変更後
UNICODE	UCS2	UTF16BE (注)
UTF8	UTF8	UTF8_4 (注)

注)

ADJUSTまたはInterstage Charset Managerが“UTF16BE”と“UTF8_4”のコード系をサポートしていない場合、CORBAサービスが使用するコード系は変更前から変わらず以下のとおりです。

- UNICODE: UCS2
- UTF8: UTF8

ADJUSTまたはInterstage Charset Managerがサポートしているコード系については、使用しているバージョンのADJUSTまたはInterstage Charset Managerのマニュアルを参照してください。

■C++アプリケーションの動作について

C++言語のCORBAアプリケーション動作時にメモリが不足した場合のアプリケーションの挙動が変更されました。

種別	変更前	変更後
全般	スケルトンの処理内で例外が発生し、かつその例外設定処理内でメモリが不足した際、以下のいずれかの事象が発生する場合があります。 • std::bad_alloc例外が通知されます。 • アプリケーションがダウン、またはハングアップします。	スケルトンの処理内で例外が発生し、かつその例外設定処理内でメモリが不足した場合、例外クラスCORBA::StExcep::NO_MEMORYが通知されます。
	スタブの処理内でメモリが不足した際、以下のいずれかの事象が発生する場合があります。 • std::bad_alloc例外が通知されます。 • アプリケーションがダウン、またはハングアップします。	スタブの処理内でメモリが不足した場合、例外クラスCORBA::StExcep::NO_MEMORYが通知されます。
	_duplicate()または_narrow()でメモリが不足した際、以下のいずれかの事象が発生する場合があります。 • std::bad_alloc例外が通知されます。 • アプリケーションがダウン、またはハングアップします。	_duplicate()または_narrow()でメモリが不足した場合、関数はNULLを返します。
any型	コピーコンストラクタ、代入演算子、左シフト代入演算子でメモリが不足した際、以下のいずれかの事象が発生する場合があります。 • std::bad_alloc例外が通知されます。 • アプリケーションがダウン、またはハングアップします。	コピーコンストラクタ、代入演算子、左シフト代入演算子でメモリが不足した場合、any型のデータ域(_value)がNULLになります。

種別	変更前	変更後
シーケンス型	コンストラクタ、代入演算子でメモリが不足した際、以下のいずれかの事象が発生する場合があります。 • <code>std::bad_alloc</code>例外が通知されます。 • アプリケーションがダウン、またはハングアップします。	コンストラクタ、代入演算子でメモリが不足した場合は、以下の動作を行います。 • サイズ指定ありのシーケンスの場合、データ域(<code>_buffer</code>)にはNULLが設定され、使用個数(<code>_length</code>)には0が設定されます。 • サイズ指定なしのシーケンスの場合、データ域(<code>_buffer</code>)にはNULLが設定され、使用個数(<code>_length</code>)と最大個数(<code>_maximum</code>)には0が設定されます。
	データ域獲得関数(<code>XXX::XXX::allocbuf</code>)でメモリが不足した際、以下のいずれかの事象が発生する場合があります。 • <code>std::bad_alloc</code>例外が通知されます。 • アプリケーションがダウン、またはハングアップします。	データ域獲得関数(<code>XXX::XXX::allocbuf</code>)でメモリが不足した場合、データ域獲得関数はNULLを返します。
	<code>length(ULong)</code>関数でメモリが不足した際、以下のいずれかの事象が発生する場合があります。 • <code>std::bad_alloc</code>例外が通知されます。 • アプリケーションがダウン、またはハングアップします。	<code>length(ULong)</code>関数でメモリが不足した場合、使用個数(<code>_length</code>)およびデータ域(<code>_buffer</code>)は変更されません。
	左シフト代入演算子でメモリが不足した際、以下のいずれかの事象が発生する場合があります。 • <code>std::bad_alloc</code>例外が通知されます。 • アプリケーションがダウン、またはハングアップします。	左シフト代入演算子でメモリが不足した場合、<code>any</code>型のデータ域(<code>_value</code>)がNULLになります。
共用体	コピーコンストラクタ、代入演算子、メンバアクセス関数でメモリが不足した際、以下のいずれかの事象が発生する場合があります。 • <code>std::bad_alloc</code>例外が通知されます。 • アプリケーションがダウン、またはハングアップします。	コピーコンストラクタ、代入演算子、メンバアクセス関数でメモリが不足した場合、共用体データ域(<code>_ptr</code>)がNULLになり、弁別情報(<code>_d</code>)が0に設定されます。
配列	領域獲得関数(<code>XXX::XXX::XXX_alloc</code>)でメモリが不足した際、以下のいずれかの事象が発生する場合があります。 • <code>std::bad_alloc</code>例外が通知されます。 • アプリケーションがダウン、またはハングアップします。	領域獲得関数(<code>XXX::XXX::XXX_alloc</code>)でメモリが不足した場合、領域獲得関数はNULLを返します。
varクラス	コンストラクタ、代入演算子でメモリが不足した場合、以下のいずれかの事象が発生する場合があります。 • <code>std::bad_alloc</code>例外が通知されます。	コンストラクタ、代入演算子でメモリが不足した場合、<code>_ptr</code>がNULLになります。

種別	変更前	変更後
	アプリケーションがダウン、またはハングアップします。	

7.4.3 Interstage Application Server / Interstage Web Server V9.0での変更内容

Interstage Application Server / Interstage Web Server V9.0(Interstage V9.0)での変更内容を説明します。

- [OD_impl_instコマンドについて](#)
- [configファイルの移行](#)
- [クライアントにおける無通信監視タイムアウトの動作について](#)
- [odadmin_exコマンドのデサポートについて](#)
- [インスタンス保持機能使用時の注意事項](#)
- [システム資源について](#)
- [ファイルの権限について](#)
- [C++アプリケーションのライブラリについて](#)

■OD_impl_instコマンドについて

OD_impl_instコマンドにおいて、-tオプション、-uオプション、および-gオプションを指定した際の動作が変更されました。また、-axオプションで指定するCORBAアプリケーション情報定義ファイルについて、設定可能な値が変更されました。

Windows

- 管理者権限を持たないユーザがOD_impl_instコマンドを実行した場合、-tオプションにS、Uを指定できません。同様に、CORBAアプリケーション情報定義ファイルのtypeにshared、unsharedを指定できません。

Solaris

Linux

- 一般ユーザがOD_impl_instコマンドを実行した場合、-uオプション、-gオプションにはそのユーザの実効ユーザID、実効グループIDのみが指定できます。同様に、CORBAアプリケーション情報定義ファイルのuid、gidにはそのユーザの実効ユーザID、実効グループIDのみを指定できます。
- -uオプション、およびCORBAアプリケーション情報定義ファイルのuidの省略値は、OD_impl_instコマンドを実行したユーザの実効ユーザIDとなります。また、-gオプション、およびCORBAアプリケーション情報定義ファイルのgidの省略値は、OD_impl_instコマンドを実行したユーザの実効グループIDとなります。

■configファイルの移行

以下のパラメタの初期値(省略値)が変更されました。

パラメタ名	初期値(省略値)	
	変更前	変更後
period_client_idle_con_timeout	0	96

Solaris

Linux

configパラメタにおいて以下の2つの既存定義が削除されました。

- *iss_uid*
- *iss_gid*

また、以下のパラメタが追加されました。

- *iss_group*

なお、これに伴い以下のパラメタの意味が変更されています。

- *iss_use*

新規パラメタおよび変更されたパラメタの意味については“チューニングガイド”を参照してください。

■クライアントにおける無通信監視タイムアウトの動作について

クライアントにおける無通信監視(configパラメタ“period_client_idle_con_timeout”で設定されるタイムアウト監視)の時間を超過した際の動作が変更されました。

・ 変更前

時間超過のタイミングではサーバとのコネクションの切断は行いません。次回リクエスト送信時にサーバとのコネクションの切断・再接続を行います。

・ 変更後

スレッドモードのクライアントアプリケーションについては、時間超過のタイミングでサーバとのコネクションを切断します。次回リクエスト送信時にサーバとのコネクションの再接続を行います。

プロセスモードのクライアントアプリケーションについては、時間超過のタイミングではコネクション切断は行わず、次回リクエスト送信時にサーバとのコネクションの切断・再接続を行います。

■odadmin_exコマンドのデサポートについて **Windows**

odadmin_exコマンドはInterstage Application Server V9.0以降でデサポートとなりました。CORBAサービスの初期環境設定を行う場合は、Interstage統合コマンドのisinitコマンドを使用するか、Interstage管理コンソールを使用してください。

■インスタンス保持機能使用時の注意事項

CORBAアプリケーション情報定義ファイルに“iswitch=ON”を指定してインスタンス保持機能を使用しているアプリケーションの場合、クライアントにおける無通信監視時間(configパラメタ“period_client_idle_con_timeout”で設定されるタイムアウト監視時間)について注意する必要があります。

Interstage V9.0以降では、“period_client_idle_con_timeout”の初期値が**変更されました**。これにより、クライアントにおける無通信監視がデフォルトで有効になります。

クライアントにおける無通信監視タイムアウトが発生した場合、サーバとのコネクションが切断され、インスタンスが開放されます。このため、インスタンス保持機能がユーザの意図どおりに動作しない可能性があります。インスタンス保持機能を使用する場合は、クライアントにおける無通信監視タイムアウトが発生しないように、サーバへのリクエスト送信間隔と“period_client_idle_con_timeout”の値についての考慮が必要となります。

■システム資源について **Solaris** **Linux**

共有メモリに対するシステムパラメタの計算式が変更されました。詳細は“チューニングガイド”を参照してください。

■ファイルの権限について **Solaris** **Linux**

以下のファイルの権限が変更されました。

ファイル名	変更前の権限	変更後の権限
<CORBAサービスのインストールパス >/var/log	0666	0644
<CORBAサービスのインストールパス >/var/log.old	0666	0644

■C++アプリケーションのライブラリについて **Solaris**

Sun WorkShop 4.2でC++のCORBAアプリケーションを開発する際に使用していたlibOMcCpp.so、libOMcncpp.so、libOMlbocpp.soは使用できません。Forte Developer 6以降でC++のCORBAアプリケーションを開発する際にはlibOMcCpp50.so、libOMcncpp50.so、libOMlbocpp50.soを使用してください。詳細は“アプリケーション作成ガイド (CORBAサービス編)”を参照してください。

7.4.4 Interstage Application Server 8.0での変更内容

Interstage Application Server 8.0(Interstage 8.0)での変更内容を説明します。

■SSLの暗号化方式について

odsetSSLコマンドで暗号化方式の指定を省略した場合(level2オプションおよびlevel3オプションを省略した場合)に設定される暗号化方式が変更されました。詳細については“リファレンスマニュアル(コマンド編)”を参照してください。

なお、INTERSTAGE Application Server V3.x(INTERSTAGE V3.x)以前で暗号化方式の指定省略時に設定される暗号化方式とInterstage 8.0で暗号化方式の指定省略時に設定される暗号化方式とは一致する暗号化方式はありません。そのため、INTERSTAGE V3.x以前の製品とSSL接続を行う場合は、暗号化方式が一致するよう設定を行う必要があります。

7.4.5 Interstage Application Server V7.0での変更内容

Interstage Application Server V7.0(Interstage V7.0)での変更内容を説明します。

- ・ システム資源について
- ・ リクエストの振り分け方式について
- ・ HTTPトンネリングについて
- ・ サンプルプログラムについて
- ・ クライアントタイムアウトの動作について
- ・ CORBA-Javaアプリケーションのメモリ使用改善について
- ・ Windows(R)のアプリケーションのコンパイルオプション変更について
- ・ プロセスモードクライアントの無通信監視切断時の動作について
- ・ CORBAサービス資源ファイルの移入について
- ・ CORBA-Javaアプリケーションのオブジェクトリファレンス生成について
- ・ CORBA-JavaアプリケーションのHelperクラスについて
- ・ Javaインタフェースについて (JDK/JRE1.3以前からJDK/JRE1.4以降への移行)
- ・ コマンドの引数指定方法について

■システム資源について **Solaris** **Linux**

共有メモリに対するシステムパラメタの計算式が変更されました。詳細は“チューニングガイド”を参照してください。

■リクエストの振り分け方式について

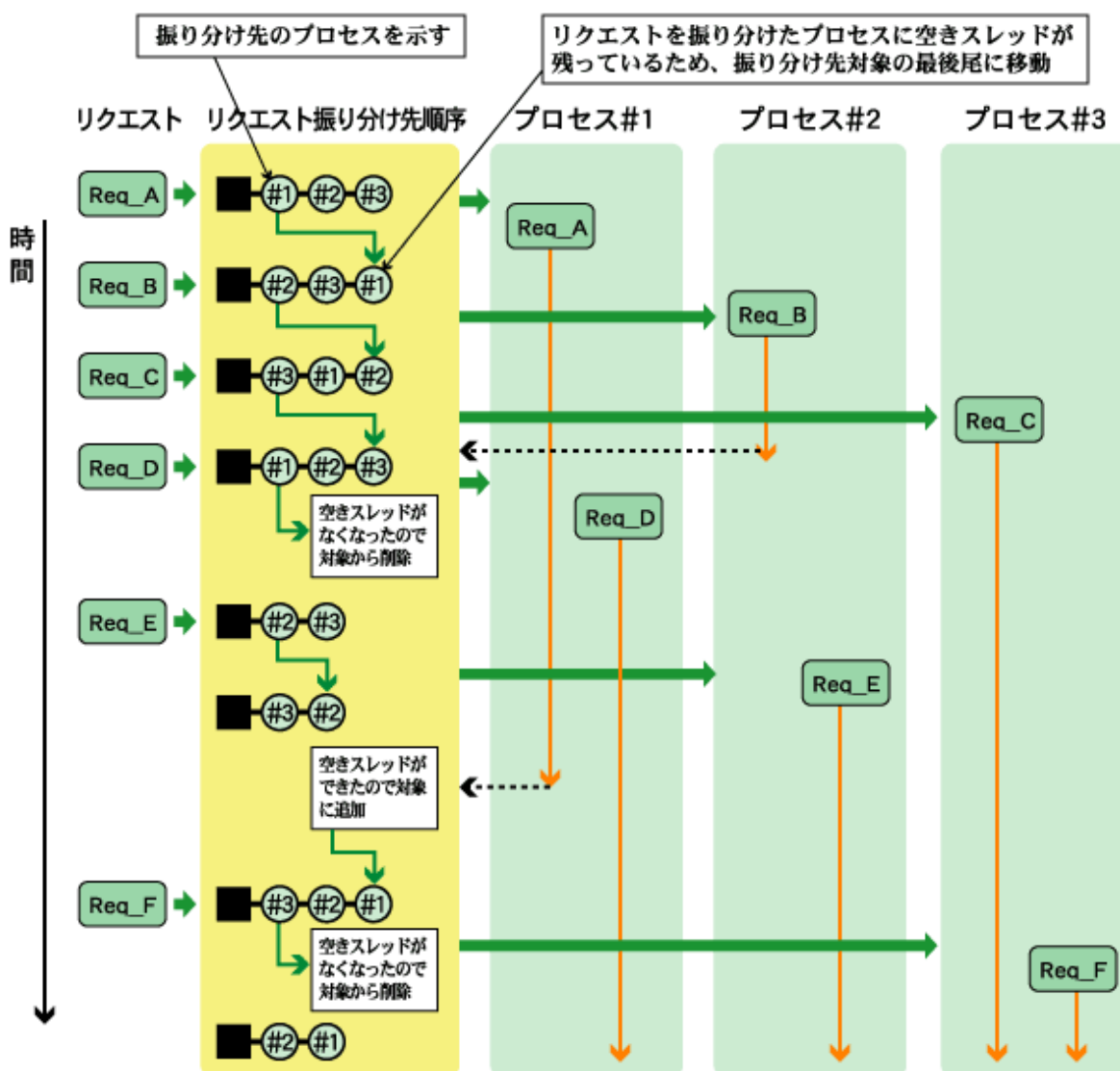
以下に該当するプロセス多重のCORBAサーバに対するリクエストの振り分け方式が変更されました。

- ・ ワークユニット定義の“Request Assignment Mode”ステートメントに“FIFO”を指定したCORBAワークユニット
- ・ ワークユニットでないCORBAサーバアプリケーション

Interstage V7.0以降でのリクエストの振り分け方式では、振り分け先の対象となるサーバプロセスの中で最も長い時間リクエストが振り分けられていないサーバプロセスにリクエストを振り分けます。

リクエストを振り分けたサーバプロセスからリクエスト処理用の空きスレッドがなくなった場合、振り分け先の対象から外れます。

振り分け先の対象から外れていたサーバプロセスに空きスレッドができた場合、振り分け先の対象に含めます。



サーバのプロセス多重度を3、スレッド多重度を2とした場合(多重度はOD_impl_instコマンドで指定します。)

■HTTPトンネリングについて

gwconfigファイルの以下のパラメタについて指定範囲が変更されました。詳細については“チューニングガイド”を参照してください。

パラメタ名	Interstage V6.0以前	Interstage V7.0以降
timeout_response	1～3600	0～100000000
timeout_session	1～900	0～2147483647
timeout_connection	1～600	0～2147483647

■サンプルプログラムについて

Interstage V7.0.1、V7.0L11では、C言語およびC++言語のサンプルプログラムのサーバの動作モードがCOMPATIBLEからSYNC_ENDに変更されました。Interstage管理コンソールを使用してサンプルのサーバアプリケーションをCORBAワークユニットに配備する場合、動作モードにSYNC_ENDを指定する必要があります。

■クライアントタイムアウトの動作について

configパラメタ“period_receive_timeout”に0を設定した際のクライアントの動作が変更されました。

・ Interstage V6.0以前(変更前)

HTTPトンネリングを使用しているクライアントは、リクエスト送信後に0秒で即時タイムアウトとなります。

HTTPトンネリングを使用していないクライアントは、リクエスト送信後にサーバから返信が来るまで無限待ちを行います。

・ Interstage V7.0以降(変更後)

HTTPトンネリングの使用・未使用にかかわらず、クライアントは、リクエスト送信後にサーバから返信が来るまで無限待ちを行います。

■CORBA-Javaアプリケーションのメモリ使用改善について

CORBA-JavaアプリケーションでCORBAサービスが獲得する通信バッファの解放タイミングが、FullGC依存から通信ごとに変更されたため、アプリケーションのメモリ使用量が改善されました。

この改善効果を得るためには、IDLのコンパイルおよびJavaのコンパイルを再度実行してください。

なお、JDK1.3を使用する場合は、ネーミングサービスへのアクセスに使用しているクラスをorg.omg.CosNaming.NamingContextから、org.omg.CosNaming.NamingContextExtに変更して、ネーミングサービスへアクセスする必要があります。

Interstage V6.0でのプログラム記述例

```
-----
org.omg.CORBA.Object tmpObj =
Orb.resolve_initial_referneces("NamingService");
org.omg.CosNaming.NamingContext Cos =
org.omg.CosNaming.NamingContextHelper.narrow(tmpObj);
-----
```

Interstage V7.0でのプログラム記述例

```
-----
org.omg.CORBA.Object tmpObj =
Orb.resolve_initial_referneces("NamingService");
org.omg.CosNaming.NamingContextExt Cos =
org.omg.CosNaming.NamingContextExtHelper.narrow(tmpObj);
-----
```


■Windows(R)のアプリケーションのコンパイルオプション変更について Windows

CORBAサービスが提供しているライブラリの使用するランタイムがスタティックバージョンのライブラリ(Visual C++のコンパイルオプションで“マルチスレッド”を選択)からダイナミックバージョンのライブラリ(Visual C++のコンパイルオプションで“マルチスレッド(DLL)”を選択)に変更されました。

Windows(R)では一つのプロセス内で、スタティックバージョンとダイナミックバージョンのランタイムライブラリの混在が許されていないため、Visual C++のコンパイルオプションで“使用するランタイム”を“マルチスレッド”に設定して作成したInterstage V6.0以前のC/C++ユーザアプリケーションは、Interstage V7.0以降で使用することができません。そのため、ユーザアプリケーションを移行する場合には、ユーザアプリケーションのコンパイルオプションも変更してアプリケーションを再ビルドする必要があります。再ビルドを行わないとCORBAワークユニットアプリケーションの標準出力/標準エラー出力がstdoutファイル/stderrファイルに出力されません。また、アプリケーションの動作に異常が発生する可能性があります。

以下のような手順で表に示すVisual C++のオプションを設定してください。

[プロジェクト]-[プロパティ]-[構成プロパティ]-[C/C++]

カテゴリ	項目	設定値
コード生成	使用するランタイム	マルチスレッド(DLL) (注)

注) Interstage V6.0以前で“マルチスレッド”と設定されていた値を変更します。

■プロセスモードクライアントの無通信監視切断時の動作について Solaris Linux

無通信監視(configパラメタ“period_idle_con_timeout”で設定されるタイムアウト監視)によるコネクション切断が発生した際の、プロセスモードクライアントにおける動作が変更されました。

・ Interstage V6.0以前(変更前)

無通信監視切断発生後に再接続を行おうとすると、COMM_FAILURE例外が発生して接続に失敗します。

・ Interstage V7.0以降(変更後)

無通信監視切断発生後に再接続を行う場合、同期通信のときは接続に成功します。非同期通信のときは、Interstage V6.0以前と同様にCOMM_FAILURE例外が発生して接続に失敗します。

■CORBAサービス資源ファイルの移入について

CORBAサービス資源ファイルの移入において、既存環境のインタフェースリポジトリサービス資源のデータベースの移入処理が変更されました。

・ Interstage V6.0以前(変更前)

CORBAサービス資源ファイルを移入する際、インタフェースリポジトリサービス資源のデータベースの格納先に、すでにインタフェースリポジトリサービス資源のデータベースファイルが存在している場合は、エラーメッセージod16271を出力し、移入処理を中止します。

そのため、移入する前に、必ずインタフェースリポジトリサービス資源のデータベースの格納先に存在するインタフェースリポジトリサービス資源のデータベースを削除しておく必要があります。

・ Interstage V7.0以降(変更後)

CORBAサービス資源ファイルを移入する際、インタフェースリポジトリサービス資源のデータベースの格納先に、すでにインタフェースリポジトリサービス資源のデータベースファイルが存在し、かつデータベースファイルが移入前のローカルホストのインタフェースリポジトリが使用しているファイルであった場合は、データベースファイルを削除して移入処理を続行します。データベースファイルが移入前のローカルホストのインタフェースリポジトリが使用しているファイルではなかった場合は、Interstage V6.0以前と同様にエラーメッセージod16271を出力し、移入処理を中止します。

したがって、移入前のローカルホストのインタフェースリポジトリを使用する環境に対して移入する場合、事前にインタフェースリポ

ジトリサービス資源のデータベースを削除しておく必要はありません。

■CORBA-Javaアプリケーションのオブジェクトリファレンス生成について

CORBA-Javaサーバアプリケーションでオブジェクトリファレンスを動的生成した場合、使用可能なオブジェクトID(oid)はシステム内で一意となります。

このため、デフォルトインスタンス方式のCORBA-Javaサーバアプリケーションをプロセス多重で動作させる場合、RequestProcessingポリシーに“USE_DEFAULT_SERVANT”を指定していないPOA (rootPOAなど)を使用して動的に作成したオブジェクトリファレンスで、サーバアプリケーションを呼び出すと、システム例外OBJECT_NOT_EXISTが発生することがあります。

デフォルトインスタンス方式のCORBA-Javaサーバアプリケーションをプロセス多重で動作させる場合、アプリケーションプログラムにおいて以下のオブジェクトリファレンスを使用してください。

- RequestProcessingポリシーに“USE_DEFAULT_SERVANT”を指定したPOAを使用して動的生成したオブジェクトリファレンス
- 事前生成方式で作成したオブジェクトリファレンス

デフォルトインスタンス方式の詳細については、“アプリケーション作成ガイド(CORBAサービス編)”の“インスタンス管理とアプリケーション形態”-“アプリケーション形態の種別”を参照してください。

■CORBA-JavaアプリケーションのHelperクラスについて

Interstage V7.0以降では、Helperクラスの修飾子にabstractが追加されました。

CORBA-Javaアプリケーションにおいて、Helperクラスをインスタンス化している場合は、アプリケーションのコンパイル時・実行時にエラーが発生します。アプリケーションにおいて、インスタンス化しないでHelperクラスのメソッドを使用するように修正してください。

■Javaインタフェースについて(JDK/JRE1.3以前からJDK/JRE1.4以降への移行)

Interstage V7.0以降では、標準インストールでインストールされるJDK/JREのバージョンがJDK/JRE1.4以降に変更されました。

JDK/JRE1.3以前で動作していたCORBA-JavaアプリケーションをJDK/JRE1.4以降へ移行する場合は、JDK1.4以降のjavaコンパイラで再度アプリケーションをコンパイルする必要があります。JavaインタフェースのJDK/JRE1.3以前からJDK/JRE1.4以降への移行については、“アプリケーション作成ガイド(CORBAサービス編)”の“Javaインタフェース(JDK/JRE1.3以前からJDK/JRE1.4以降への移行)”を参照してください。

■コマンドの引数指定方法について

Solaris

Linux

odlistnsコマンドおよびodimportnsコマンドのオプションの指定方法が変更されました。

• Interstage V6.0以前(変更前)

odlistnsコマンドおよびodimportnsコマンドの実行時に、1つのオプションの要素に複数のオプションを指定した場合は、複数のオプションとして動作します。たとえば、“odlistns -lR”と実行すると、“odlistns -l -R”として動作します。ただし、推奨ではありません。

• Interstage V7.0以降(変更後)

プラットフォーム間でコマンドの指定方法を統一したことにより、“リファレンスマニュアル(コマンド編)”の“形式”で説明している指定方法以外では動作しません。したがって、odlistnsコマンドおよびodimportnsコマンドの実行時に、1つのオプション要素に複数のオプションを指定した場合は、最初のオプションだけが有効となり、2つ目以降のオプションは無効となります。複数のオプションを指定する場合は、“リファレンスマニュアル(コマンド編)”の“形式”のとおり、オプションごとに指定してください。

7.4.6 Interstage Application Server V6.0での変更内容

Interstage Application Server V6.0(Interstage V6.0)での変更内容を説明します。

- [configファイルの移行](#)
- [SSL環境の構築](#)

- [OD_impl_instの定義ファイル](#)
- [マニュアルから削除された定義](#)
- [サンプルプログラム](#)

■configファイルの移行

configパラメタにおいて以下の2つの既存定義が削除されました。

- `max_IIOp_init_con`
- `max_IIOp_req_per_con`

なお、以下の3つのパラメタが追加されました。新規パラメタの意味については“チューニングガイド”を参照してください。

新規パラメタ	旧定義から新定義への移行方法
<code>max_IIOp_resp_requests</code>	$max_IIOp_resp_con \times (max_IIOp_req_per_con + 1)$
<code>max_IIOp_local_init_con</code> (省略値:256 右記が256の場合、記述不要)	以下の内の最大値 • <code>max_IIOp_init_con</code> • 256
<code>max_IIOp_local_init_requests</code> (省略値:4096 右記が4096以下の場合、記述不要)	以下の内の最大値 • $max_IIOp_init_con \times max_IIOp_req_per_con$ • 4096

config移行後、`max_IIOp_init_con`と`max_IIOp_req_per_con`の旧定義は削除してください。

旧定義のconfigを移行せずに使用した場合、`max_IIOp_local_init_con`と`max_IIOp_local_init_requests`は、上記の計算式に従って計算されます。また、`max_IIOp_resp_requests`は、以下の計算式に従って計算されます。

max_IIOp_resp_requests

条件		設定値
number_of_common_bufferが0以外		$max_IIOp_resp_con \times (max_IIOp_req_per_con + 1)$
number_of_common_bufferが未定義、または0	$max_IIOp_resp_con < (max_IIOp_req_per_con + 1)$	$[max_IIOp_resp_con \text{ と } 10 \text{ の内の最小値}] \times (max_IIOp_req_per_con + 1)$
	$max_IIOp_resp_con \geq (max_IIOp_req_per_con + 1)$	$max_IIOp_resp_con \times [(max_IIOp_req_per_con + 1) \text{ と } 10 \text{ の内の最小値}]$

number_of_common_bufferの計算式については“チューニングガイド”の“CORBAサービスの動作環境ファイル”にあるnumber_of_common_bufferの説明箇所を参照してください。

■SSL環境の構築

Interstage V5.x以前において使用していたodsetSSLコマンドはInterstage V6.0以降においても使用可能ですが、odsetSSLコマンドを使用して構築したSSL環境はInterstage管理コンソールを使用して操作することはできません。

Interstage管理コンソールによって構築したSSL環境を使用する設定になっている場合にodsetSSLコマンドを使用すると、odsetSSLコマンドによって構築したSSL環境が有効になります。逆に、odsetSSLコマンドによって構築したSSL環境を使用する設定になっている場合にInterstage管理コンソールによって構築したSSL環境を使用する設定に変更するとInterstage管理コンソールによって構築したSSL環境が有効となります。

■OD_impl_instの定義ファイル

Interstage V6.0以降では、サーバアプリケーション起動時に、サーバアプリケーション開発言語の自動判定を行います。それに伴い、OD_impl_instコマンドの-axオプションで指定する定義ファイルについて、以下の定義が廃止となりました。

- lang

Interstage V5.x以前と同様に、lang定義を指定して登録することも可能です。

■マニュアルから削除された定義

以下の定義についてのマニュアル記載が削除されました。マニュアルから削除された定義については、定義された場合の動作は旧バージョンと変更はありませんが、説明についてはInterstage V5.x以前の旧バージョンマニュアルを参照してください。

マニュアル削除定義	旧マニュアル参照箇所
server per method	“リファレンスマニュアル(コマンド編)” > “OD_impl_inst” Interstage Application Server Standard/Enterprise Editionのマニュアルの場合は以下の箇所も参照できます。 “アプリケーション作成ガイド(CORBAサービス編)” > “CORBAアプリケーションの設計” > “サーバアプリケーションの起動タイプ”
max_file_descriptors	本定義はWindows(R)版には存在しませんので、Windows(R)版のマニュアル記事は旧バージョンにもありません。SolarisとLinuxについては以下の場所です。 “チューニングガイド” > “CORBAサービスの動作環境ファイル” > “config”
client_bind ssllib msg_compatible	“チューニングガイド” > “CORBAサービスの動作環境ファイル” > “config”
max_session max_connectionmax_request	“チューニングガイド” > “CORBAサービスの動作環境ファイル” > “gwconfig”

■サンプルプログラム Solaris

C++サンプルプログラムのMakefileがWorkShop 5.0以降のコンパイラ用に変更されました。WorkShop 4.2シリーズのC++コンパイラを使用する場合はMakefileを修正する必要があります。

7.4.7 Interstage Application Server V5.1での変更内容

Interstage Application Server V5.1(Interstage V5.1)での変更内容を説明します。

■クライアントタイムアウトの監視方法について

period_receive_timeout/パラメタで設定するサーバメソッド待機時間の監視方法がコネクション単位から、リクエスト単位に変更されました。

7.4.8 INTERSTAGE Application Server V3.xからの移行

INTERSTAGE Application Server V3.x(INTERSTAGE V3.x)のCORBAサービス環境を移行するための手順、および注意事項について説明します。

なお、Interstage Application Server V5.1以降の変更内容については各VLの記事を参照してください。

■バックアップコマンドの移行

INTERSTAGE V3.x Enterprise Edition 又はINTERSTAGE V3.x Standard Editionのバックアップコマンドを使用してCORBAサービス資源を移行するための手順について説明します。

使用するコマンド

Windows

機能	コマンド
CORBAサービス資源(インタフェースリポジトリ以外)のバックアップ	C:\¥INTERSTAGE¥ODWIN¥bin¥odbkup.exe
インタフェースリポジトリ環境情報ファイルのバックアップ	C:\¥INTERSTAGE¥ODWIN¥bin¥irbkup.exe
インタフェースリポジトリデータファイルのバックアップ	C:\¥INTERSTAGE¥ODWIN¥bin¥obfbkup.exe
CORBAサービス資源のリストア	C:\¥Interstage¥ODWIN¥bin¥odrestoresys.exe

上記のコマンドは、CORBAサービス(“OD_start”)が停止している状態で実行する必要があります。コントロールパネルの「サービス」でサービスの状態を確認してください。

Solaris

機能	コマンド
CORBAサービス資源(インタフェースリポジトリ以外)のバックアップ	/opt/FSUNod/bin/odbkup
インタフェースリポジトリ環境情報ファイルのバックアップ	/opt/FSUNod/bin/irbkup
インタフェースリポジトリデータファイルのバックアップ	/opt/FSUNod/bin/obfbkup
CORBAサービス資源のリストア	/opt/FSUNod/bin/odrestoresys

上記のコマンドは、CORBAサービスが停止している状態で実行する必要があります。

INTERSTAGE V3.x環境のバックアップ方法 Windows

バックアップ先パスがX:\¥V3Backupの場合の操作例を以下に示します。

1. バックアップ用フォルダを作成します。
mkdir X:\¥V3Backup
2. odbkupコマンドを実行して、インタフェースリポジトリ以外の資源ファイルをバックアップします。
odbkup X:\¥V3Backup
3. irbkupコマンドを実行して、インタフェースリポジトリ環境情報ファイルをバックアップします。
irbkup X:\¥V3Backup
4. obfbkupコマンドを実行して、インタフェースリポジトリデータファイルをバックアップします。
obfbkup X:\¥V3Backup
5. CORBAサービスの環境情報が格納されたファイルをバックアップします(Interstage V3.1のみ)。
 - isinitコマンドで初期設定を行った場合:
copy C:\¥INTERSTAGE¥TD¥var¥iscom¥isei_odenvfile X:\¥V3Backup
rename X:\¥V3Backup¥isei_odenvfile X:\¥V3Backup¥odenvfile

- CORBAサービス初期設定コマンド(odadmin_ex)で初期設定を行った場合:
copy C:¥INTERSTAGE¥ODWIN¥var¥odenvfile X:¥V3Backup

INTERSTAGE V3.x環境のバックアップ方法 **Solaris**

バックアップ先パスが/V3Backupの場合の操作例を以下に示します。

1. バックアップ用ディレクトリを作成します。
mkdir /V3Backup
2. odbkupコマンドを実行して、インタフェースリポジトリ以外の資源ファイルをバックアップします。
odbkup /V3Backup
3. irbkupコマンドを実行して、インタフェースリポジトリ環境情報ファイルをバックアップします。
irbkup /V3Backup
4. obfbkupコマンドを実行して、インタフェースリポジトリデータファイルをバックアップします。
obfbkup /V3Backup
5. CORBAサービスの環境情報が格納されたファイルをバックアップします(Interstage V3.1のみ)。
 - isinitコマンドで初期設定を行った場合:
cp /opt/FSUNtd/var/iscom/isei_odenvfile /V3backup
mv /V3Backup/isei_odenvfile /V3Backup/odenvfile
 - CORBAサービス初期設定コマンド(odadmin_ex)で初期設定を行った場合:
cp /opt/FSUNod/var/odenvfile /V3Backup

本バージョン・レベルへのリストア方法 **Windows**

バックアップ先パスがX:¥V3Backupの場合の操作例を以下に示します。

odrestoresysコマンドを実行して、CORBAサービス資源ファイルをリストアします。
odrestoresys -r -v3 X:¥V3Backup

本バージョン・レベルへのリストア方法 **Solaris**

バックアップ先パスが/V3Backupの場合の操作例を以下に示します。

odrestoresysコマンドを実行して、CORBAサービス資源ファイルをリストアします。
odrestoresys -r -v3 /V3Backup

■移行時の注意事項

◆IDLコンパイラのJava言語生成物

Interstage V4.x以降のIDLコンパイラが生成するJava言語マッピングのスタブ・スケルトンでは、クライアントアプリケーションのin、inoutパラメタ、およびサーバアプリケーションのout、inoutパラメタ、復帰値において、以下のデータ型に“nullオブジェクト”が設定された場合、java.lang.NullPointerExceptionが通知されるようになりました。

- 文字列型
- any型
- シーケンス型

- 構造体
- 共用体
- 配列

INTERSTAGE V3.x以前のアプリケーションにおいて、転送対象のオブジェクトに“nullオブジェクト”を指定している場合は、“nullオブジェクト”を転送対象としないように修正してください。

7.5 イベントサービスの移行

イベントサービスは、以下の製品で利用可能です。

- Interstage Application Server Enterprise Edition
- Interstage Application Server Standard-J Edition

7.5.1 Interstage Application Server V9.0での変更内容

Interstage Application Server V9.0(Interstage V9.0)での変更内容を説明します。

■esmkchnlコマンドに関する注意事項

esmkchnlコマンドにおいて-autodisconオプションを指定してイベントチャネルを作成する場合、CORBAサービスのクライアントにおける無通信監視時間(CORBAサービスの動作環境ファイルのパラメタ“period_client_idle_con_timeout”で設定されるタイムアウト監視時間)について注意する必要があります。

Interstage V9.0以降では、“period_client_idle_con_timeout”の初期値が**変更されました**。これにより、クライアントにおける無通信監視がデフォルトで有効になります。クライアントにおける無通信監視時間超過によりコネクションが切断された場合、継続して通信するためには再度イベントチャネルへの接続を行う必要があります。運用上問題がある場合、“period_client_idle_con_timeout”の値を大きくする、または無効にするといった対処が必要となります。

7.5.2 Interstage Application Server 8.0での変更内容

Interstage Application Server 8.0(Interstage 8.0)での変更内容を説明します。

- [イベントチャネルを閉塞モードで停止する場合の停止処理](#)
- [esmonitorchnlコマンドの表示形式](#)
- [essetcnfコマンドおよびessetcnfchnlコマンドのオプション](#)
- [トレース情報](#)
- [サンプルプログラム](#)
- [イベントサービスの自動セットアップ](#)

■イベントチャネルを閉塞モードで停止する場合の停止処理

Point-To-Pointモデルのメッセージングモデルのイベントチャネルを閉塞モードで停止した場合の停止処理が以下のように変更されました。

- **Interstage V7.0以前(変更前)**
閉塞終了モードを指定した場合、イベントチャネルにイベントデータが蓄積されている間は、イベントチャネルは停止しません。イベントチャネルに蓄積されているイベントデータがコンシューマに配信されて削除されるか、生存時間に達して削除された段階で、イベントチャネルが停止しました。
- **Interstage 8.0以降(変更後)**
閉塞終了モードを指定した場合、コンシューマが接続された状態でイベントチャネルにイベントデータが蓄積されている間は、イ

イベントサービスは停止しません。イベントチャンネルに蓄積されているイベントデータがすでに接続されているコンシューマに配信されて削除されるか、生存時間に達して削除された段階で、イベントサービスが停止します。

Interstage V7.0(変更前)以前の停止処理を行いたい場合は、esetcnfコマンドおよびesetcnfchnlコマンドで設定できます。esetcnfコマンドおよびesetcnfchnlコマンドの詳細については、“リファレンスマニュアル(コマンド編)”の“esetcnf”および“esetcnfchnl”を参照してください。

■esmonitorchnlコマンドの表示形式

esmonitorchnlコマンドの表示形式が以下のように変更されました。

• Interstage V7.0以前(変更前)

- “接続時刻”は、“Date”と表示されました。
- “プロキシ種別(Kind)”が表示されました。

• Interstage 8.0以降(変更後)

- “接続時刻”は、“Date”から“Connected Time”に変更して表示されます。
- “プロキシ種別(Kind)”が削除され、代わりに“最終アクセス時間(Final Access Time)”が表示されます。

esmonitorchnlコマンドの表示形式の詳細については、“リファレンスマニュアル(コマンド編)”の“esmonitorchnl”を参照してください。

■esetcnfコマンドおよびesetcnfchnlコマンドのオプション

esetcnfコマンドおよびesetcnfchnlコマンドのオプションが以下のように変更されました。

• Interstage V7.0以前(変更前)

- dcacheオプションにより不揮発チャンネル運用時のイベントデータのメモリキャッシュ数の設定が可能でした。

• Interstage 8.0以降(変更後)

- dcacheオプションが削除となり、不揮発チャンネル運用時のイベントデータのメモリキャッシュ数は10000000となりました。

■トレース情報 Linux

トレース情報がプロセス単位でも採取可能となったことから、イベントサービス運用時に必要なディスク容量、および共用メモリサイズが変更されました。

ディスク容量については“チューニングガイド”の“運用時に必要なディスク容量”―“サーバ機能を使用する場合”を、共用メモリサイズについては“チューニングガイド”の“サーバ機能運用時に必要なシステム資源”―“イベントサービスのシステム環境の設定”を参照してください。

■サンプルプログラム Windows

C言語およびC++言語のサンプルプログラムは、Microsoft(R) Visual C++(R) .NET Standardのコンパイラ用に変更されています。

■イベントサービスの自動セットアップ

Interstage 8.0以降では、Interstage V6.0で対応していたインストール後、自動的にイベントサービスのセットアップを行っていた処理が対応されていません。

そのため、`essetup`コマンド、Interstage統合コマンド、またはInterstage管理コンソールを使用してイベントサービスのセットアップを行ってください。

7.5.3 Interstage Application Server V6.0での変更内容

Interstage Application Server 6.0(Interstage V6.0)での変更内容を説明します。

■イベントサービスのセットアップ、アンセットアップ **Windows**

イベントサービスおよびイベントファクトリのサービスのスタートアップの種類が“手動”に変更されました。イベントサービスおよびイベントファクトリの起動は、Interstage管理コンソールを使用して操作してください。`essetup`コマンドを使用してイベントサービスのセットアップを行った場合、システムの起動時にイベントサービスおよびイベントファクトリを自動的に起動させるには、サービスのスタートアップの種類を“自動”に変更してください。

サービスのスタートアップの種類を確認するには、Administrator権限でログインし、[コントロールパネル]の[サービス]または[コントロールパネル]—[管理ツール]—[サービス]を起動後、“EventService”および“EventFactory”のスタートアップの種類を参照します。サービスのスタートアップの種類を変更する場合には、“EventService”または“EventFactory”を選択し、[スタートアップ]または[プロパティ]により行います。

■イベントチャネルの起動

イベントサービスの起動時にイベントチャネルを自動的に起動するモードがサポートされました。これにより、Interstage V5.x以前までInterstageの起動(イベントサービスの起動)後に行っていたイベントチャネルの起動処理が不要となりました。

イベントチャネル自動起動の設定は、Interstage管理コンソールまたはイベントサービス運用コマンド(`esetconf`、`esetconfchnl`)を使用して変更することができます。初期値は“自動起動する”です。

Windows

イベントチャネルの自動起動を行う場合は、イベントサービスおよびイベントファクトリのサービスのスタートアップの種類を“手動”にしてください。`esstart`コマンドを使用してイベントサービスを起動する場合は、CORBAサービス(ネーミングサービスを含む)の起動が完了している必要があります。

7.6 コンポーネントランザクションサービスの移行

■ワークユニット自動起動設定ファイルについて

Interstage Application Server V6.0以降では、ワークユニット自動起動設定ファイルの構文が変更されています。ワークユニット名に続いて、@ (アットマーク)の直後にユーザ名を指定します。ファイルの構文については、OLTPサーバ運用ガイドを参照してください。

7.7 データベース連携サービスの移行

データベース連携サービスは以下の製品で利用可能です。

- Interstage Application Server Enterprise Edition
- Interstage Application Server Standard-J Edition

7.7.1 Interstage Application Server V9.0での変更内容

Interstage Application Server V9.0(Interstage V9.0)での変更内容を説明します。

■フェーズ間タイムアウト検出時間(2PC_TIME_OUT)を指定する際の注意事項

`config`ファイルの“2PC_TIME_OUT”を指定してフェーズ間タイムアウト検出時間を設定する場合、CORBAサービスのクライアントにおける無通信監視時間(CORBAサービスの動作環境ファイルのパラメタ“`period_client_idle_con_timeout`”で設定されるタイムアウト監視時間)について注意する必要があります。

Interstage V9.0では、“period_client_idle_con_timeout”の初期値が**変更されました**。これにより、クライアントにおける無通信監視時間はデフォルトで480秒となります。フェーズ間タイムアウト検出時間は、この時間よりも短く設定されている必要があります。

フェーズ間タイムアウト検出時間を設定する場合は、クライアントにおける無通信監視時間の方が長くなるように“period_client_idle_con_timeout”の値を設定してください。

■JTS用リソース管理プログラムが利用するJavaの移行

JDK/JREのサポートバージョンの変更によりJDK/JRE1.3は使用できません。

Configファイルの“JAVA_VERSION”に14を指定してください。Configファイルの“PATH”については、使用するJDK/JREのバージョンにあわせて指定してください。

7.7.2 Interstage Application Server V7.0での変更内容

Interstage Application Server V7.0(Interstage V7.0)での変更内容を説明します。

■ots.iniファイルの改善について **Windows**

ots.iniファイルに設定する“shmaddr”が不要になりました。本バージョン・レベルからは、データベース連携サービスが獲得した共用メモリを固定アドレスにマッピングせずに、オペレーティングシステムが自動的に割り当てたアドレスを使用します。これにより、ユーザはデータベース連携サービスの共用メモリアドレスを意識する必要がなくなりました。

ots.iniファイルの詳細については、“チューニングガイド”の“データベース連携サービスのチューニング”を参照してください。



注意

ots.iniファイルに“shmaddr”を設定した場合でも、設定した値は有効になりません。

7.7.3 Interstage Application Server V5.xからの移行方法

Interstage Application Server V5.x(Interstage V5.x)からの移行方法を説明します。

以前のバージョン・レベルで使用していたデータベース連携サービスを動作させるためには以下の移行手順が必要です。ここではその移行方法について説明します。

- [configファイルの移行](#)
- [JavaVersionの移行](#)

■configファイルの移行

configファイルの移行は、以前のバージョン・レベルのJava実行環境と異なるJava環境においてJTS用リソース管理プログラムを使用する場合に必要となります。

configファイルの“PATH=”行にJTSリソース管理プログラムで利用するjavaコマンドへのフルパスを定義します。

以下に移行手順を示します。

1. Interstageを停止します。
2. configファイルの“PATH”行を参照し、現在のJava環境に修正します。

Windows

- JDK/JRE1.4を利用している場合
PATH=C:¥Interstage¥JDK14¥bin¥java.exe
- JDK/JRE5を利用している場合
PATH=C:¥Interstage¥JDK5¥bin¥java.exe

■JavaVersionの移行

JavaVersionの移行は、以前のバージョン・レベルにおいて、JTS用リソース管理プログラムが利用するJDK/JREを設定している場合のみ必要となります。



ポイント

JDK/JREが設定されているかどうかは、RMP.propertiesに“JavaVersion=”行が存在するかどうかで判断できます。

以下に移行手順を示します。

1. JTS用リソース管理プログラムが停止していることを確認してください。
otsalive

2. RMP.propertiesファイルから“JavaVersion=”の行を削除してください。

3. configファイルの“JAVA_VERSION=”行を以下に修正してください。

— JAVA_VERSION=14

7.7.4 INTERSTAGE Application Server V3.xからの移行

ここでは、INTERSTAGE Application Server V3.x(INTERSTAGE V3.x)のデータベース連携サービスから改善された機能について説明し、移行方法を説明します。ただし、以下の製品に関しては、J2EE機能を持たないINTERSTAGE V3.xからの移行作業は不要です。

- Interstage Application Server Standard-J Edition

■改善された機能

INTERSTAGE V3.x以降、いくつかの機能改善が行われました。それにより、以下に示す手順に沿って移行を行う必要があります。移行を行わなかった場合には、データベース連携サービスを利用するアプリケーションが動作しなくなります。以下は、改善された機能です。

- 登録されたリソース定義ファイルを名前で扱うようになりました。
- 上記に伴いリソース定義ファイルを扱うコマンドのオプションが変更されました。
- 以下に示すコマンドの権限が管理者権限のみ対象になりました。

— otsalive

— otssetrsc

- クラスタ用のデータベース連携サービスシステムおよびリソース管理プログラムの監視コマンド(otsmonitor)が追加されました。
- セットアップ情報ファイルが追加され、セットアップ情報をすべて保存可能になりました。
- リソース定義ファイルが拡張され、リソース定義情報をすべて保存可能になりました。

以降で、これらの移行方法を順に説明します。

■移行方法

以下の移行が必要です。

- [リソース定義ファイルの移行](#)
- [ワークユニット定義の移行](#)
- [状態遷移プロシジャの移行](#)

- ・ 自動化プログラムの移行

以下に示す順序で、該当するものだけ対処するようにしてください。ここで示す移行方法は、データベース連携サービスを使用する場合にだけ必要です。データベース連携サービスを利用しない場合、または新規に利用する場合には対処する必要はありません。ここで説明するすべてのコマンドの詳細については、“リファレンスマニュアル(コマンド編)”で説明されています。

◆リソース定義ファイルの移行

一度登録されたリソース定義ファイルは、リソース定義名で扱うようになりました。これにより、旧環境で使用していたリソース定義ファイルを修正し、otssetrscコマンドで再登録が必要になります。

[移行箇所]

otssetrscコマンドで登録済のすべてのリソース定義ファイル。

[移行手順]

1. リソース定義ファイルの修正
リソース定義ファイルに、NAME句を追加し名前をつけてください。名前は、マシン内でユニークな名前をつけるようにしてください。
2. リソース管理プログラムの登録
otssetrscコマンドで登録してください。

[注意事項]

- ー リソース定義ファイルの詳細については、“アプリケーション作成ガイド(データベース連携サービス編)”の“リソース定義ファイルの作成”を参照してください。
- ー otssetrscコマンドで新たに登録する予定のリソース定義ファイルについても同様に移行を行うようにしてください。

◆ワークユニット定義の移行

データベース連携サービスを使用するワークユニットのワークユニット定義を一部修正する必要があります。

[移行箇所]

ワークユニット定義の[Resource Manager]セクションに書かれているFile:部分。

[移行手順]

File:リソース定義ファイルパスの部分にName:リソース定義名に変更してください。リソース名は、該当リソース定義ファイル移行時のNAMEに指定した名前を指定してください。リソース定義ファイルの詳細は、“アプリケーション作成ガイド(データベース連携サービス編)”の“リソース定義ファイルの作成”を参照してください。

◆状態遷移プロシジャの移行

リソース定義ファイルの移行により、リソース定義ファイルがリソース定義名で扱えるようになりました。そのため、クラスタ環境を利用している場合には状態遷移プロシジャで使用されているコマンドを正しい名前に変更する必要があります。

[移行箇所]

- ・ リソース定義ファイル格納場所の移動
- ・ 状態遷移プロシジャの修正

[移行手順-リソース定義ファイル格納場所の移動]

リソース定義ファイルの格納場所の移動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。

1. 以下のフォルダを丸ごと共有ディスクにコピーまたは移動してください。

Windows

C:\Interstage\ots\etc\repository

Solaris

/opt/FSUNots/etc/repository

2. 以下のファイルのREPOSITORYに1.でコピーした場所をフルパスで記述してください。

Windows

C:\%Interstage%\ots\systeminfo\system

Solaris

/opt/FSUNots/systeminfo/system

3. これらの作業を、運用系ノード、待機系ノードの両方で実施してください。

[移行手順-状態遷移プロシジャの修正:SafeCLUSTERの場合] Windows

以下のように修正してください。

OTS_LISTに記述しているリソース定義ファイルへのフルパスをリソース定義名に変更してください。以下のように修正してください(太字の部分を修正してください)。

```
RESOURCE_PARAM01=c:\%rmp01.exe+リソース定義名1 (注)
RESOURCE_PARAM02=c:\%rmp02.exe+リソース定義名2 (注)
RESOURCE_PARAM03=c:\%rmp03.exe+リソース定義名3 (注)
.....
```

(注) リソース定義ファイルの移行時に指定したリソース定義名(NAMEに指定した名前)に修正してください。

OTS_INTERSTAGE.batを以下のように修正してください。

```
169行目付近:
【修正前】
:START_REC
%START_COMMAND% %RMP% %RSCDEF%
【修正後】
:START_REC
%START_COMMAND% -pg %RMP% -n %RSCDEF%
```

すべてのプロシジャを修正後、SafeCLUSTERに再登録してください。

[移行手順-状態遷移プロシジャの修正:MSCSまたはMicrosoft(R) Windows Server(R) 2008のフェールオーバー クラスタリング機能の場合] Windows

データベース連携サービスおよびワークユニット起動バッチファイルを以下のように修正してください。

```
【変更前】
echo off
otsstart
otsstartsc c:\%temp%\ots\resource.exe c:\%temp%\ots\resource.def
tdstartwu WU1
echo on
【変更後】
echo off
otsstart
otsstartsc -pg c:\%temp%\ots\resource.exe -n リソース定義名 (注)
tdstartwu WU1
echo on
```

(注) リソース定義の移行時に指定したリソース定義名(NAMEに指定した名前)に修正してください。

[移行手順-状態遷移プロシジャの修正:SafeCLUSTERの場合] Solaris

以下のように修正してください。

OTS_LISTに記述しているリソース定義ファイルへのフルパスをリソース定義名に変更してください。以下のように修正してください(太字の部分を修正してください)。

```
RESOURCE_PROG01="/home/ots/otsresource1"
RESOURCE_DEF01="リソース定義名1" (注)
RESOURCE_PARAM01="$RESOURCE_PROG01 $RESOURCE_DEF01"
RESOURCE_PROG01="/home/ots/otsresource2"
RESOURCE_DEF01="リソース定義名2" (注)
RESOURCE_PARAM01="$RESOURCE_PROG01 $RESOURCE_DEF01"
```

(注) リソース定義の移行時に指定したリソース定義名(NAMEに指定した名前)に修正してください。

OTS_INTERSTAGEを以下のように修正してください。

```
45行目付近:
【修正前】
    START_COMMAND="$OTS_HOME/bin/otsstart"
    STOP_COMMAND="$OTS_HOME/bin/otsstop"
【修正後】
    START_COMMAND="$OTS_HOME/bin/otsstartsrc"
    STOP_COMMAND="$OTS_HOME/bin/otsstopsrc"

148行目付近:
【修正前】
    $START_COMMAND -r "$prog $parm" > /dev/null 2>&1
【修正後】
    $START_COMMAND -pg $prog -n $parm > /dev/null 2>&1

333行目付近:
【修正前】
    $STOP_COMMAND -f -r "$def" > /dev/null 2>&1
【修正後】
    $STOP_COMMAND -f -n $def > /dev/null 2>&1
```

OTS_LOG_INTERSTAGE、**OTS_REC_INTERSTAGE**、**OTS_TRN_INTERSTAGE**を以下のように修正してください。

```
148行目付近:
【修正前】
# start System
    $START_COMMAND -s > /dev/null 2>&1
【修正後】
# start System
    $START_COMMAND > /dev/null 2>&1

155行目付近:
【修正前】
    sgetpid $MONPROC_NAME
    $MONPROC_COMMAND -s -a "$START_COMMAND -s" -p $PID -c 0 > /dev/null 2>&1
【修正後】
    sgetpid $MONPROC_NAME
    $MONPROC_COMMAND -s -a "$START_COMMAND" -p $PID -c 0 > /dev/null 2>&1

336行目付近:
【修正前】
# stop System
    $STOP_COMMAND -f -s > /dev/null 2>&1
【修正後】
# stop System
    $STOP_COMMAND -f > /dev/null 2>&1
```

すべてのプロシジャを修正後、SafeCLUSTERに再登録してください。

[移行手順-状態遷移プロシジャの修正:Sun Clusterの場合] Solaris

以下のように修正してください。

OTSRSC_listに記述しているリソース定義ファイルへのフルパスをリソース定義名に変更してください。以下のように修正してください(太字の部分を変更してください)。

```
RESOURCE_PROG01="/home/ots/otsresource1"
RESOURCE_DEF01="リソース定義名1" (注)
RESOURCE_PARAM01="$RESOURCE_PROG01
$RESOURCE_DEF01"RESOURCE_PROG01="/home/ots/otsresource2"
RESOURCE_DEF01="リソース定義名2" (注)
RESOURCE_PARAM01="$RESOURCE_PROG01 $RESOURCE_DEF01"
```

(注) リソース定義の移行時に指定したリソース定義名(NAMEに指定した名前)に修正してください。

INTERSTAGE_start_netを以下のように修正してください。

```
45行目付近:
【修正前】
# start System
$OTS_START_COMMAND -s > /dev/null 2>&1
【修正後】
# start System
$OTS_START_COMMAND > /dev/null 2>&1
```

INTERSTAGE_abort_net、**INTERSTAGE_stop_net**を以下のように修正してください。

```
67行目付近:
【修正前】
OTS_STOP_COMMAND="$OTS_HOME/bin/otsstop"
【修正後】
OTS_STOP_COMMAND="$OTS_HOME/bin/otsstop -f"
```

OTSRSC_start_netを以下のように修正してください。

```
151行目付近:
【修正前】
$OTS_START_COMMAND -r "$prog $parm" > /dev/null 2>&1
【修正後】
$OTS_START_COMMAND -pg $prog -n $parm > /dev/null 2>&1
```

OTSRSC_abort_net、**OTSRSC_stop_net**を以下のように修正してください。

```
129行目付近:
【修正前】
$OTS_STOP_COMMAND -f -r "$def" > /dev/null 2>&1
【修正後】
$OTS_STOP_COMMAND -f -n $def > /dev/null 2>&1
```

すべてのプロシジャを修正後、Sun Clusterに再登録してください。

[注意事項]

- 修正したプロシジャは必ず再登録してください。
- 修正箇所に記載されている行番号はおおよその数値です。

◆自動化プログラムの移行

データベース連携サービスのコマンドを利用して自動化プログラムなどを作成している場合には、“リファレンスマニュアル(コマンド編)”を参照し、コマンドの記述内容を修正してください。

7.8 Portable-ORBの移行

7.8.1 Interstage Application Server V9.1での変更内容

Interstage Application Server V9.1(Interstage V9.1)での変更内容を説明します。

■Internet ExplorerのJava VMを使用するアプレット運用のサポート停止について

Microsoft Java Virtual Machine (MSJVM) のライフサイクルが終了となるため、Interstage V9.0以前の製品で提供していた「Internet ExplorerのJava VMを使用するアプレット運用」のサポートが停止となりました。

Internet ExplorerのJava VMを使用するアプレット運用を行われていた場合は、JBKプラグインを使用するようにHTMLを変更してください。HTMLの変更方法については、“アプリケーション作成ガイド(CORBAサービス編)”の“アプリケーションの開発(Java言語)” – “Javaアプレットの開発” – “開発手順(Portable-ORB)” – “HTMLファイルの記述”を参照してください。

7.8.2 Interstage Application Server V9.0での変更内容

Interstage Application Server V9.0(Interstage V9.0)での変更内容を説明します。

- ・ [アプレットによるporbeditenvの使用停止について](#)
- ・ [無通信監視時間の初期値の変更について](#)

■アプレットによるporbeditenvの使用停止について

セキュリティ強化の観点により、アプレットによるPortable-ORB環境設定コマンド(porbeditenv)の運用は、使用停止となりました。Portable-ORBの動作環境ファイルを編集する場合は、porbeditenvをコマンドで使用してください。

■無通信監視時間の初期値の変更について

クライアントにサーバメソッドが復帰したあと、同一のコネクションを使用して再度リクエスト送信するまでの時間(無通信時間)の初期値が以下のように変更されました。

- ・ **変更前**
初期値: “0” (無通信監視を行わない)
- ・ **変更後**
初期値: “96” (480秒)

Interstage Application Server 8.0以前と同様に、無通信監視を行わずに運用する場合は、Portable-ORB環境設定コマンド(porbeditenv)を使用して[無通信監視時間]に“0”を設定してください。

第8章 その他の機能の移行

ここでは、以下について説明します。

- [MessageQueueDirectorの移行](#)
Interstage Application Server V6.0以前から移行する場合に参照してください。
- [SOAPサービスの移行](#)
Interstage Application Server V7.0以前から移行する場合に参照してください。
- [負荷計測エージェントの移行](#)
Interstage Application Server V9.1以前から移行する場合に参照してください。

8.1 MessageQueueDirectorの移行

■環境定義の変更

Interstage Application Server V7.0(Interstage V7.0)以降では、MessageQueueDirector(MQD)の以下の環境定義が不要となりました。

- TransactionセクションのTransactionMax

この変更により、MQDのシステム環境で同時に発生するトランザクション数の最大は常に1024となります。ただし、TransactionMaxを設定した場合は、旧版と同様にMQDのシステム環境で同時に発生するトランザクション数の最大を設定することが可能です。

上記記述はMQDのシステム環境を新たに作成する場合に関係してくる話です。以前のバージョン・レベルで作成したMQDのシステム環境をInterstage V7.0以降でそのまま使用する場合、環境作成時に指定したTransactionMaxの値が有効となります。

■旧バージョンからの移行について **Windows**

旧バージョンからの環境およびデータ移行はできません。

定義ファイルは旧バージョンと完全に互換性がありますので、新しいマシンで環境を作成する場合には、以前に使用していた定義ファイルを使用して本バージョン・レベルで環境を再作成してください。

■旧バージョンからの移行について **Linux**

MQDはInterstageのアンインストールの際にMQDシステムを削除しません。したがって作成・使用していたMQDシステムは、旧バージョンのInterstageのアンインストールの後に本バージョン・レベルのInterstageのインストールを行うことによって、そのまま使用できます。

■MessageQueueDirector3.0以降からの移行について **Solaris**

MQDはInterstageのアンインストールの際にMQDシステムを削除しません。したがって作成・使用していたMQDシステムは、旧バージョンのInterstageのアンインストールの後に本バージョン・レベルのInterstageのインストールを行うことによって、そのまま使用できます。

■MessageQueueDirector1.1以前からの移行について **Solaris**

MessageQueueDirector(MQD)1.1以前で作成したMQDのシステム環境をそのまま利用することはできません。必ずMQDのシステム環境を削除し、本バージョン・レベルで再作成してください。

8.2 負荷計測エージェントの移行

■提供方法の変更

Interstage Application Server V9.2(Interstage V9.2)以降では、負荷計測エージェントはInterstage Application Server Enterprise Editionでは提供されず、IPCOMで提供されるようになりました。

IPCOM連携機能を使用する場合は、IPCOMで提供されているInterstage Application Server連携機能に対応した負荷計測エージェントを別途インストールしてください。

IPCOMで提供されている負荷計測エージェントの入手方法については、以下のIPCOMのサイトを参照してください。

<http://fenics.fujitsu.com/products/ipcom/>

<http://fenics.fujitsu.com/products/ipcom/contact/> (お問い合わせ)

IPCOMで提供されている負荷計測エージェントのインストール方法については、IPCOMのマニュアルを参照してください。

IPCOM連携機能については、“高信頼性システム運用ガイド”を参照してください。

8.3 SOAPサービスの移行

Interstage Application Server 8.0(Interstage 8.0)から、J2EEの機能の1つとして提供されているWebService for J2EE規約に準拠したWebサービスのWARファイルおよびクライアント(“J2EEのWebサービス機能”)が標準のWebサービスの実行環境となりました。SOAPサービスから、J2EEのWebサービス機能に移行する場合は、以下を参照してください。

- [J2EEのWebサービス機能への移行](#)

Interstage Application Server V7.0(Interstage V7.0)以前からのSOAPサービスのアプリケーション実行環境も、J2EEのWebサービス機能とは別に旧バージョンと同様、SOAPサービスとしてあわせて提供しています。SOAPサービスを引き続き使用する場合は、以下を参照してください。

- [SOAPサービス環境の注意](#)
- [Interstage Application Server V6.0以前からの移行方法](#)

なお、Interstage SOAPサービスは、以下の製品で利用可能です。

- Interstage Application Server Enterprise Edition
- Interstage Application Server Standard-J Edition

8.3.1 J2EEのWebサービス機能への移行

ここでは、J2EEのWebサービス機能への移行について説明します。

- [機能の確認](#)
- [アプリケーションの移行](#)
- [定義の移行・アプリケーションの管理](#)

■機能の確認

以下の機能は、J2EEのWebサービス機能では提供されていません。これらの機能が必要な場合は、J2EEのWebサービス機能ではなく、SOAPサービスを利用してください。

- CORBA/SOAPゲートウェイ機能
- 高信頼性Webサービス(SOAPメッセージの電子署名、XML暗号、SOAPメッセージに対するユーザ認証、送達保証機能)
- Messaging方式のアプリケーション
- V5.0以前のSOAPサービスAPI(パッケージ名が `com.fujitsu.interstage.soap`. で始まるクラス)
- クライアントパッケージ環境でのSSL機能
- SOAP1.2

■アプリケーションの移行

以下の機能は、J2EEのWebサービス機能では提供されていない、または、差異があります。これらの機能を利用している場合は、アプリケーションの修正が必要です。

- [一部のデータ型](#)(Enumerationなど)
- [SOAPサービス固有API](#)(名前が `com.fujitsu.interstage.soapx`. で始まるクラス、またはプロパティ)
- [“Application”以外のサーバアプリケーションのライフサイクル](#)

- ・ クライアントアプリケーションの開発方式
- ・ デフォルトのstyle/useなど
- ・ メソッドのオーバーロード
- ・ voidの返り値のダミー返却
- ・ JavaパッケージとXMLの名前空間の対応
- ・ Fault返却時のクライアントアプリケーションでの例外

◆一部のデータ型

サポートされないデータ型

以下のデータ型は、J2EEのWebサービス機能ではサポートされません。

- ・ 列挙型
- ・ java.util.Vector型
- ・ com.fujitsu.interstage.soap.typesパッケージに含まれるクラス

SOAPサービスでこれらのデータ型を利用していた場合は、“J2EE ユーザーズガイド”の“Webサービスの開発”を参照して、これらのデータ型を利用しないインタフェースに変更してください。

マッピングに変更があるデータ型

以下のデータ型は、XMLのデータ型とのマッピングがJ2EEのWebサービス機能とSOAPサービスで差異があります。

Webサービスアプリケーションのパラメータ に使用するJavaのデータ型	XMLで使用されるデータ型	
	SOAPサービス	J2EEのWebサービス機能
byte[]	xsd:byteの配列	xsd:base64Binary
java.math.BigDecimal	xsd:decimal	soapenc:decimal ※
java.math.BigInteger	xsd:integer	soapenc:integer ※

(※のデータ型はencoded利用の場合のデータ型です)

SOAPサービスでこれらのデータ型を利用していた場合は、“J2EE ユーザーズガイド”の“Webサービスの開発”を参照して、アプリケーションまたはインタフェースを変更してください。

構造体型とBean型の混成とみなされるデータ型の注意

構造体型のpublicフィールドと、フィールド名と同名のプロパティのsetメソッドまたはgetメソッドの片方のみを持つデータ型は、J2EEのWebサービス機能とSOAPサービスで扱いに差異があります。以下について確認し、該当する場合は必要に応じてデータ型を修正してください。

- ・ フィールド名と同名のプロパティのsetメソッドのみを持ち、getメソッドを持たない場合は、J2EEのWebサービス機能では該当のフィールド(プロパティ)は値取得不可のプロパティとみなされて、値は送信されません。
- ・ フィールド名と同名のプロパティのgetメソッドのみを持ち、setメソッドを持たない場合は、J2EEのWebサービス機能では該当のフィールド(プロパティ)は値設定不可のプロパティとみなされて、データを受信した場合でも値が設定されません。



例

下記のデータ型を使用した場合、J2EEのWebサービス機能ではnameの値は送信されません。

```
public class Human {
    public String name;           //nameのpublicフィールド
    public void setName(String n) { name = n; } //nameのsetメソッド
    //  ※getName()メソッドなし //
    public Human() { }
    public Human(String n, int a) { name=n; age=a; }
```

}

添付ファイル型

添付ファイル型のデータを使用している場合、以下について確認し対応を行ってください。

- SOAPサービスの添付ファイル機能は、WS-I Attachments Profileに対応していません。移行前のSOAPサービスと同様の通信方法にするには、添付ファイルに対応するXMLのデータ型には`apache:soap`名前空間のXMLデータ型を使用してください。添付ファイルに対応するXMLのデータ型の種類は、`iswsgen wsdl`コマンドの`-attachmentsType`オプションで指定します。詳細については、“リファレンスマニュアル(コマンド編)”の“`iswsgen`”を参照してください。
- SOAPサーバアプリケーションでは、受信した添付ファイルのパラメタをリクエスト処理完了後も保持することが可能でした。Interstage Webサービスの既定では、リクエスト処理完了後は、添付ファイルのパラメタはリソース解放のため利用できなくなる場合があります。リクエスト処理完了後もパラメタのまま保持する必要がある場合は、“レスポンス返却時の、Webサービスアプリケーションで受信した添付ファイルデータ削除(資源解放)”の設定を変更して下さい。詳細については、“J2EE ユーザーズガイド”の“Webサービス設定ファイル”を参照してください。

◆SOAPサービス固有API

プログラム修正	機能	SOAPサービス	J2EEのWebサービス機能
推奨	クライアントでの、プロキシを経由した接続の設定	名前が <code>com.fujitsu.interstage.soapx.proxy</code> で始まるプロパティをプログラムで設定します。	“J2EE ユーザーズガイド”の“Webサービスアプリケーションの運用”を参照してください。 (SOAPサービスでの設定方法は無視されます)
推奨	クライアントでの、接続タイムアウトの設定	<code>com.fujitsu.interstage.soapx.socket.timeout</code> プロパティをプログラムで設定します。デフォルトは5分です。	“J2EE ユーザーズガイド”の“Webサービスの開発”を参照してください。 デフォルトは10分です。 (SOAPサービスでの設定方法は無視されます)
必須 (該当する場合)	クライアントでの、セッション継続の設定	<code>com.fujitsu.interstage.soapx.client.Stub#setMaintainSession()</code> メソッドで設定します。	“J2EE ユーザーズガイド”の“Webサービスの開発”を参照してください。 (SOAPサービスでの設定メソッドは提供されません)
必須 (該当する場合)	JAX-RPC規定(<code>javax.xml.rpc.holders</code> パッケージ)以外のHolderクラス	<code>com.fujitsu.interstage.soapx.holders</code> パッケージでHolderクラスが提供されています。	<code>javax.xml.rpc.holders.Holder</code> インタフェースをimplementsして、アプリケーションで作成してください。

◆“Application”以外のサーバアプリケーションのライフサイクル

プログラム修正	機能	SOAPサービス	J2EEのWebサービス機能
必須 (該当する場合)	サーバアプリケーションのライフサイクルの設定	Webサービス情報に設定します。 “Application”、“Session”、“Request”からサーバアプリケーションのインスタンスの管理方式を選択します。	SOAPサービスにおいてApplicationが指定された場合と同等になります。リクエスト単位でサーバアプリケーションのインスタンス化が必要な場合、アプリケーションをラップする新たなアプリケーションクラスを作成してインスタンス生成と委譲するなどの対処が必要です。セッションを利用する場合、アプリケーションのフィールドではなく、

プログラム修正	機能	SOAPサービス	J2EEのWebサービス機能
			HttpSessionオブジェクトに情報を格納します。詳細については、“J2EE ユーザーズガイド”の“Webサービスの開発”を参照してください。 (SOAPサービスでの設定方法は提供されません)

◆クライアントアプリケーションの開発方式

SOAPサービスでは、統合開発環境ツール(IDE)を利用したスタブ方式と上級者向けの複雑なAPIを使用するDII方式の2つの方式がサポートされています。J2EEのWebサービス機能では、開発容易性に優れたスタブ方式をInterstage Application Serverとしてサポートしています。SOAPサービスでDII方式のクライアントアプリケーションを利用していた場合は、SOAPのクライアント部分をスタブ方式に修正してください。

また、スタブはJ2EEのWebサービス機能で生成したものに置き換える必要があります。

プログラム修正	機能	SOAPサービス	J2EEのWebサービス機能
必須	ロケータオブジェクト (Serviceオブジェクト)の取得	javax.xml.rpc.ServiceFactory#createService(javax.xml.namespace.QName serviceName)メソッドを使用します。	javax.xml.rpc.ServiceFactory#loadService(java.lang.Class generatedServiceClass)メソッドを使用します。

◆デフォルトのstyle/useなど

SOAPサービスのRPCアプリケーションでは、通信はWSDLのrpc/encoded相当の方式で行われます。J2EEのWebサービス機能は、rpc/encoded、rpc/literal、document/literalのWSDLをサポートしており、デフォルトはdocument/literalです。

またJ2EEのWebサービス機能ではJavaパッケージと名前空間名の対応、ユーザ定義型のJavaクラス名とXMLデータ型のローカル名の対応、および、オペレーションのパラメタの名前にデフォルトを持っています。デフォルトと移行前の値が異なる場合はiswsgenコマンドに対する指定が必要です。オペレーションのパラメタの名前(rpc/encodedのWSDLのpart名)は、WSDLの該当部分を書き換えてください。

詳細については、“リファレンスマニュアル(コマンド編)”の“iswsgen”を参照してください。

◆メソッドのオーバーロード

J2EEのWebサービス機能では、Webサービスアプリケーションのメソッドのオーバーロード(オペレーション名が同一で引数が異なる複数のメソッド)をサポートしていません。SOAPサービスでメソッドのオーバーロードを利用していた場合は、アプリケーションをラップするなどして、メソッドのオーバーロードを使用しないインタフェースに変更してください。

◆voidの戻り値のダミー返却

SOAPサービスでは、戻り値がvoidのアプリケーションについて、プロパティ指定により戻り値としてダミーの値が返却するオプションが提供されていました。SOAPサービスで本オプションを使用していた場合は、アプリケーションをラップするなどして、戻り値をvoidからintに変更して、ダミーの値0を常に返却するように変更してください。

◆JavaパッケージとXMLの名前空間の対応

J2EEのWebサービス機能では、Webサービスアプリケーションで使用されるJavaのパッケージとXMLの名前空間が対応付けられます。移行前の名前空間の利用を継続するには、“リファレンスマニュアル(コマンド編)”の“iswsgen”を参照し、オプションで名前空間名を指定してください。Javaのパッケージと名前空間が1対1に対応していない場合は、既存のクラスをラップするクラスを作成するなどして1対1に対応させてください。

◆Fault返却時のクライアントアプリケーションでの例外

SOAPサービスでは、WebサービスからFaultが返却されると、クライアントアプリケーションにはFault情報を取得できるAPIを持つjavax.xml.rpc.soap.SOAPFaultExceptionがthrowされました。

それに対して、J2EEのWebサービス機能ではその様なAPIを持たないjava.rmi.RemoteExceptionがthrowされます。同例外では、ス

タクトレースにFault情報も含んで出力されます。

JAX-RPC仕様上は、両方の例外がthrowされる可能性があるため、両方のExceptionをcatchしてログにスタックトレース・Fault情報を出力するようにしてください。

■定義の移行・アプリケーションの管理

◆アプリケーションの形態、配備方法

SOAPサービスでは、サーバアプリケーションは、任意のJARファイルにまとめるなどして、FTPなどでサーバに転送し、IJServer(Servletコンテナ)のクラスパスに設定する必要があります。また、あらかじめIJServerにはSOAPサービスの環境を構築しておく必要があります。

J2EEのWebサービス機能では、WebサービスアプリケーションはWARファイルに含め、通常のWARファイルと同様にInterstage管理コンソールなどからIJServerに配備することで利用可能になります。IJServerに事前の特別の環境構築は不要です。WebサービスアプリケーションのWARファイルの作成については、“J2EE ユーザーズガイド”の“Webサービスの開発”を参照してください。

◆アプリケーションの定義情報

SOAPサービスでは、Webサービス情報として、アプリケーションの定義情報が管理されており、Webサービス情報編集ツール(GUI)またはsoapmodifyddコマンドと記述ファイルで環境に登録・更新を行います。

J2EEのWebサービス機能では、Webサービス情報に相当する情報をdeployment descriptorとして記述し、アプリケーションのWARファイルに含めます。deployment descriptorの記述およびWARファイルの作成については、“J2EE ユーザーズガイド”の“Webサービスの開発”を参照してください。

8.3.2 SOAPサービス環境の注意

J2EEのWebサービス機能の提供に伴い、SOAPサービスを利用する場合の環境について以下の注意事項があります。

- ・ [インストール](#)
- ・ [環境設定](#)

■インストール

Interstage V7.0以前では、SOAPサービスはInterstage Application Serverのインストールで、標準でインストールされていましたが、Interstage 8.0以降では、標準ではインストールされません。インストーラのメニューからSOAPサービスを明示的に選択してインストールしてください。

Linux

RHEL-AS4(x86)/AS4(EM64T)およびRHEL5(x86)/(Intel64)で使用することができます。

■環境設定

Interstage V9.1.0の“SOAPサービス ユーザーズガイド”の“環境設定”を参照し、マニュアルに従って環境構築を行ってください。

Windows

Interstage V7.0以前ではSOAPサービスのインストール時に、システム環境変数のCLASSPATHにSOAPサービスのライブラリが設定されていましたが、Interstage 8.0以降でSOAPサービスをインストールしても設定されません。

SOAPサービスを使用する場合に、コマンドプロンプトなどで設定するか、SOAPサービスで動作するアプリケーションの起動バッチなどに記述してください。システム環境変数のCLASSPATHにSOAPサービスのライブラリを設定すると、J2EEのWebサービス機能の動作に悪影響を与えます(IJServer上で動作するアプリケーションの場合は、システム環境変数のCLASSPATHの影響を受けないため、問題ありません)。

◆IJServer環境について



注意

Interstage Java EE (IJServerクラスタ) への移行はできません。J2EE環境へ移行してください。J2EE環境については“J2EE ユーザーズガイド”を参照してください。

SOAPサーバアプリケーション

互換機能のSOAPのサーバアプリケーションを利用する場合は、IJServerに対して、Interstage管理コンソールの[システム] > [ワークユニット] > “ワークユニット名” > [環境設定]タブ > [共通定義]の“コンテナのWebサービス機能”を“無効”に設定してから、従来の設定(SOAP.WARの配備など)を行ってください。

“コンテナのWebサービス機能”を“無効”に設定した場合、そのIJServerではJ2EEのWebサービス機能は利用できません。

また、Interstage V6.0およびInterstage V7.0では、IJServerに対して、サーバアプリケーションおよびユーザ定義型のクラスファイルを配置したパス名もしくはJARファイル名を指定していましたが、Interstage 8.0からは、これらに加えてInterstage V9.1.0の“SOAPサービスユーザーズガイド”に従ってSOAPサービスのJARも指定してください。

SOAPクライアントアプリケーション

IJServer上で、互換機能のクライアントアプリケーションを利用する場合も同様にIJServerに対して、Interstage管理コンソールの[システム] > [ワークユニット] > “ワークユニット名” > [環境設定]タブ > [共通定義]の“コンテナのWebサービス機能”を“無効”に設定してから、従来の設定と、SOAPサービスのJARのクラスパス指定を行ってください。

◆IJServer外環境について



Interstage Java EE (Java EEアプリケーションクライアント) への移行はできません。J2EE環境へ移行してください。J2EE環境については“J2EE ユーザーズガイド”を参照してください。

Interstage Application Serverがインストールされた環境で、IJServer上ではないJavaプログラムで互換機能のクライアント側機能を利用する場合は、Interstage V9.1.0の“SOAPサービスユーザーズガイド”に従ってプログラムのクラスパスにissoap.jarなどを指定してください。isws.jarは設定しないでください。

Windows(R)では、システム環境変数にisws.jarが設定されている場合があるので、注意してください。

◆CORBA/SOAPクライアントゲートウェイについて

SOAPサービスのCORBA/SOAPクライアントゲートウェイは、Interstage V9.1.0の“SOAPサービス ユーザーズガイド”に従ってクラスパスにissoap.jarなどを指定してください。

isws.jarは設定しないでください。

Windows(R)では、CORBA/SOAPクライアントゲートウェイは、システム環境変数を参照して動作します。システム環境変数のクラスパスにissoap.jarなどを指定してください。システム環境変数にisws.jarが設定されている場合、isws.jarの設定を削除する必要があります。なお、システム環境変数にisws.jarではなくissoap.jarなどのSOAPサービスのJARファイルを指定した場合、システム環境変数を参照している環境では、J2EEのWebサービス機能は利用できなくなります。

◆クライアントパッケージ環境について

クライアントパッケージがインストールされた環境で、互換機能のクライアント側機能を利用する場合は、Interstage V9.1.0の“SOAPサービス ユーザーズガイド”に従ってプログラムのクラスパスにissoap.jarなどを指定してください。

isws.jarは設定しないでください。

Windows(R)では、システム環境変数にisws.jarが設定されている場合があるので、注意してください。

◆GUIによるWebサービス情報編集ツール内の「Webサービス情報生成」ボタンについて

「Webサービス情報生成」ボタンを使用してWebサービス情報を生成する際、システム環境変数のCLASSPATHからの検索は行われなくなりましたので、事前にWebサービスアプリケーションファイル名を指定して「Webサービス情報生成」ボタンの操作を行ってください。Webサービスアプリケーションファイル名にはSOAPサーバアプリケーションJARファイルまたはクラスファイルのパス名を指定してください。複数のクラスファイルを指定する必要がある場合は、JARファイルの指定をお奨めします。

8.3.3 Interstage Application Server V6.0以前からの移行方法

Interstage Application Server V6.0以前のバージョン・レベルのSOAPサービスから、本製品のSOAPサービスに移行する場合には、以下の対処が必要です。

- [CORBA/SOAPゲートウェイの移行](#)

Interstage Application Server V6.0/V5.1のCORBA/SOAPゲートウェイを移行する場合に必要な対処です。

- [V5.1 RPCアプリケーションの移行](#)

Interstage Application Server V5.1のRPCアプリケーションを移行する場合に必要な対処です。

- [V5.0/V4 RPCアプリケーションのJavaAPI](#)

Interstage Application Server V5.0/V4.xのRPCアプリケーションのJavaAPIについては、添付のJavaDOCを参照してください。

- [特定のデータ型を使用するSOAPサーバアプリケーションの対応](#)

特定のデータ型を使用するInterstage Application Server V5.0以前のSOAPサーバアプリケーションを移行する場合に必要な対処です。

- [Webサービス情報編集ツール](#)

Interstage Application Server V6.0以降のWebサービス情報編集ツールは、Interstage管理コンソールから起動します。Interstage管理コンソールからWebサービス情報編集ツールを使用する場合は、Interstage Application Server V4.xのディレクトリサービス(InfoDirectory)で管理していたユーザ情報をInterstage管理コンソールの認証方式に変更する必要があります。

- [Webサービス情報管理ファイルの移行](#)

- ー Interstage Application Server V5.1からの移行

Webサービス情報管理ファイルが、以下のディレクトリ下にある場合は移行の必要はありません。異なる場合はInterstageを停止した状態で複写もしくは移動してください。

Windows

%IS_HOME%\F3FMsoap\etc\DeployService

Solaris

Linux

/opt/FJSVsoap/etc/DeployService

- ー Interstage Application Server V5.0/V4.xからの移行

以前のバージョン・レベルで開発したWebサービスアプリケーション、CORBA/SOAPゲートウェイを動作させるためには、Webサービス情報管理ファイルの移行が必要です。以下の2つの移行方法について説明します。

- [Webサービス情報管理ファイルの移行－バックアップコマンドによる移行](#)
- [Webサービス情報管理ファイルの移行－soapdd2wsddコマンドによる移行](#)



ポイント

Webサービス情報管理ファイルの移行完了後、Webサービスコンテナの環境構築を行ってください。詳細についてはInterstage V9.1.0の“SOAPサービス ユーザーズガイド”の“サーバシステムの環境構築”を参照してください。

J2EEのWebサービス機能の提供に伴うSOAPサービスの注意事項については、“[8.3.2 SOAPサービス環境の注意](#)”を参照してください。

8.3.4 CORBA/SOAPゲートウェイの移行

以下の条件に該当するInterstage Application Server V6.0/V5.1のCORBA/SOAPゲートウェイを移行する場合、下記の対処を実施してください。

■条件

下記の条件に該当するCORBAの構造体型/例外型、または、EJBアプリケーションのクラス型/例外型を、SOAPで送信または受信している場合。

- 名前の最初の1文字目が小文字であり、2文字目が大文字である公開メンバを持つ

例1) CORBAの構造体型の場合 (IDL)


```

module xxxModule {
    struct xxxStruct {
        long    aParam; /* 1 文字目が小文字であり、2 文字目が大文字*/
    };
};

```

例2) EJBアプリケーションのクラス型の場合 (Java)

```

public class xxxData implements java.io.Serializable {
    public String aParam; /* 1 文字目が小文字であり、2 文字目が大文字*/
}

```

■対処方法

本バージョン・レベルでEJBアプリケーションを再配備またはIDLコンパイルにおいてCORBA/SOAPゲートウェイを再生成した場合、SOAPのインタフェース上で該当するメンバ名が移行前と異なったメンバ名で生成されます。再配備または再生成/再IDLコンパイルを行う場合は、下記の**A)**または**B)**のいずれかの方法で対処してください。

A) 該当のCORBA/SOAPゲートウェイのSOAP通信先で該当のメンバ名を修正します。

SOAPの通信先で該当する構造体のメンバ名の1文字目を大文字として処理しているかを確認し、該当する場合は1文字目を小文字として処理するように修正します。

例) aParam → AParam

B) 再配備または再生成/IDLコンパイルの前に、IDL上(CORBA)またはJava上(EJB)の該当のメンバ名を以下のように変更します。

EJBアプリケーションを再配備またはCORBA/SOAPゲートウェイの再生成およびIDLコンパイルを行う前に、IDL上(CORBA)またはJava上(EJB)のメンバ名の1文字目を大文字に変更します。

例) aParam → AParam

8.3.4.1 V5.1 RPCアプリケーションの移行

Interstage Application Server V5.1(Interstage V5.1)では、以下のMIMEタイプの添付ファイルを受信した場合、javax.activation.DataHandlerオブジェクトとして取得後、同オブジェクトのgetContentメソッドからjava.io.InputStreamオブジェクトが返り値として返りました。

- image/jpeg, image/gif
- multipart/*
- text/xml, application/xml

Interstage Application Server V5.1.1(Interstage V5.1.1)以降では、それぞれ、以下のオブジェクトが返り値として返ります。

- java.awt.Imageオブジェクト
- javax.mail.internet.MimeMultipartオブジェクト
- javax.xml.transform.Sourceオブジェクト

Interstage V5.1.1以降でjavax.activation.DataHandlerオブジェクトから添付ファイルのデータをjava.io.InputStreamオブジェクトとして取得するには、javax.activation.DataHandlerオブジェクトに対して、getContentメソッドではなく、以下のメソッドを使用してください。

```

public InputStream getInputStream() throws IOException

```

上記のようにjavax.activation.DataHandlerオブジェクトの動作が異なるため、Interstage V5.1で開発した添付ファイルを扱うRPCアプリケーションは、以下のどちらかの方法で移行する必要があります。

■RPCアプリケーションを移行する場合

以下にRPCアプリケーションを移行するコーディング例を示します。

移行前

```
DataHandler dh = ...;    // 引数
java.io.InputStream is = (java.io.InputStream) dh.getContent();
```

移行後

```
DataHandler dh = ...;    // 引数
java.io.InputStream is = dh.getInputStream();
```

■Interstage Application Server V5.1互換環境で実行する場合

Interstage管理コンソールの[システム] > [ワークユニット] > [IJSerwerワークユニット名を選択] > [環境設定]タブ > [ワークユニット設定]の“JavaVMオプション”に以下のプロパティを設定します。

プロパティ	指定可能な値
- Dcom.fujitsu.interstage.soapx.DataHandlerMode=値	• V51(英大文字・英小文字の区別なし): ISAPS 5.1 の javax.activation.DataHandler オブジェクトの getContentメソッドと同じ返り値が返ります。 • その他(省略値): 設定していないものとみなします。

8.3.4.2 特定のデータ型を使用するSOAPサーバアプリケーションの対応

以下の条件に該当するInterstage Application Server V5.0(Interstage V5.0)以前のSOAPサーバアプリケーションを移行する場合、下記のA)またはB)のいずれかの方法で対処してください。

■条件

以下のデータ型を受信している場合、または以下のデータ型を構造体のメンバまたは配列で使用している場合。

- com.fujitsu.interstage.soap.util.Base64
- com.fujitsu.interstage.soap.util.Base64Binary
- com.fujitsu.interstage.soap.util.UnsignedInt
- com.fujitsu.interstage.soap.util.UnsignedShort
- com.fujitsu.interstage.soap.util.UnsignedLong
- com.fujitsu.interstage.soap.util.UnsignedByte

■対処方法

下記のA)またはB)のいずれかの方法で対処してください。

A) 該当するRPCサーバアプリケーションのWebサービス情報を再登録する。

1. 該当するRPCサーバアプリケーションのWebサービス情報ファイルを用意します。
Webサービス情報ファイルがない場合は、以下のようにして取得してください。

例)

```
soapmodifydd -query -url http://localhost:12000/soap_admin/servlet/WSContainer -urn urn:old-app > oldappwsdl.xml
```

soapmodifyddコマンドについては、“リファレンスマニュアル(コマンド編)”を参照してください。

2. 1.のWebサービス情報ファイル(上例の場合oldappwsdl.xml)のserviceタグ配下に以下の内容のtypeMappingタグを追加します。

```
<typeMapping qname="xsd:base64Binary"
```

```

xmlns:xsd="http://www.w3.org/2001/XMLSchema"
languageSpecificType="com.fujitsu.interstage.soap.util.Base64Binary"
serializer="com.fujitsu.interstage.soapx.encoding.ser.V50Base64SerializerFactory"
deserializer="com.fujitsu.interstage.soapx.encoding.ser.V50Base64DeserializerFactory"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
<typeMapping qname="soapenc:base64"
xmlns:soapenc="http://schemas.xmlsoap.org/soap/encoding/"
languageSpecificType="com.fujitsu.interstage.soap.util.Base64"
serializer="com.fujitsu.interstage.soapx.encoding.ser.V50Base64SerializerFactory"
deserializer="com.fujitsu.interstage.soapx.encoding.ser.V50Base64DeserializerFactory"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
<typeMapping qname="xsd:unsignedInt"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
languageSpecificType="com.fujitsu.interstage.soap.util.UnsignedInt"
serializer="com.fujitsu.interstage.soapx.encoding.ser.V50UnsignedNumberSerializerFactory"
deserializer="com.fujitsu.interstage.soapx.encoding.ser.V50UnsignedNumberDeserializerFactory"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
<typeMapping qname="xsd:unsignedShort"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
languageSpecificType="com.fujitsu.interstage.soap.util.UnsignedShort"
serializer="com.fujitsu.interstage.soapx.encoding.ser.V50UnsignedNumberSerializerFactory"
deserializer="com.fujitsu.interstage.soapx.encoding.ser.V50UnsignedNumberDeserializerFactory"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
<typeMapping qname="xsd:unsignedLong"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
languageSpecificType="com.fujitsu.interstage.soap.util.UnsignedLong"
serializer="com.fujitsu.interstage.soapx.encoding.ser.V50UnsignedNumberSerializerFactory"
deserializer="com.fujitsu.interstage.soapx.encoding.ser.V50UnsignedNumberDeserializerFactory"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />
<typeMapping qname="xsd:unsignedByte"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
languageSpecificType="com.fujitsu.interstage.soap.util.UnsignedByte"
serializer="com.fujitsu.interstage.soapx.encoding.ser.V50UnsignedNumberSerializerFactory"
deserializer="com.fujitsu.interstage.soapx.encoding.ser.V50UnsignedNumberDeserializerFactory"
encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" />

```

3. soapmodifydd コマンドで編集したWebサービス情報ファイルを登録し直します。

例)

```

soapmodifydd -set -url http://localhost:12000/soap_admin/servlet/WSContainer oldappwsdl.xml

```

4. Webサービスコンテナを運用モード(employ)で運用している場合は、Webサービスコンテナが動作しているワークユニットを再起動します。

B) 条件に該当するSOAPRPCサーバアプリのパラメタのデータ型を、Interstage V5.1以降で提供するデータ型に変更する。

Interstage V5.1以降で提供されているデータ型については、Interstage V9.1.0の“SOAPサービス ユーザーズガイド”の“サポートされるデータ型”を参照してください。

例)

変更前

```

int serviceMethod(com.fujitsu.interstage.soap.util.Base64 bindata,
com.fujitsu.interstage.soap.util.UnsignedInt serialNo)

```

変更後

```

int serviceMethod(byte[] bindata,

```

com.fujitsu.interstage.soapx.types.UnsignedInt serialNo)

8.3.4.3 Webサービス情報管理ファイルの移行ーバックアップコマンドによる移行

バックアップコマンドによる移行は、バックアップ(isssoapbackup、issoaprestore)コマンド実行時に、Interstage Application Server V5.0/V4.x(Interstage V5.0/V4.x)と同じディレクトリにWebサービス情報管理ファイルを移行できます。

バックアップコマンドの詳細については、“リファレンスマニュアル(コマンド編)”を参照してください。

■はじめに

Webアプリケーション環境設定(web.xml)で指定したWebサービス情報管理ファイルが、以下のディレクトリ下にある場合は、バックアップコマンドで移行可能です。異なる場合はバックアップコマンドでInterstage SOAPサービス資源の移出と移入を行った後、“[8.3.4.4 Webサービス情報管理ファイルの移行ーsoapdd2wsddコマンドによる移行](#)”を行ってください。

Windows

```
%IS_HOME%\F3FMsoap\etc\DeployService
```

Solaris

Linux

```
/opt/FJSVsoap/etc/DeployService
```

■移行手順

1) 環境のバックアップ Windows

バックアップ先のディレクトリがC:\Backupの場合の操作例を以下に示します。

1. Webサーバを停止します。
2. Interstage管理コンソール用Servletサービスを停止します。
C:\Interstage\F3FMjs2su\bin\jssvstop
3. バックアップ用ディレクトリを作成します。
mkdir C:\Backup
4. isssoapbackupコマンドを実行して、Webサービス資源ファイルをバックアップします。
isssoapbackup -d C:\Backup

1) 環境のバックアップ Solaris Linux

バックアップ先のディレクトリが/Backupの場合の操作例を以下に示します。

1. Webサーバを停止します。
2. Interstage管理コンソール用Servletサービスを停止します。
/opt/FJSVjs2su/bin/jssvstop
3. バックアップ用ディレクトリを作成します。
mkdir /Backup
4. isssoapbackupコマンドを実行して、Webサービス資源ファイルをバックアップします。
isssoapbackup -d /Backup

2) 環境のリストア Windows

バックアップ先のディレクトリがC:\Backupの場合の操作例を以下に示します。

1. Webサーバを停止します。
2. Interstageを停止します。
isstop -f
3. Interstage管理コンソール用Servletサービスを停止します。
C:\Interstage\F3FMjs2su\bin\jssvstop
4. isoaprestoreコマンドを実行して、ISAPS5.0/4のWebサービス資源ファイルを新バージョンにリストアします。
isoaprestore -d C:\Backup

2) 環境のリストア **Solaris** **Linux**

バックアップ先のディレクトリが/Backupの場合の操作例を以下に示します。

1. Webサーバを停止します。
2. Interstageを停止します。
isstop -f
3. Interstage管理コンソール用Servletサービスを停止します。
/opt/FJSVjs2su/bin/jssvstop
4. isoaprestoreコマンドを実行して、ISAPS5.0/4のWebサービス資源ファイルを新バージョンにリストアします。
isoaprestore -d /Backup

■移行結果の確認

Webサービス情報管理ファイルの移行結果の確認

以下のファイルにWebサービス情報管理ファイルのWebサービス毎に移行結果が出力されます。

Windows

```
C:\Interstage\F3FMsoap\log\logdd2wsdd.txt
```

Solaris

Linux

```
/opt/FJSVsoap/log/logdd2wsdd.txt
```

同名のWebサービス識別名の有無の確認

同名のWebサービス識別名が存在していた場合、移行しなかったWebサービスのWebサービス情報をファイルに出力します。

Windows

```
C:\Interstage\F3FMsoap\log\log_same.wsdd
```

Solaris

Linux

```
/opt/FJSVsoap/log/log_same.wsdd
```

8.3.4.4 Webサービス情報管理ファイルの移行－soapdd2wsddコマンドによる移行

ここでは、Interstage Application Server V5.0/V4.x(Interstage V5.0/V4.x)の他システムからsoapdd2wsddコマンドによる移行手順について説明します。

なおsoapdd2wsddコマンドの詳細については、“リファレンスマニュアル(コマンド編)”を参照してください。

■はじめに

Interstage V5.0/V4.xのシステム運用で使用していた(移行する)Webサービス情報管理ファイルのディレクトリ名を確認しておいてください。

■移行手順 Windows

以下に移行手順を示します。

1. Interstage V5.0/V4.xシステムのWebサーバを停止します。
2. Interstage V5.0/V4.xのServletサービスを停止します。
3. ネットワーク(FTP等)を経由、もしくは外部媒体を使用してWebサービス情報管理ファイルをInterstage V5.0/V4.xシステムから新バージョンのシステムに転送します。
4. 本バージョン・レベルのWebサーバを停止します。
5. 本バージョン・レベルのInterstageを停止します。
`isstop -f`
6. 本バージョン・レベルのInterstage管理コンソール用Servletサービスを停止します。
`C:\Interstage\F3FMjs2su\bin\jssvstop`
7. soapdd2wsddコマンドを使用して、ISAPS5.0/4のWebサービス情報を変換します。変換先のWebサービス情報管理ファイルに、変換元と同名のWebサービス識別名が存在した場合は、上書きせずにsoapdd2wsddコマンド実行ディレクトリにlog_same.wsddを出力します。
`soapdd2wsdd -none -i C:\tmp\deppath -sf log_same.wsdd -info C:\tmp\logdd2wsdd.txt`

■移行手順 Solaris Linux

以下に移行手順を示します。

1. Interstage V5.0/V4.xシステムのWebサーバを停止します。
2. Interstage V5.0/V4.xシステムのServletサービスを停止します。
3. ネットワーク(FTP等)を経由、もしくは外部媒体を使用してWebサービス情報管理ファイルをInterstage V5.0/V4.xシステムから新バージョンのシステムに転送します。
4. 本バージョン・レベルのWebサーバを停止します。
5. 本バージョン・レベルのInterstageを停止します。
`isstop -f`
6. 本バージョン・レベルのInterstage管理コンソール用Servletサービスを停止します。
`/opt/FJSVjs2su/bin/jssvstop`

7. soapdd2wsddコマンドを使用して、Interstage V5.0/V4.xのWebサービス情報を変換します。変換先のWebサービス情報管理ファイルに、変換元と同名のWebサービス識別名が存在した場合は、上書きせずにsoapdd2wsddコマンド実行ディレクトリにlog_same.wsddを出力します。
soapdd2wsdd -none -i /tmp/deppath -sf log_same.wsdd -info /tmp/logdd2wsdd.txt

■移行結果の確認

Webサービス情報管理ファイルの移行結果の確認

soapdd2wsddコマンドの-infoオプションで指定したファイルに移行結果が出力されます。-infoオプションを省略した場合は、標準出力に出力されます。

同名のWebサービス識別名の有無の確認

同名のWebサービス識別名が存在していた場合、移行しなかったWebサービス情報をファイルに出力します。

移行モードが-noneオプションと-sfオプションを指定した場合は、同名のWebサービス識別名が存在していたために移行しなかったWebサービス情報を-sfオプションで指定したファイルに出力します。

付録A V5.1以前のServletサービス環境定義の移行

本バージョン・レベルでは、V5.1以前のServletサービスを同梱していません。“[4.2 Servletサービス\(Tomcat 5.5ベースのサーブレット実行環境\)への移行](#)”を参照して、移行してください。

ここでは、V5.1以前のServletサービスの環境定義をTomcat5.5ベースのServletサービスへ移行する際の定義項目の対応を示します。

Fujitsu XMLプロセッサを使用する場合は、“J2EEユーザーズガイド”の“動作環境のカスタマイズと確認”―“動作環境のカスタマイズ”―“Fujitsu XMLプロセッサを使用する場合の設定”を参照してください。

A.1 JServlet環境定義の移行

以下にV5.1以前のServletサービスにおけるJServlet環境定義ファイルの定義項目と、Interstage管理コンソールの対応表を示します。なお、isj2eeadminコマンドを使用して、以下の各設定を行うこともできます。詳細は、“リファレンスマニュアル(コマンド編)”を参照してください。

V5.1以前のServletサービス	Interstage管理コンソール
[containername].containerconf	必要ありません。
[containername].bin	ありません。V9以降のIJServer(Tomcat5.5ベースのServletサービス)では、JDK5.0のみサポートします。
[containername].bin.parameters	[ワークユニット] > [新規作成]タブ > [詳細設定] > [ワークユニット設定] > [JavaVMオプション] (注1)(注2)
[containername].env	[ワークユニット] > [新規作成]タブ > [詳細設定] > [ワークユニット設定] > [クラスパス]、[パス]および[環境変数] (注1)(注3)
[containername].port	[ワークユニット] > [新規作成]タブ > [詳細設定] > [Servletコンテナ設定] > [ポート番号] (注1)
[containername].log	[ワークユニット] > [新規作成]タブ > [詳細設定] > [ワークユニット設定] > [ログ出力ディレクトリ] (注1) [ワークユニット] > “ワークユニット名” > [ログ参照]タブ > [起動情報]から参照できます。
[containername].startup	[ワークユニット] > [新規作成]タブ > [詳細設定] > [ワークユニット設定] > [ワークユニット自動起動] (注1)
[containername].ha.shutdown	ありません。
[containername].execdir	[ワークユニット] > [新規作成]タブ > [詳細設定] > [ワークユニット設定] > [カレントディレクトリ] (注1)
[containername].ipaddress	(注4) [ワークユニット] > [新規作成]タブ > [詳細設定] > [Servletコンテナ設定] > [IPアドレス] (注1)
default.bin default.bin.parameters default.env default.execdir	ありません。
container.checkCount	[ワークユニット] > [新規作成]タブ > [詳細設定] > [ワークユニット設定] > [リトライカウント] (注1)
container.checkFrequency	[ワークユニット] > [新規作成]タブ > [詳細設定] > [ワークユニット設定] > [リトライカウントリセット時間] (注1)
container.restart	trueの場合:[リトライカウント]を“0”または“2”以上、[リトライカウントリセット時間]を“1”以上に設定してください。 falseの場合:[リトライカウント]を“1”、[リトライカウントリセット時間]を“0”に設定してください。

V5.1以前のServletサービス	Interstage管理コンソール
container.shutdownInterval	[ワークユニット] > [新規作成]タブ > [詳細設定] > [ワークユニット設定] > [プロセス強制停止時間] (注1)
container.uid	IJServerワークユニットを起動したユーザの権限でコンテナのプロセスを起動します。 特定ユーザの権限でIJServerワークユニットを自動起動する場合には、以下に示すInterstage管理コンソールの画面で起動ユーザを設定します。 [ワークユニット] > [新規作成]タブ > [詳細設定] > [ワークユニット設定] > [ワークユニット自動起動] (注1)
container.gid	IJServerワークユニットを起動したユーザのプライマリグループになります。

注1)

[ワークユニット] > [新規作成]タブ > [詳細設定]による操作は、[ワークユニット] > “ワークユニット名” > [環境設定]タブ > [詳細設定]によっても操作できます。

注2)

以下のパラメタは指定する必要はありません。

- -Dcom.fujitsu.interstage.jservlet.j2ee=true
- -Dorg.omg.CORBA.ORBClass=com.fujitsu.ObjectDirector.CORBA.ORB
- -Dorg.omg.CORBA.ORBSingletonClass=com.fujitsu.ObjectDirector.CORBA.SingletonORB
- -Djavax.rmi.CORBA.StubClass=com.fujitsu.ObjectDirector.rmi.CORBA.StubDelegateImpl
- -Djavax.rmi.CORBA.UtilClass=com.fujitsu.ObjectDirector.rmi.CORBA.UtilDelegateImpl
- -
- Djavax.rmi.CORBA.PortableRemoteObjectClass=com.fujitsu.ObjectDirector.rmi.CORBA.PortableRemoteObjectDelegateImpl

注3)

Servletコンテナ(IJServer)自身の実行に必要なパスは自動的に設定されるため、V5.1以前のServletサービスで環境変数PATH、CLASSPATH、LD_LIBRARY_PATHに設定していた以下の内容は指定する必要はありません。

Windows

- JDKのインストールディレクトリ¥lib¥tools.jar
- C:¥Interstage¥F3FMjs2¥classes¥jsboot.jar
- C:¥Interstage¥F3FMjs2¥classes¥servlet.jar
- C:¥Interstage¥J2EE¥bin
- C:¥Interstage¥ODWIN¥etc¥class¥ODjava2.jar …JDK1.3の場合
- C:¥Interstage¥ODWIN¥etc¥class¥ODjava4.jar …JDK1.4の場合
- C:¥Interstage¥jms¥bin
- C:¥Interstage¥eswin¥lib¥esnotifyjava2.jar …JDK1.3の場合
- C:¥Interstage¥eswin¥lib¥esnotifyjava4.jar …JDK1.4の場合
- C:¥Interstage¥jms¥lib¥fjmsprovider.jar
- SunのホームページからダウンロードしたFile System Service Provider(providerutil.jarおよびfscontext.jar)
- C:¥Interstage¥bin

Solaris

- JDKのインストールディレクトリ/lib/tools.jar
- /opt/FJSVjs2/classes/jsboot.jar
- /opt/FJSVjs2/classes/servlet.jar
- /opt/FJSVj2ee/bin
- /opt/FSUNod/etc/class/ODjava2.jar …JDK1.3の場合
- /opt/FSUNod/etc/class/ODjava4.jar …JDK1.4の場合
- /opt/FSUNod/lib
- /opt/FJSVjms/bin
- /opt/FJSVes/lib/esnotifyjava2.jar …JDK1.3の場合
- /opt/FJSVes/lib/esnotifyjava4.jar …JDK1.4の場合
- /opt/FJSVjms/lib/fjmsprovider.jar
- /opt/FJSVjms/lib
- SunのホームページからダウンロードしたFile System Service Provider(providerutil.jarおよびfscontext.jar)

Linux

- JDKのインストールディレクトリ/lib/tools.jar
- /opt/FJSVjs2/classes/jsboot.jar
- /opt/FJSVjs2/classes/servlet.jar
- /opt/FJSVj2ee/bin
- /opt/FJSVod/etc/class/ODjava2.jar …JDK1.3の場合
- /opt/FJSVod/etc/class/ODjava4.jar …JDK1.4の場合
- /opt/FJSVod/lib
- /opt/FJSVjms/bin
- /opt/FJSVes/lib/esnotifyjava2.jar …JDK1.3の場合
- /opt/FJSVes/lib/esnotifyjava4.jar …JDK1.4の場合
- /opt/FJSVjms/lib/fjmsprovider.jar
- /opt/FJSVjms/lib
- SunのホームページからダウンロードしたFile System Service Provider(providerutil.jarおよびfscontext.jar)

アプリケーションで必要なクラスについては、“J2EE ユーザーズガイド”の“JServerで使用するクラスの設定について”を参照してください。

注4)

WebサーバコネクタとServletコンテナが別システム([システム] > [環境設定]タブ > [Servletサービスの詳細設定] > [Webサーバとワークユニットを同一のマシンで運用する]で[運用しない]を選択)の場合に設定が必要であり、同一システムの場合は必要ありません。

A.2 サブレット・ゲートウェイ環境定義の移行

以下にV5.1以前のServletサービスにおけるサブレット・ゲートウェイ環境定義ファイルの定義項目と、Interstage管理コンソールの対応表を示します。なお、isj2eeadminコマンドを使用して、以下の各設定を行うこともできます。詳細は、“リファレンスマニュアル(コマンド編)”を参照してください。

A.2.1 移行元の環境で使用していたWebサーバがInterstage HTTP Serverの場合

サブレット・ゲートウェイ	Interstage管理コンソール
ApJServDefaultPort	ありません。
ApJServMount	(注2) [Webサーバ] > “Webサーバ名” > [Webサーバコネクタ] > [新規作成]タブ > [ServletコンテナのIPアドレス:ポート番号]および[Webアプリケーション名] (注1)
ApJServLogFile	[Webサーバ] > “Webサーバ名” > [Webサーバコネクタ] > [ログ設定]タブ > [ログ出力ディレクトリ] [Webサーバ] > “Webサーバ名” > [Webサーバコネクタ] > [ログ参照]タブから参照できます。
ApJServLogLevel	[Webサーバ] > “Webサーバ名” > [Webサーバコネクタ] > [ログ設定]タブ > [デバッグ情報]
ApJServRetryAttempts	ありません。
ApJServBalance	複数のServletコンテナを使用する場合は、“■1つのWebアプリケーションを複数のServletコンテナで対応する”または“■複数のWebアプリケーションを複数のServletコンテナで対応する”を参照してください。
ApJServHost	
ApJServDefinitionRead	常に有効です。
ApJServSessionRecovery	ありません。

注1)

[Webサーバ] > “Webサーバ名” > [Webサーバコネクタ] > [新規作成]タブによる操作は、[Webサーバ] > “Webサーバ名” > [Webサーバコネクタ] > “ワークユニット名”によっても操作できます。

注2)

WebサーバコネクタとServletコンテナが別システム([システム] > [環境設定]タブ > [Servletサービス詳細設定] > [Webサーバとワークユニットを同一のマシンで運用する]で[運用しない]を選択)の場合に設定が必要であり、同一システムの場合は必要ありません。

■1つのWebアプリケーションを複数のServletコンテナで対応する

Webアプリケーションを配備したIJSERVERワークユニットの以下の値を変更することで、複数のServletコンテナでWebアプリケーションを実行できるようになります。これにより、1つのServletコンテナに通信が集中することなく、プロセス多重度で指定した多重度数のServletコンテナに負荷を分散させることができます。

[ワークユニット] > [新規作成]タブ > [詳細設定] > [ワークユニット設定] > [プロセス多重度]

または

[ワークユニット] > “ワークユニット名” > [環境設定]タブ > [ワークユニット設定] > [プロセス多重度]

■複数のWebアプリケーションを複数のServletコンテナで対応する

複数のIIServerワークユニットを作成し、Webアプリケーションを業務単位に異なるIIServerワークユニットに配備します。

IIServerワークユニットは、IIServerワークユニット単位に起動、停止できるため、業務運用中であっても他業務へ影響を与えることなく、該当する業務のWebアプリケーションの変更・保守(復旧)作業を行うことができます。

■WebサーバコネクタとServletコンテナが別システムの定義例

WebサーバコネクタとServletコンテナが別システムの定義例は、“J2EE ユーザーズガイド”の“Servletサービスの運用準備”－“IIServerとWebサーバを分離して運用する場合の手順”を参照してください。

A.2.2 移行元の環境で使用していたWebサーバがInterstage HTTP Server以外の場合

サーブレット・ゲートウェイ	Interstage管理コンソール
DefaultPort	ありません。
Timeout	(注2) [Webサーバ] > “Webサーバ名” > [Webサーバコネクタ] > [新規作成]タブ > [詳細設定] > [送受信タイムアウト] (注1)
LogFile	[Webサーバ] > “Webサーバ名” > [Webサーバコネクタ] > [ログ設定]タブ > [ログ出力ディレクトリ] [Webサーバ] > “Webサーバ名” > [Webサーバコネクタ] > [ログ参照]タブから参照できます。
Mount	(注2) [Webサーバ] > “Webサーバ名” > [Webサーバコネクタ] > [新規作成]タブ > [Webアプリケーション名] (注1)
Container	(注2) [Webサーバ] > “Webサーバ名” > [Webサーバコネクタ] > [新規作成]タブ > [ServletコンテナのIPアドレス:ポート番号] (注1) 複数のServletコンテナを使用する場合は、“■1つのWebアプリケーションを複数のServletコンテナで対応する”または“■複数のWebアプリケーションを複数のServletコンテナで対応する”を参照してください。
DefaultHost	ありません。
ErrorPage	ありません。 (注3)
MaxConnection	(注2) [Webサーバ] > “Webサーバ名” > [Webサーバコネクタ] > [新規作成]タブ > [詳細設定] > [Servletコンテナへの最大接続数] (注1)
DefinitionRead	常に有効です。
SessionRecovery	ありません。

注1)

[Webサーバ] > “Webサーバ名” > [Webサーバコネクタ] > [新規作成]タブによる操作は、[Webサーバ] > “Webサーバ名” > [Webサーバコネクタ] > “ワークユニット名”によっても操作できます。

注2)

WebサーバコネクタとServletコンテナが別システム([システム] > [環境設定]タブ > [Servletサービス詳細設定] > [Webサーバとワークユニットを同一のマシンで運用する]で[運用しない]を選択)の場合に設定が必要であり、同一システムの場合は必要ありません。

注3)

本バージョン・レベルでは、Webサーバコネクタ(サーブレット・ゲートウェイ)のErrorPageをWebサーバのエラーページ機能を使って設定します。

- Interstage HTTP Serverの場合
Webサーバの環境定義ファイル(httpd.conf)のErrorDocumentディレクティブで設定します。詳細については、“Interstage HTTP Server 運用ガイド”―“ディレクティブ一覧”―“ErrorDocument”を参照してください。
- Internet Information Servicesの場合
インターネット サービス マネージャを使用して、ISAPIフィルタ・ISAPIエクステンションを設定した、Webサイトおよび仮想ディレクトリのカスタムエラーの設定をします。詳細については、Internet Information Servicesのインターネット サービス マネージャのマニュアルを参照してください。
注) Internet Information Services 5.0では、エラーページの設定は行えません。
- Sun Java System Web Serverの場合
obj.conf構成ファイルのErrorディレクティブで設定します。詳細については、Sun Java System Web Serverのマニュアルを参照してください。

運用パターンの定義例は、“■1つのWebアプリケーションを複数のServletコンテナで対応する”および“■複数のWebアプリケーションを複数のServletコンテナで対応する”を参照してください。

■WebサーバコネクタとServletコンテナが別システムの定義例

WebサーバコネクタとServletコンテナが別システムの定義例は、“J2EE ユーザーズガイド”の“Servletサービスの運用準備”―“IJSERVERとWebサーバを分離して運用する場合の手順”を参照してください。

■Sun Java System Web Server(iPlanet Web Server)の構成ファイル

本バージョン・レベルでは、Sun Java System Web Serverのmagnus.confとobj.confの記述形式が、Interstageの旧バージョンと異なります。本バージョン・レベルにおけるobj.confとmagnus.confの記述形式については、“J2EEユーザーズガイド”の“Servletサービスの運用準備”―“Sun Java System Web ServerとInterstageの連携”を参照してください。

また、本バージョン・レベルでは、Sun Java System Web Server 4.xとの連携をサポートしていません。InterstageとSun Java System Web Serverを連携させる場合は、Sun Java System Web Serverの6.xを使用してください。

A.3 サーブレット・コンテナ環境定義の移行

以下にV5.1以前のServletサービスにおけるサーブレット・コンテナ環境定義ファイルの定義項目と、Interstage管理コンソールの対応表を示します。なお、isj2eadminコマンドを使用して、以下の各設定を行うこともできます。詳細は、“リファレンスマニュアル(コマンド編)”を参照してください。

—:該当項目なし

V5.1以前のServletサービス		Interstage管理コンソール
タグ	属性	
Server	—	必要ありません。
Logger	name	必要ありません。 [ワークユニット] > “ワークユニット名” > [ログ参照]タブ > [コンテナログ]で参照できます。
	path	[ワークユニット] > [新規作成]タブ > [詳細設定] > [ワークユニット設定] > [ログ出力ディレクトリ] (注1) [ワークユニット] > “ワークユニット名” > [ログ参照]タブ > [コンテナログ]から参照できます。
	backup	[ワークユニット] > “ワークユニット名” > [ログ定義]タブ > [ログファイルのロールオーバー]
	size	
	timestamp	必要ありません。

V5.1以前のServletサービス		Interstage管理コンソール
タグ	属性	
ContextManager	workDir	変更できません。以下のディレクトリになります。 Windows J2EE共通ディレクトリ¥ijserver¥IJSERVERワークユニット名¥work Solaris Linux J2EE共通ディレクトリ/ijserver/IJSERVERワークユニット名/work (注3)
	clientSession	指定できません。
	version	指定できません。
	errResponse	指定できません。
	sessionRecovery	[ワークユニット] > [新規作成]タブ > [詳細設定] > [セッションリカバリ設定] (注1) あわせて、“J2EEユーザーズガイド”の“セッションリカバリ機能”を参照してください。
	name	ありません。
	sessionIdPrefix	ありません。
ContextInterceptor	classname	必要ありません。
RequestInterceptor	classname	必要ありません。
Connector	classname	必要ありません。
Parameter	handler	必要ありません。
	port	[ワークユニット] > [新規作成]タブ > [詳細設定] > [Servletコンテナ設定] > [ポート番号] (注1)
	ipaddress	(注4) [ワークユニット] > [新規作成]タブ > [詳細設定] > [Servletコンテナ設定] > [ServletコンテナのIPアドレス] (注1)
	acsaddress	(注4) [ワークユニット] > [新規作成]タブ > [詳細設定] > [Webサーバコネクタ(コネクタ)設定] > [要求を受付けるWebサーバのIPアドレス] (注1)
	min_spare_threads	[ワークユニット] > [新規作成]タブ > [詳細設定] > [Servletコンテナ設定] > [同時処理数]の“初期値” (注1)
	max_spare_threads	[ワークユニット] > [新規作成]タブ > [詳細設定] > [Servletコンテナ設定] > [同時処理数]の“待機中の最大値” (注1)
	max_threads	[ワークユニット] > [新規作成]タブ > [詳細設定] > [Servletコンテナ設定] > [同時処理数]の“最大値” (注1)
	connection_timeout	[ワークユニット] > [新規作成]タブ > [詳細設定] > [Servletコンテナ設定] > [タイムアウト] (注1)
Context	path	必要ありません。
	docBase	[ワークユニット] > “ワークユニット名” > [配備]タブ > [配備設定]で、[サーバ上の任意の位置で実行するWebアプリケーションを配備する]を選択して配備。 上記以外で配備の場合は以下のディレクトリになります。 Windows J2EE共通ディレクトリ¥ijserver¥IJSERVERワークユニット名¥webapps¥Webアプリケーション名

V5.1以前のServletサービス		Interstage管理コンソール
タグ	属性	
		Solaris Linux J2EE共通ディレクトリ/ijserver/IJServerワークユニット名/webapps/Webアプリケーション名 (注3)
	reloadable	クラスのオートリロード: [ワークユニット] > [新規作成]タブ > [詳細設定] > [共通定義] > [オートリロード] (注1) JSPのリロード: [ワークユニット] > [新規作成]タブ > [詳細設定] > [Servletコンテナ設定] > [JSPのリロード] (注1)
	inputcode	[ワークユニット] > “ワークユニット名” > [配備]タブ > [Webアプリケーション設定] > [エンコーディング] (注2) [ワークユニット] > [新規作成]タブ > [詳細設定] > [Servletコンテナ設定] > [リクエストボディ処理のエンコーディングをクエリパラメタに使用] (注1)
	jsAuthentication	[ワークユニット] > “ワークユニット名” > [配備]タブ > [Webアプリケーション設定] > [認証] (注2)
	dirList	[ワークユニット] > [新規作成]タブ > [詳細設定] > [Servletコンテナ設定] > [ファイルの一覧表示] (注1)
	tagPooling	[ワークユニット] > [新規作成]タブ > [詳細設定] > [Servletコンテナ設定] > [カスタムタグプーリングの使用] (注1)
	clientSession	[ワークユニット] > “ワークユニット名” > [配備]タブ > [Webアプリケーション設定] > [セッション] (注2)
	default	[クッキーに設定する]を選択
	disable	[クッキーに設定しない]を選択
	permanent	[Webブラウザでセッションを保存する]をチェック ([クッキーに設定する]を選択要)
	urlEncoding	必要ありません。
APITracer	—	“トラブルシューティング集”の“Javaツール機能” – “メソッドトレース機能”を参照してください。
TagPooling	—	必要ありません。

注1)

[ワークユニット] > [新規作成]タブ > [詳細設定]による操作は、[ワークユニット] > “ワークユニット名” > [環境設定]タブ > [詳細設定]によっても操作できます。

注2)

[ワークユニット] > “ワークユニット名” > [配備]タブによる操作は、[ワークユニット] > “ワークユニット名” > “Webアプリケーションのモジュール名” > [環境設定]タブ >によっても操作できます。

注3)

J2EE共通ディレクトリのデフォルトは、Windows(R)ではC:\Interstage¥J2EE¥var¥deployment、Solaris/Linuxでは/opt/FJSVj2ee/var/deploymentです。

注4)

WebサーバコネクタとServletコンテナが別システム([システム] > [環境設定]タブ > [Servletサービスの詳細設定] > [Webサーバとワークユニットを同一のマシンで運用する]で[運用しない]を選択)の場合に設定が必要であり、同一システムの場合は必要ありません。

A.4 JServletプロパティファイルの移行

以下にV5.1以前のServletサービスにおけるJServletプロパティファイルの定義項目と、Interstage管理コンソールの対応表を示します。
なお、isj2eeadminコマンドを使用して、以下の各設定を行うこともできます。詳細は、“リファレンスマニュアル(コマンド編)”を参照してください。

V5.1以前のServletサービス	Interstage管理コンソール
com.fujitsu.interstage.jservle t.sessionrecovery.mode	ありません。
com.fujitsu.interstage.jservle t.session.cookie.secure.mode	[ワークユニット] > “ワークユニット名” > [配備]タブ > [Webアプリケーション設定] > [セッション] > [クッキーにSecure属性を常に付加する] または、 [ワークユニット] > “ワークユニット名” > “Webアプリケーションのモジュール名” > [環境設定]タブ > [セッション] > [クッキーにSecure属性を常に付加する] V5.1以前のServletサービスで“auto”を指定していた場合は指定する必要はありません。
com.fujitsu.interstage.jservle t.dbcs	必要ありません。

索引

[E]

Entity Bean内で発生したシステム例外処理.....130

[G]

getRollbackOnly().....131

[R]

remove(PrimaryKey).....131

[S]

Session Bean内で発生したシステム例外処理.....130

Session BeanのMandatoryトランザクション属性.....131

Session BeanのPrimaryKey関連メソッド.....130

setRollbackOnly().....131

STATEFUL Session Bean.....131

[U]

UserTransactionのlookup.....131

[か]

コンテナのリソース接続者管理.....131

[た]

単一参照.....59

[ら]

リエントラント制御.....130