

uCosminexus DocumentBroker Version 3  
クラスライブラリ Java リファレンス

文法書

3000-3-F17-20

## 対象製品

R-1M95D-43 uCosminexus DocumentBroker Development Kit Version 3 03-60 (適用 OS : AIX 5L V5.1 , AIX 5L V5.2 , AIX 5L V5.3)

R-1595D-43 uCosminexus DocumentBroker Development Kit Version 3 03-70 (適用 OS : Windows Server 2003 , Windows Server 2003 R2 , Windows Server 2003 R2 x64 Edition , Windows Server 2008 x86 , Windows Server 2008 R2 , Windows Server 2012 , Windows XP , Windows Vista , Windows 7 , Windows 8)

これらのプログラムプロダクトのほかにもこのマニュアルをご利用になれる場合があります。詳細は「リリースノート」でご確認ください。

印の製品については、サポート時期をご確認ください。

## 輸出時の注意

本製品を輸出される場合には、外国為替及び外国貿易法の規制並びに米国輸出管理規則など外国の輸出関連法規をご確認の上、必要な手続きをお取りください。

なお、不明な場合は、弊社担当営業にお問い合わせください。

## 商標類

Active Directory は、米国 Microsoft Corporation の、米国およびその他の国における登録商標または商標です。

AIX は、米国およびその他の国における International Business Machines Corporation の商標です。

CORBA は、Object Management Group が提唱する分散処理環境アーキテクチャの名称です。

GIF は、米国 CompuServe Inc. が開発したフォーマットの名称です。

Microsoft は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

Microsoft Office Word は、米国 Microsoft Corporation の商品名称です。

Microsoft は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

Microsoft Office Word は、米国 Microsoft Corporation の商品名称です。

OLE は、米国 Microsoft Corporation が開発したソフトウェア名称です。

Oracle と Java は、Oracle Corporation 及びその子会社、関連会社の米国及びその他の国における登録商標です。

TIFF は、米国 Aldus Corp. が開発したフォーマットの名称です。

UNIX は、The Open Group の米国ならびに他の国における登録商標です。

Windows は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

Windows Server は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

Windows Vista は、米国 Microsoft Corporation の米国およびその他の国における登録商標または商標です。

活文、PDFstaff は、株式会社日立ソリューションズの登録商標です。

その他記載の会社名、製品名は、それぞれの会社の商標もしくは登録商標です。

## 発行

2013 年 9 月 3000-3-F17-20

## 著作権

All Rights Reserved. Copyright (C) 2006, Hitachi, Ltd.

All Rights Reserved. Copyright (C) 2006, 2013, Hitachi Solutions, Ltd.

## 変更内容

変更内容 ( 3000-3-F17-20 ) DocumentBroker Development Kit Version 3 03-70 ( 適用 OS : Windows Server 2003 , Windows Server 2003 R2 , Windows Server 2003 R2 x64 Edition , Windows Server 2008 x86 , Windows Server 2008 x64 , Windows Server 2008 R2 , Windows Server 2012 , Windows XP , Windows Vista , Windows 7 , Windows 8 )

追加・変更内容	変更箇所
次の前提 OS をサポートしました。 <ul style="list-style-type: none"><li>• Windows Server 2012</li><li>• Windows 8</li></ul>	-

単なる誤字・脱字などはお断りなく訂正しました。



# はじめに

---

このマニュアルは、次に示すプログラムプロダクトで提供する Java クラスライブラリの詳細、インターフェースの詳細、メソッドの文法、およびメッセージの詳細についてリファレンス形式で説明したものです。

- R-1M95D-43 uCosminexus DocumentBroker Development Kit Version 3
- R-1595D-43 uCosminexus DocumentBroker Development Kit Version 3

## 対象読者

このマニュアルは、uCosminexus DocumentBroker Development Kit が提供する Java クラスライブラリを使用して、ユーザアプリケーションプログラムを開発し、システムを構築、運用および管理する方を対象にしています。なお、次の内容を理解されていることを前提としています。

- UNIX または Windows に関する知識
- Java™ 言語に関する知識
- XML に関する知識
- SQL 言語に関する知識

## マニュアルの構成

このマニュアルは、次に示す章と付録から構成されています。

### 第 1 章 Java クラスライブラリのクラス、インターフェース、およびメソッド

Java クラスライブラリのクラス、インターフェース、およびメソッドを一覧表形式で説明しています。また、Java クラスライブラリで提供するクラスライブラリのリファレンスマニュアルのクラス、インターフェース、およびメソッドの記述形式（ページレイアウト）について説明しています。

### 第 2 章 ファクトリクラス詳細

ファクトリクラスのクラス、インターフェース、およびメソッドについて説明しています。

### 第 3 章 パラメタクラス詳細

パラメタクラスのインターフェース、およびメソッドについて説明しています。

### 第 4 章 文書管理クラス詳細

文書管理クラスのインターフェース、およびメソッドについて説明しています。

### 第 5 章 メタクラス詳細

メタクラスのインターフェース、およびメソッドについて説明しています。

### 第 6 章 例外クラス詳細

例外クラスの詳細について説明しています。

### 第 7 章 定数定義クラス詳細

定数定義クラスの定義内容についてカテゴリごとに説明しています。

### 第 8 章 ライブラリ情報取得クラス詳細

ライブラリ情報取得クラスのクラスとそのメソッドについて説明しています。

### 第 9 章 トレースクラス詳細

トレースクラスのクラスとそのメソッドについて説明しています。

### 第 10 章 メッセージの一覧

Java クラスライブラリが出力するメッセージについて説明しています。

付録 A 文書空間で使用する文字コード種別が UTF-8 の場合の機能差異

文書空間で使用する文字コード種別が UTF-8 の場合のクラス、インターフェースおよびメソッドの機能差異について説明しています。

付録 B 用語解説

Java クラスライブラリで使用する用語について説明しています。

## 関連マニュアル

このマニュアルは次のマニュアルと関連がありますので、必要に応じてお読みください。なお、本文に記載のマニュアル名称は、「uCosminexus DocumentBroker」を「DocumentBroker」と表記しています。

uCosminexus DocumentBroker Version 3 クラスライブラリ Java 解説 (3000-3-F16)

Java クラスライブラリの機能および環境設定のしかたについて知りたい場合に参照してください。

uCosminexus DocumentBroker Version 3 クラスライブラリ Java サンプル Web アプリケーション (3000-3-F18)

uCosminexus DocumentBroker Development Kit および uCosminexus DocumentBroker Runtime が提供している Java クラスライブラリを使用したサンプル Web アプリケーションの機能と使用方法、およびサンプル Web アプリケーションを参考にした Web アプリケーションの開発方法について知りたい場合に参照してください。

DocumentBroker Version 3 システム導入・運用ガイド (3000-3-D01)

uCosminexus DocumentBroker Version 3 システム導入・運用ガイド (3020-3-U71)

uCosminexus DocumentBroker を使用する環境を定義、管理および運用する場合に参照してください。

UNIX の場合は、資料番号が 3000-3-D01 のマニュアルを参照してください。

Windows の場合は、資料番号が 3020-3-U71 のマニュアルを参照してください。

uCosminexus DocumentBroker Version 3 メッセージ (3000-3-F12)

uCosminexus DocumentBroker が出力するメッセージについて知りたい場合に参照してください。

## 読書手順

このマニュアルは、利用目的に合わせて章を選択してお読みいただけます。次に示す表を参考にして、お読みになる章を選択してください。

このマニュアルを読む目的	記述箇所
どのような Java クラスライブラリのクラス、インターフェースおよびメソッドがあるかを知りたい	1 章
クラス、インターフェースおよびメソッドの説明形式について知りたい	1.9 節
ファクトリクラスのインターフェースおよびメソッドについて知りたい	2 章
パラメタクラスのインターフェースおよびメソッドについて知りたい	3 章
文書管理クラスのインターフェースおよびメソッドについて知りたい	4 章
メタクラスのインターフェースおよびメソッドについて知りたい	5 章
例外クラスについて知りたい	6 章
定数定義クラスの定義内容について知りたい	7 章
ライブラリ情報取得クラスのクラスおよびメソッドについて知りたい	8 章
トレースクラスのクラスおよびメソッドについて知りたい	9 章
出力されるメッセージについて知りたい	10 章

このマニュアルを読む目的	記述箇所
文書空間で使用する文字コード種別が UTF-8 の場合のクラス、インターフェースおよびメソッドの機能差異について知りたい	付録 A
uCosminexus DocumentBroker の用語を知りたい	付録 B

## このマニュアルでの表記

このマニュアルでは、製品名称を次に示す略称で表記しています。

製品名称	略称
AIX 5L V5.1	AIX
AIX 5L V5.2	
AIX 5L V5.3	
uCosminexus DocumentBroker Development Kit Version 3	DocumentBroker
uCosminexus DocumentBroker Runtime Version 3	
uCosminexus DocumentBroker Server Version 3	
uCosminexus DocumentBroker Development Kit Version 3	DocumentBroker Development Kit
uCosminexus DocumentBroker Runtime Version 3	DocumentBroker Runtime
uCosminexus DocumentBroker Server Version 3	DocumentBroker サーバ
Microsoft(R) Active Directory(R)	Active Directory
Microsoft(R) Office Word	Word
Microsoft(R) Word	
Microsoft(R) Windows Server(R) 2003, Enterprise Edition 日本語版	Windows Server 2003
Microsoft(R) Windows Server(R) 2003, Standard Edition 日本語版	
Microsoft(R) Windows Server(R) 2003 R2, Enterprise Edition 日本語版	Windows Server 2003 R2
Microsoft(R) Windows Server(R) 2003 R2, Standard Edition 日本語版	
Microsoft(R) Windows Server(R) 2003 R2, Enterprise x64 Edition 日本語版	
Microsoft(R) Windows Server(R) 2003 R2, Standard x64 Edition 日本語版	Windows Server 2003 R2 または Windows Server 2003 R2 x64 Edition
Microsoft(R) Windows Server(R) 2008 Enterprise 32-bit 日本語版	Windows Server 2008 または Windows Server 2008 x86
Microsoft(R) Windows Server(R) 2008 Standard 32-bit 日本語版	
Microsoft(R) Windows Server(R) 2008 R2 Enterprise 日本語版	Windows Server 2008 R2
Microsoft(R) Windows Server(R) 2008 R2 Standard 日本語版	
Microsoft(R) Windows Server(R) 2012 Standard 日本語版	Windows Server 2012
Microsoft(R) Windows Server(R) 2012 Datacenter 日本語版	
Microsoft(R) Windows(R) XP Professional Operating System	Windows XP
Microsoft(R) Windows Vista(R) Ultimate 日本語版	Windows Vista
Microsoft(R) Windows Vista(R) Business 日本語版	
Microsoft(R) Windows Vista(R) Enterprise 日本語版	
Microsoft(R) Windows(R) 7 Professional 日本語版 (32 ビット版)	Windows 7
Microsoft(R) Windows(R) 7 Enterprise 日本語版 (32 ビット版)	
Microsoft(R) Windows(R) 7 Ultimate 日本語版 (32 ビット版)	

製品名称	略称
Microsoft(R) Windows(R) 7 Professional 日本語版 (64 ビット版)	
Microsoft(R) Windows(R) 7 Enterprise 日本語版 (64 ビット版)	
Microsoft(R) Windows(R) 7 Ultimate 日本語版 (64 ビット版)	
Windows(R) 8 Pro 日本語版 (32 ビット版)	Windows 8
Windows(R) 8 Pro 日本語版 (64 ビット版)	
Windows(R) 8 Enterprise 日本語版 (32 ビット版)	
Windows(R) 8 Enterprise 日本語版 (64 ビット版)	
Sun Java(TM) System Directory Server 5.1	Sun Java System Directory Server
Sun Java(TM) System Directory Server 5.2	
Sun Java(TM) System Directory Server 6.3	

このほか、このマニュアルでは、次に示す表記方法を使用しています。

- Java™ を Java と表記します。
- uCosminexus DocumentBroker Development Kit および uCosminexus DocumentBroker Runtime が提供する Java 言語対応のクラスライブラリを Java クラスライブラリと表記します。
- AIX を UNIX と表記することがあります。
- Windows Server 2003, Windows Server 2003 R2, Windows Server 2003 R2 x64 Edition, Windows Server 2008 x86, Windows Server 2008 R2, Windows Server 2012, Windows XP, Windows Vista, Windows 7, および Windows 8 を合わせて Windows と表記することがあります。

## uCosminexus DocumentBroker のマニュアルで使用する略語

uCosminexus DocumentBroker のマニュアルで使用する英略語を次に示します。

英略語	英字での表記
ACE	<u>A</u> ccess <u>C</u> ontrol <u>E</u> lement
ACFlag	<u>A</u> ccess <u>C</u> ontrol <u>F</u> lag
ACL	<u>A</u> ccess <u>C</u> ontrol <u>L</u> ist
AIIM	<u>A</u> ssociation for <u>I</u> nformation and <u>I</u> mage <u>M</u> anagement <u>I</u> nternational
API	<u>A</u> pplication <u>P</u> rogramming <u>I</u> nterface
ASCII	<u>A</u> merican <u>S</u> tandard <u>C</u> ode for <u>I</u> nformation <u>I</u> nterchange
BES	<u>B</u> ack <u>E</u> nd <u>S</u> erver
BLOB	<u>B</u> inary <u>L</u> arge <u>O</u> bject
BMP	<u>B</u> it <u>M</u> ap
BNF	<u>B</u> ackus <u>N</u> ormal <u>F</u> orm
BOA	<u>B</u> asic <u>O</u> bject <u>A</u> dapter
CD-ROM	<u>C</u> ompact <u>D</u> isc <u>R</u> ead <u>O</u> nly <u>M</u> emory
CGI	<u>C</u> ommon <u>G</u> ateway <u>I</u> nterface
CORBA	<u>C</u> ommon <u>O</u> bject <u>R</u> equest <u>B</u> roker <u>A</u> rchitecture
CPU	<u>C</u> entral <u>P</u> rocessing <u>U</u> nit
CR	<u>C</u> arriage <u>R</u> eturn



英略語	英字での表記
CSV	<u>C</u> omma <u>S</u> eparated <u>V</u> alue
DAP	<u>D</u> irectory <u>A</u> ccess <u>P</u> rotocol
DAT	<u>D</u> igital <u>A</u> udio <u>T</u> ape
DB	<u>D</u> atab <u>a</u> se
DBMS	<u>D</u> atab <u>a</u> se <u>M</u> anagement <u>S</u> ystem
DCD	<u>D</u> ocument <u>C</u> ontent <u>D</u> escription
DDE	<u>D</u> ynamic <u>D</u> ata <u>E</u> xchange
DIT	<u>D</u> irectory <u>I</u> nformation <u>T</u> ree
DLL	<u>D</u> ynamic <u>L</u> inking <u>L</u> ibrary
DMA	<u>D</u> ocument <u>M</u> anagement <u>A</u> lliance
DN	<u>D</u> istinguished <u>N</u> ame
DTD	<u>D</u> ocument <u>T</u> ype <u>D</u> efinition
EOF	<u>E</u> nd of <u>F</u> ile
ESIS-B	<u>E</u> lement <u>S</u> tructure <u>I</u> nformation <u>S</u> et- <u>B</u> inary Format
EUC	<u>E</u> xtended <u>U</u> NIX <u>C</u> ode
FAM	<u>F</u> ile <u>A</u> ccess <u>M</u> odule
GIF	<u>G</u> raphics <u>I</u> nterchange <u>F</u> ormat
GUI	<u>G</u> raphical <u>U</u> ser <u>I</u> nterface
GUID	<u>G</u> lobally <u>U</u> nique <u>I</u> dentifier
HTML	<u>H</u> ypertext <u>M</u> arkup <u>L</u> anguage
HTTP	<u>H</u> ypertext <u>T</u> ransfer <u>P</u> rotocol
IANA	<u>I</u> nternet <u>A</u> ssigned <u>N</u> umbers <u>A</u> uthority
ID	<u>I</u> dentifier
IPF	<u>I</u> tanium(R) <u>P</u> rocessor <u>F</u> amily
ISO	<u>I</u> nternational <u>O</u> rganization for <u>S</u> tandardization
JIS	<u>J</u> apanese <u>I</u> ndustrial <u>S</u> tandards
JPEG	<u>J</u> oint <u>P</u> hotographic <u>E</u> xpert <u>G</u> roup
LAN	<u>L</u> ocal <u>A</u> rea <u>N</u> etwork
LDAP	<u>L</u> ightweight <u>D</u> irectory <u>A</u> ccess <u>P</u> rotocol
LF	<u>L</u> ine <u>F</u> eed
MFC	<u>M</u> icrosoft <u>F</u> oundation <u>C</u> lass
MIME	<u>M</u> ultipurpose <u>I</u> nternet <u>M</u> ail <u>E</u> xtensions
OCR	<u>O</u> ptical <u>C</u> haracter <u>R</u> eaders
OIID	<u>O</u> bject <u>I</u> nstance <u>I</u> dentifier
OLE	<u>O</u> bject <u>L</u> inking and <u>E</u> MBEDDING
OMG	<u>O</u> bject <u>M</u> anagement <u>G</u> roup
ORB	<u>O</u> bject <u>R</u> equest <u>B</u> roker
ORDB	<u>O</u> bject <u>R</u> elational <u>D</u> atabase

英略語	英字での表記
OS	<u>O</u> perating <u>S</u> ystem
OTS	<u>O</u> bject <u>T</u> ransaction <u>S</u> ervice
PC	<u>P</u> ersonal <u>C</u> omputer
PDF	<u>P</u> ortable <u>D</u> ocument <u>F</u> ormat
RDB	<u>R</u> elational <u>D</u> atabase
RDN	<u>R</u> elative <u>D</u> istinguished <u>N</u> ame
RFC	<u>R</u> equest for <u>C</u> omment
RTF	<u>R</u> ich <u>T</u> ext <u>F</u> ormat
SQL	<u>S</u> tructured <u>Q</u> uery <u>L</u> anguage
TCP/IP	<u>T</u> ransmission <u>C</u> ontrol <u>P</u> rotocol/ <u>I</u> nternet <u>P</u> rotocol
TIFF	<u>T</u> ag Image <u>F</u> ile <u>F</u> ormat
UOC	<u>U</u> ser <u>O</u> wn <u>C</u> oding
URL	<u>U</u> niform <u>R</u> esource <u>L</u> ocator
UTC	<u>U</u> niversal <u>T</u> ime <u>C</u> oordinated
UTF-8	<u>8</u> -bit <u>U</u> CS <u>T</u> ransformation <u>F</u> ormat
W3C	<u>W</u> orld <u>W</u> ide <u>W</u> eb <u>C</u> onsortium
WWW	<u>W</u> orld <u>W</u> ide <u>W</u> eb
XML	<u>E</u> xtensible <u>M</u> arkup <u>L</u> anguage

## このマニュアルで使用する記号

このマニュアルで使用する記号を次に示します。

記号	意味
	横に並べられた複数の項目に対する項目間の区切りを示し、「または」の意味を表します。 (例) A   B A または B を指定することを示します。
_	括弧で囲まれた複数項目のうち 1 項目に対し使用され、括弧内のすべてを省略した場合にシステムが取る標準値を示します。 (例) [ <u>A</u>   B ] 何も指定しない場合は A が仮定されます。
{ }	この記号で囲まれている複数の項目のうちから一つを選択することを示します。項目が横に並べられ、記号   で区切られている場合は、そのうちの一つを選択します。 (例) { A   B   C } A, B または C のどれかを指定することを示します。
[ ]	この記号で囲まれている項目は省略してもよいことを意味します。複数の項目が横に並べて記述されている場合には、すべてを省略するか、記号 { } と同じくどれか一つを選択します。 (例 1) [ A ] 「何も指定しない」か「A を指定する」ことを示します。 (例 2) [ B   C ] 「何も指定しない」か「B または C を指定する」ことを示します。
< >	この記号で囲まれている項目は、該当する要素を指定することを示します。 (例) < プロパティ > プロパティを記述します。

記号	意味
: : =	この記号の左にあるものを右にあるもので定義することを示します。 (例) A : : = B 「A とは B である」と定義することを示します。
...	記述が省略されていることを示します。 (例) ABC... ABC の後ろに記述があり、その記述が省略されていることを示します。
...	この記号の直前の項目を繰り返し、複数個指定できることを示します。 (例) A ... A を複数個指定できることを示します。

## このマニュアルで使用する構文要素記号

このマニュアルで使用する構文要素の種類を次に示します。

構文要素記号	意味
英字	A ~ Z a ~ z
英小文字	a ~ z
英大文字	A ~ Z
数字	0 ~ 9
英数字	A ~ Z a ~ z 0 ~ 9
16進数字	0 ~ 9 A ~ F a ~ f
記号	! " # \$ % & ' ( ) + , _ . / : ; < = > @ [ ] ^ - { } 空白 ¥

注 すべての半角文字を使用してください。

## 常用漢字以外の漢字の使用について

このマニュアルでは、常用漢字を使用することを基本としていますが、次に示す用語については、常用漢字以外の漢字を使用しています。

個所 (かしよ)

## KB (キロバイト) などの単位表記について

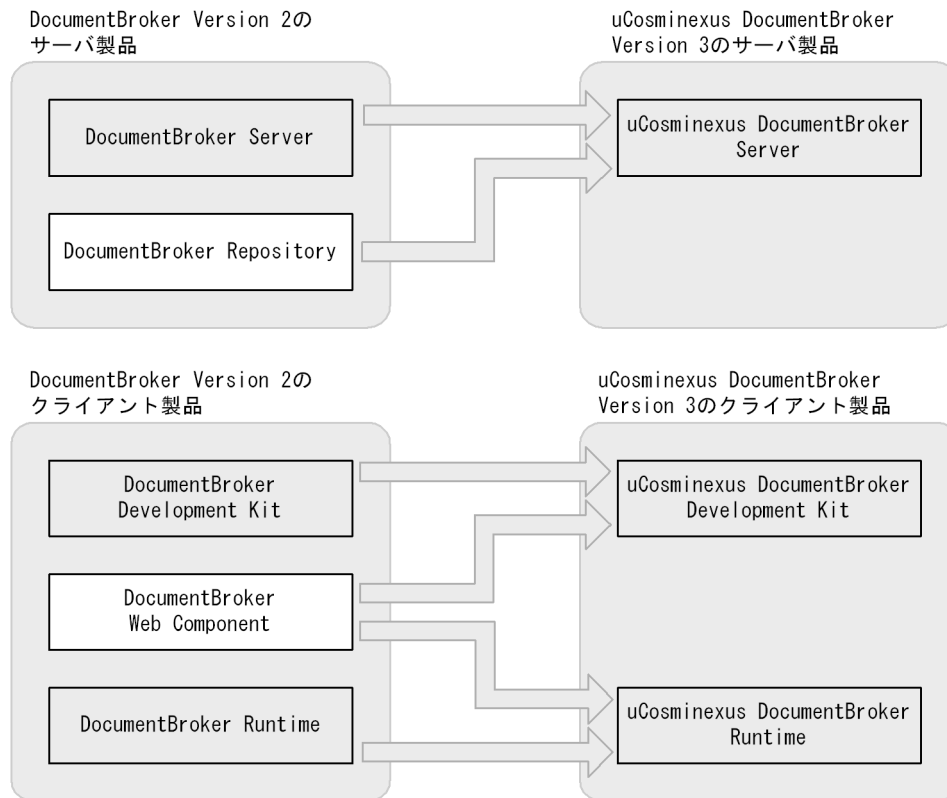
1KB (キロバイト), 1MB (メガバイト), 1GB (ギガバイト), 1TB (テラバイト) はそれぞれ  $1,024$  バイト,  $1,024^2$  バイト,  $1,024^3$  バイト,  $1,024^4$  バイトです。

## DocumentBroker Version 2 と uCosminexus DocumentBroker Version 3 の製品体系の違い

uCosminexus DocumentBroker Version 3 では次のように製品の統合を行いました。

- DocumentBroker Repository を uCosminexus DocumentBroker Server に統合しました。
- DocumentBroker Web Component を uCosminexus DocumentBroker Development Kit および uCosminexus DocumentBroker Runtime に統合しました。

DocumentBroker Version 2 と uCosminexus DocumentBroker Version 3 の製品体系の違いを次に示します。



## DocumentBroker Version 2 と uCosminexus DocumentBroker Version 3 のマニュアルの対応

バージョンアップおよび製品体系の変更に伴い、uCosminexus DocumentBroker Version 3 では次に示すようにマニュアル名称を変更しました。

Version 2 のマニュアル名称	Version 3 のマニュアル名称
DocumentBroker Version 2 システム導入・運用ガイド	uCosminexus DocumentBroker Version 3 システム導入・運用ガイド
DocumentBroker Version 2 クラスライブラリ 解説	uCosminexus DocumentBroker Version 3 クラスライブラリ C++ 解説
DocumentBroker Version 2 クラスライブラリ リファレンス 基本機能編	uCosminexus DocumentBroker Version 3 クラスライブラリ C++ リファレンス 基本機能編
DocumentBroker Version 2 クラスライブラリ リファレンス 概念 SGML 文書管理機能編	廃版
DocumentBroker Version 2 オブジェクト操作ツール	uCosminexus DocumentBroker Version 3 オブジェクト操作ツール
DocumentBroker Version 2 統計解析ツール	uCosminexus DocumentBroker Version 3 統計解析ツール
DocumentBroker Version 2 メッセージ	uCosminexus DocumentBroker Version 3 メッセージ
DocumentBroker Web Component Version 2 解説	uCosminexus DocumentBroker Version 3 クラスライブラリ Java 解説
DocumentBroker Web Component Version 2 リファレンス	uCosminexus DocumentBroker Version 3 クラスライブラリ Java リファレンス

Version 2 のマニュアル名称	Version 3 のマニュアル名称
DocumentBroker Web Component Version 2 サンプル Web アプリケーション	uCosminexus DocumentBroker Version 3 クラスライブラリ Java サンプル Web アプリケーション
DocumentBroker Text Search Index Loader Version 2	uCosminexus DocumentBroker Text Search Index Loader Version 3
DocumentBroker Rendering Option システム導入・運用ガイド	uCosminexus DocumentBroker Rendering Option Version 3
DocumentBroker Object Loader Version 2	uCosminexus DocumentBroker Object Loader Version 3



## 目次

<b>1</b>	<b>Java クラスライブラリのクラス、インターフェース、およびメソッド</b>	<b>1</b>
1.1	ファクトリクラスのインターフェース、およびメソッド一覧	2
1.1.1	ファクトリクラスのクラスおよびインターフェース	2
1.1.2	ファクトリクラスのメソッド一覧	2
1.2	パラメタクラスのインターフェース、およびメソッド一覧	4
1.2.1	パラメタクラスのインターフェース一覧	4
1.2.2	パラメタクラスのメソッド一覧	6
1.3	文書管理クラスのインターフェース、およびメソッド一覧	14
1.3.1	文書管理クラスのインターフェース一覧	14
1.3.2	文書管理クラスのメソッド一覧	15
1.4	メタクラスのインターフェース、およびメソッド一覧	21
1.4.1	メタクラスのインターフェース一覧	21
1.4.2	メタクラスのメソッド一覧	21
1.5	例外クラスのクラス一覧、スーパークラス、およびコンストラクタ	23
1.5.1	例外クラスのクラス一覧	23
1.5.2	例外クラスのスーパークラス	29
1.5.3	例外クラスのコンストラクタ	30
1.6	定数定義クラスのカテゴリ一覧	31
1.7	ライブラリ情報取得クラスのクラス、およびメソッド一覧	32
1.7.1	ライブラリ情報取得クラスのクラス	32
1.7.2	ライブラリ情報取得クラスのメソッド一覧	32
1.8	トレースクラスのクラス、およびメソッド一覧	33
1.8.1	トレースクラスのクラス	33
1.8.2	トレースクラスのメソッド一覧	33
1.9	クラス、インターフェースおよびメソッドの説明形式	34
1.9.1	メソッドで説明する項目	34
1.9.2	パラメタクラスのインターフェースで説明する項目	34
1.9.3	文書空間オブジェクトのプロパティのデータ型	35
<b>2</b>	<b>ファクトリクラス詳細</b>	<b>37</b>
2.1	DbjFactory0200 クラス	38
2.1.1	getFactory (ファクトリインターフェースの取得)	39
2.1.2	getMetaManager (メタマネージャーインターフェースの取得)	40
2.2	DbjFactory インターフェース	41
2.2.1	createACE (ACE の作成)	43
2.2.2	createBooleanQParam (BOOL 型の ? パラメタの作成)	44
2.2.3	createFetchInfo (検索結果取得情報オブジェクトの作成)	45
2.2.4	createInteger32QParam (INT 型の ? パラメタの作成)	47

2.2.5	createOIDQParam (OID の ? パラメタの作成)	48
2.2.6	createObjQParam (Object 型の ? パラメタの作成)	49
2.2.7	createPropSet (プロパティ値集合オブジェクトの作成)	50
2.2.8	createPublicACLIdElm (パブリック ACL の OID エLEMENTの作成)	51
2.2.9	createReferencePathInfo (リファレンスファイル文書のパス情報オブジェクトの作成)	52
2.2.10	createReferenceUploadInfo (リファレンスファイル文書のアップロード情報オブジェクトの作成)	53
2.2.11	createSeedDocQParam (種文章の ? パラメタの作成)	54
2.2.12	createSession (Java クラスライブラリのセッションオブジェクトの作成)	55
2.2.13	createSetDCRLinkInfo (直接型リンク設定情報の作成)	56
2.2.14	createSetRCRLinkInfo (参照型リンク設定情報の作成)	57
2.2.15	createSetRelLinkInfo (文書間リンク設定情報の作成)	58
2.2.16	createSetVTFixLinkInfo (構成管理モードが FIX モードの構成管理型リンク設定情報の作成)	59
2.2.17	createSetVTFloatingLinkInfo (構成管理モードが FLOATING モードの構成管理型リンク設定情報の作成)	60
2.2.18	createStringQParam (String 型の ? パラメタの作成)	61
2.2.19	createUploadInfo (文書のアップロード情報オブジェクトの作成)	62
2.2.20	createVArray (可変長配列オブジェクトの作成)	63
2.2.21	createXmlTranslator (XML トランスレーターの取得)	64
2.2.22	createXmlUploadInfo (XML 文書のアップロード情報オブジェクトの作成)	65

### 3

パラメタクラス詳細	67
3.1 DbjACE インターフェース	69
3.1.1 getPermission (パーミッションの取得)	70
3.1.2 getSubject (サブジェクトの取得)	72
3.1.3 getSubjectType (サブジェクト種別の取得)	73
3.1.4 propSet (ACE のプロパティ値集合の取得)	74
3.1.5 setGroupSubject (サブジェクトをグループサブジェクトとして設定)	75
3.1.6 setPermission (パーミッションの設定)	76
3.1.7 setPropSet (ACE のプロパティ値集合の設定)	78
3.1.8 setSubject (サブジェクトの設定)	79
3.1.9 setSubjectType (サブジェクト種別の設定)	80
3.1.10 setSystemSubject (サブジェクトをシステムサブジェクトとして設定)	81
3.1.11 setUserSubject (サブジェクトをユーザサブジェクトとして設定)	82
3.2 DbjBooleanQParam インターフェース	83
3.3 DbjCheckOutInfo インターフェース	84
3.3.1 getCheckOutUserId (チェックアウトしたユーザのユーザ識別子の取得)	85
3.3.2 getCheckOutVersionId (仮のバージョン識別子の取得)	86
3.3.3 isCheckOut (チェックアウト中かどうかの判定)	87
3.4 DbjContentInfo インターフェース	88
3.4.1 getRenditionType (コンテンツのレンディションタイプの取得)	89
3.4.2 getRetrievalName (コンテンツのファイル名の取得)	90
3.5 DbjFetchInfo インターフェース	91



3.5.1	getCacheKey ( キャッシュキーの取得 )	93
3.5.2	getCacheName ( キャッシュ名の取得 )	94
3.5.3	getCacheTotal ( 検索結果キャッシュの全件数の取得 )	95
3.5.4	getComparator ( Comparator インターフェースの取得 )	96
3.5.5	getFetchCount ( 検索結果の取得件数の取得 )	97
3.5.6	getMaxFetchCount ( 検索結果の最大取得件数の取得 )	98
3.5.7	getStartIndex ( 検索結果の取得開始位置の取得 )	99
3.5.8	setCacheKey ( キャッシュキーの設定 )	100
3.5.9	setCacheName ( キャッシュ名の設定 )	101
3.5.10	setCacheTotal ( 検索結果キャッシュの全件数の設定 )	102
3.5.11	setComparator ( Comparator インターフェースの設定 )	103
3.5.12	setFetchCount ( 検索結果の取得件数の設定 )	104
3.5.13	setMaxFetchCount ( 検索結果の最大取得件数の設定 )	105
3.5.14	setStartIndex ( 検索結果の取得開始位置の設定 )	106
3.6	DbjInteger32QParam インターフェース	107
3.7	DbjMappedProp インターフェース	108
3.7.1	getClassList ( マッピングされた DMA クラス名のリストの取得 )	109
3.7.2	propSet ( マッピングされたプロパティ値集合の取得 )	110
3.8	DbjObjQParam インターフェース	111
3.9	DbjOIDQParam インターフェース	112
3.10	DbjPropSet インターフェース	113
3.10.1	changePropName ( プロパティ名の変更 )	114
3.10.2	changePropNames ( プロパティ名の一括変更 )	115
3.10.3	getIntegerVal ( Integer 型でのプロパティ値の取得 )	116
3.10.4	getIntVal ( int 型でのプロパティ値の取得 )	117
3.10.5	getListRef ( List 型でのプロパティ値の取得 )	118
3.10.6	getStringVal ( String 型でのプロパティ値の取得 )	119
3.10.7	getVArrayRef ( VARRAY 型でのプロパティ値の参照の取得 )	120
3.10.8	getVArrayVal ( VARRAY 型でのプロパティ値の取得 )	121
3.10.9	isNull ( プロパティ値が NULL 値かどうかの判定 )	122
3.10.10	setNull ( NULL 値プロパティの設定 )	123
3.10.11	setPropRef ( プロパティ値の参照の設定 )	124
3.10.12	setPropVal ( プロパティ値の設定 )	125
3.11	DbjPublicACLIdElm インターフェース	127
3.11.1	getId ( パブリック ACL の OID 文字列の取得 )	128
3.11.2	propSet ( パブリック ACL の OID 文字列のプロパティ値集合の取得 )	129
3.11.3	setId ( パブリック ACL の OID 文字列の設定 )	130
3.11.4	setPropSet ( パブリック ACL の OID 文字列のプロパティ値集合の設定 )	131
3.12	DbjQParam インターフェース	132
3.12.1	getVal ( ? パラメタ値の取得 )	133
3.13	DbjReferenceContentInfo インターフェース	134
3.13.1	getContentLocation ( コンテンツロケーションの取得 )	135

3.13.2	getReferenceType (リファレンス種別の取得)	136
3.14	DbjReferencePathInfo インターフェース	137
3.14.1	getContentOperateMode (コンテンツのパス操作モードの取得)	139
3.14.2	getDeleteRootPath (削除するディレクトリのルートパスの取得)	140
3.14.3	getEntry (登録するコンテンツのパスまたはダウンロード先のパスの取得)	141
3.14.4	getTargetPath (コンテンツ格納先パスの取得)	142
3.14.5	setContentOperateMode (コンテンツのパス操作モードの設定)	143
3.14.6	setDeleteRootPath (削除するディレクトリのルートパスの設定)	144
3.14.7	setEntry (登録するコンテンツのパスまたはダウンロード先のパスの設定)	145
3.14.8	setTargetPath (コンテンツ格納先パスの設定)	146
3.15	DbjReferenceUploadInfo インターフェース	147
3.15.1	getReferencePathInfo (リファレンスファイル文書のパス情報の取得)	148
3.15.2	setReferencePathInfo (リファレンスファイル文書のパス情報の設定)	149
3.16	DbjRenditionInfo インターフェース	150
3.16.1	getRenditionType (レンディションタイプの取得)	151
3.16.2	propSet (レンディションのプロパティ値集合の取得)	152
3.17	DbjRenditionList インターフェース	153
3.17.1	getRenditionInfo (レンディション情報の取得)	154
3.17.2	getRenditionTypeList (レンディションタイプのリストの取得)	155
3.18	DbjResultSet インターフェース	156
3.18.1	absolute (カーソルの絶対指定行への移動)	157
3.18.2	afterLast (カーソルの最終行の後ろへの移動)	158
3.18.3	beforeFirst (カーソルの先頭行の前への移動)	159
3.18.4	distinct (列の重複の排除)	160
3.18.5	findColumnName (列インデクスの取得)	161
3.18.6	first (カーソルの先頭行への移動)	162
3.18.7	getColumnCount (列数の取得)	163
3.18.8	getColumnMetaName (列名の取得 (表名を含む))	164
3.18.9	getColumnName (列名の取得)	165
3.18.10	getColumnType (列のデータ型の取得)	166
3.18.11	getColumnVals (列データの取得)	167
3.18.12	getIntegerVal (Integer 型での列データの取得)	168
3.18.13	getIntVal (int 型での列データの取得)	169
3.18.14	getObjectRef (データの参照の取得)	170
3.18.15	getObjectVal (データの取得)	171
3.18.16	getPropSet (行データをプロパティ値集合として取得)	172
3.18.17	getRow (カーソル行インデクスの取得)	173
3.18.18	getRowCount (行数の取得)	174
3.18.19	getRowVals (行データの取得)	175
3.18.20	getStringVal (String 型での列データの取得)	176
3.18.21	getTableName (表名の取得)	177
3.18.22	getVArrayRef (VARRAY 型でのデータの参照の取得)	178

3.18.23	getVArrayVal ( VARRAY 型でのデータの取得 )	179
3.18.24	isAfterLast ( カーソルが最終行の後ろかどうかの判定 )	180
3.18.25	isBeforeFirst ( カーソルが先頭行の前かどうかの判定 )	181
3.18.26	isFirst ( カーソルが先頭行かどうかの判定 )	182
3.18.27	isLast ( カーソルが最終行かどうかの判定 )	183
3.18.28	isNamed ( 名前付き検索結果かどうかの判定 )	184
3.18.29	isNull ( NULL 値かどうかの判定 )	185
3.18.30	last ( カーソルの最終行への移動 )	186
3.18.31	next ( カーソルの次の行への移動 )	187
3.18.32	previous ( カーソルの前の行への移動 )	188
3.18.33	reduct ( 名前なし列の削除 )	189
3.18.34	setColumnMetaName ( 列名の設定 )	190
3.19	DbjSeedDocQParam インターフェース	191
3.20	DbjSetDCRLinkInfo インターフェース	192
3.21	DbjSetLinkInfo インターフェース	193
3.21.1	getLinkType ( リンク種別の取得 )	194
3.21.2	getTargetObj ( リンク先オブジェクトの取得 )	195
3.21.3	propSet ( リンクオブジェクトのプロパティ値集合の取得 )	196
3.21.4	setPropSet ( リンクオブジェクトのプロパティ値集合の設定 )	197
3.21.5	setTargetObj ( リンク先オブジェクトの設定 )	198
3.22	DbjSetRCRLinkInfo インターフェース	199
3.23	DbjSetRelLinkInfo インターフェース	200
3.24	DbjSetVTFixLinkInfo インターフェース	201
3.25	DbjSetVTFloatLinkInfo インターフェース	202
3.26	DbjSetVTLinkInfo インターフェース	203
3.27	DbjStringQParam インターフェース	204
3.28	DbjUploadInfo インターフェース	205
3.28.1	getFilePath ( 登録する文書のフルパスの取得 )	207
3.28.2	getIndexPath ( 全文検索インデックス作成用ファイルのフルパスの取得 )	208
3.28.3	getRenditionType ( 登録する文書のレンディションタイプの取得 )	209
3.28.4	getRetrievalName ( 登録する文書のファイル名の取得 )	210
3.28.5	renditionPropSet ( レン디션プロパティ値集合の取得 )	211
3.28.6	setFilePath ( 登録する文書のフルパスの設定 )	212
3.28.7	setIndexPath ( 全文検索インデックス作成用ファイルのフルパスの設定 )	213
3.28.8	setRenditionPropSet ( レン디션プロパティ値集合の設定 )	214
3.28.9	setRenditionType ( 登録する文書のレンディションタイプの設定 )	215
3.28.10	setRetrievalName ( 登録する文書のファイル名の設定 )	216
3.29	DbjVArray インターフェース	217
3.29.1	addPropSet ( プロパティ値集合 ( 行 ) の追加 )	219
3.29.2	addPropVals ( プロパティ値 ( 列 ) の追加 )	220
3.29.3	changePropName ( 可変長配列オブジェクトのプロパティ名の変更 )	221
3.29.4	changePropNames ( 可変長配列オブジェクトのプロパティ名の一括変更 )	222

3.29.5	getPropCount (メタプロパティ数の取得)	223
3.29.6	getPropNameSet (メタプロパティの取得)	224
3.29.7	getPropVals (可変長配列オブジェクトのプロパティ値のリストの取得)	225
3.29.8	propSet (プロパティ値集合の取得)	226
3.29.9	removeProp (プロパティの削除)	227
3.30	DbjXmlUploadInfo インターフェース	228
3.30.1	clearParseMessage (構文解析結果メッセージのクリア)	230
3.30.2	getFilterFilePath (フィルタリング定義ファイルのフルパスの取得)	231
3.30.3	getIndexType (全文検索インデクスの種別の取得)	232
3.30.4	getMappingId (マッピング定義名の取得)	233
3.30.5	getParseLevel (構文解析レベルの取得)	234
3.30.6	getParseMessage (構文解析結果メッセージの取得)	235
3.30.7	setFilterFilePath (フィルタリング定義ファイルのフルパスの設定)	236
3.30.8	setIndexType (全文検索インデクスの種別の設定)	237
3.30.9	setMappingId (マッピング定義名の設定)	238
3.30.10	setParseLevel (構文解析レベルの設定)	239
4	文書管理クラス詳細	241
4.1	DbjDocSpace インターフェース	242
4.1.1	changeSearchACLMode (検索実行時のアクセス制御モードの変更)	243
4.1.2	createDocument (バージョンなし文書の作成)	244
4.1.3	createFolder (バージョンなしフォルダの作成)	247
4.1.4	createIndependentData (独立データの作成)	249
4.1.5	createLinkObjList (複数のリンクオブジェクトアクセスインターフェースの取得)	251
4.1.6	createObjConnection (文書空間オブジェクトアクセスインターフェースの取得)	252
4.1.7	createObjList (複数の文書空間オブジェクトアクセスインターフェースの取得)	253
4.1.8	createPublicACL (パブリック ACL の作成)	255
4.1.9	createVrDocument (バージョン付き文書の作成)	257
4.1.10	createVrFolder (バージョン付きフォルダの作成)	260
4.1.11	executeSearch (検索の実行)	262
4.1.12	getMeta (文書空間のメタ情報の取得)	265
4.1.13	getSearchACLMode (検索実行時のアクセス制御モードの取得)	266
4.1.14	isAccessControlMode (アクセス制御モードの取得)	267
4.1.15	removeObjects (検索条件を指定したオブジェクトの削除)	268
4.2	DbjLinkObj インターフェース	269
4.2.1	changeToVTFix (構成管理型リンクの構成管理モードを FIX モードに変更)	270
4.2.2	changeToVTFloat (構成管理型リンクの構成管理モードを FLOATING モードに変更)	271
4.2.3	getLinkId (リンク識別子の取得)	272
4.2.4	getLinkType (リンク種別の取得)	273
4.2.5	getOwnerObj (リンク元オブジェクトの取得)	274
4.2.6	getTargetObj (リンク先オブジェクトの取得)	275
4.2.7	propSet (リンク Proxy オブジェクトのプロパティ値集合インターフェースの取得)	276

4.2.8	readProperties (リンクプロパティの取得)	277
4.2.9	removeObject (リンクオブジェクトの削除)	278
4.2.10	setPropSet (リンク Proxy オブジェクトのプロパティ値集合の設定)	279
4.2.11	writeProperties (リンクプロパティ値の設定)	280
4.3	DbjLinkObjList インターフェース	282
4.3.1	changeToVTFix (複数の構成管理型リンクの構成管理モードを一括して FIX モードに変更)	283
4.3.2	changeToVTFloat (複数の構成管理型リンクの構成管理モードを一括して FLOATING モードに変更)	284
4.3.3	getLinkIdList (ターゲットリンク識別子プロパティのリストの取得)	285
4.3.4	getLinkObj (リスト要素の DbjLinkObj インターフェースの取得)	286
4.3.5	getOwnerObjList (リンク元オブジェクトのリストの取得)	287
4.3.6	getTargetObjList (リンク先オブジェクトのリストの取得)	288
4.3.7	readProperties (リンクプロパティ値の一括取得)	289
4.3.8	removeObjects (リンクオブジェクトの一括削除)	290
4.3.9	writeProperties (リンクプロパティ値の一括設定)	291
4.4	DbjObj インターフェース	293
4.4.1	addRendition (レンディションの追加)	294
4.4.2	bindPublicACL (パブリック ACL のバインド)	296
4.4.3	cancelCheckOut (バージョン付きオブジェクトのチェックアウトの取り消し)	298
4.4.4	changeMasterRendition (マスタレンディションの変更)	300
4.4.5	changeToVTFix (構成管理型リンクの構成管理モードを FIX モードに変更)	302
4.4.6	changeToVTFloat (構成管理型リンクの構成管理モードを FLOATING モードに変更)	304
4.4.7	checkIn (バージョン付きオブジェクトのチェックイン)	306
4.4.8	checkOut (バージョン付きオブジェクトのチェックアウト)	308
4.4.9	deleteRendition (レンディションの削除)	310
4.4.10	deleteVersion (バージョンの削除)	311
4.4.11	downloadContents (文書のコンテンツのダウンロード)	313
4.4.12	getBindObjectList (パブリック ACL にバインドしている文書空間オブジェクト一覧の取得)	315
4.4.13	getCheckOutStatus (バージョン付きオブジェクトのチェックアウト状態の取得)	317
4.4.14	getChildList (フォルダのリンク先オブジェクト (下位オブジェクト) の取得)	318
4.4.15	getClassName (アクセス対象文書空間オブジェクトを構成する DMA クラス名の取得)	321
4.4.16	getDCRParent (直接型リンクによるリンク元オブジェクトの取得)	322
4.4.17	getLockType (Proxy オブジェクトのアクセスロック種別の取得)	323
4.4.18	getObjType (アクセス対象文書空間オブジェクトのオブジェクト種別の取得)	324
4.4.19	getOiid (アクセス対象文書空間オブジェクトの OIID の取得)	325
4.4.20	getParentList (フォルダのリンク元オブジェクトの取得)	326
4.4.21	getPublicACLList (バインドしているパブリック ACL 一覧の取得)	328
4.4.22	getRelList (文書間リンク一覧の取得)	329
4.4.23	getRenditionList (レンディション情報一覧の取得)	331
4.4.24	getTargetVersion (アクセス対象ターゲットバージョンのバージョン識別子の取得)	332
4.4.25	getVersionId (バージョンオブジェクトのバージョン識別子の取得)	333
4.4.26	getVersioningInfo (バージョン付きオブジェクトのバージョンングオブジェクトの取得)	334
4.4.27	getVersionObjList (バージョン付きオブジェクトのバージョン一覧の取得)	335

4.4.28	link (リンク先オブジェクトとのリンク)	337
4.4.29	lock (アクセスロック種別の異なる文書空間オブジェクトインターフェースの取得)	339
4.4.30	move (直接型リンクが設定されている文書空間オブジェクトの移動)	340
4.4.31	propSet (Proxy オブジェクトのプロパティ値集合インターフェースの取得)	341
4.4.32	readProperties (文書空間オブジェクトのプロパティ値の読み込み)	342
4.4.33	removeObject (文書空間オブジェクトの削除)	344
4.4.34	setPropSet (Proxy オブジェクトのプロパティ値集合の設定)	346
4.4.35	setTargetVersion (アクセス対象ターゲットバージョンのバージョン識別子の変更)	347
4.4.36	unbindPublicACL (パブリック ACL のアンバインド)	348
4.4.37	unlink (リンク先オブジェクトとのリンクの解除)	350
4.4.38	unlinkByLinkId (リンク識別子の指定によるリンク先オブジェクトとのリンクの解除)	352
4.4.39	uploadContents (文書のコンテンツのアップロード)	353
4.4.40	writeProperties (文書空間オブジェクトのプロパティ値の設定)	355
4.4.41	writeRenditionProperties (レンディションプロパティの設定)	357
4.5	DbjObjList インターフェース	359
4.5.1	getObj (リスト要素の取得)	360
4.5.2	lock (アクセスロック種別の異なる複数文書空間オブジェクトインターフェースの取得)	361
4.5.3	move (直接型リンクが設定されている文書空間オブジェクトの一括移動)	362
4.5.4	readProperties (複数の文書空間オブジェクトプロパティ値の一括取得)	363
4.5.5	removeObjects (文書空間オブジェクトの一括削除)	365
4.5.6	setPropSet (要素のプロパティ値集合の設定)	366
4.5.7	writeProperties (複数の文書空間オブジェクトプロパティ値の一括設定)	367
4.6	DbjSession インターフェース	369
4.6.1	begin (トランザクションの開始)	370
4.6.2	checkSession (セッションが有効かどうかのチェック)	371
4.6.3	commit (トランザクションの確定)	372
4.6.4	getLoginUserInfo (ユーザ情報の取得)	373
4.6.5	getReferencePath (コンテンツ格納先ベースパスの取得)	374
4.6.6	login (ログイン)	375
4.6.7	logout (ログアウト)	376
4.6.8	rollback (トランザクションの取り消し)	377
4.6.9	setReferencePath (コンテンツ格納先ベースパスの設定)	378
4.7	DbjVerObj インターフェース	379
4.7.1	getVersionId (文書空間オブジェクトのバージョン識別子の取得)	380
4.8	DbjVerObjList インターフェース	381
4.8.1	getVerObj (要素の DbjVerObj インターフェースの取得)	382
4.8.2	getVersionIdList (文書空間オブジェクトのバージョン識別子リストの取得)	383
4.9	DbjXmlTranslator インターフェース	384
4.9.1	getMappedProperties (XML 文書の XML プロパティマッピング情報の作成)	385
5	メタクラス詳細	387
5.1	DbjClassDesc インターフェース	388

5.1.1	getName ( クラス名の取得 )	389
5.1.2	getProperties ( プロパティディスクリプションの取得 )	390
5.1.3	getSubClasses ( サブクラスのクラスディスクリプションの取得 )	391
5.1.4	getSuperClass ( スーパークラスのクラスディスクリプションの取得 )	392
5.2	DbjMeta インターフェース	393
5.2.1	getClassDesc ( 指定したクラスのクラスディスクリプションの取得 )	394
5.2.2	getDocSpaceId ( 文書空間識別子の取得 )	395
5.2.3	getExtFromRenditionType ( レン디션タイプに対応する拡張子の取得 )	396
5.2.4	getPropDataType ( 指定したプロパティのデータ型の取得 )	397
5.2.5	getPropDesc ( 指定したプロパティのプロパティディスクリプションの取得 )	398
5.2.6	getRenditionType ( 拡張子に対応するレン디션タイプの取得 )	399
5.3	DbjMetaManager インターフェース	400
5.3.1	getMeta ( メタ情報の取得 )	401
5.4	DbjPropDesc インターフェース	402
5.4.1	getDataType ( プロパティのデータ型の取得 )	403
5.4.2	getName ( プロパティ名の取得 )	404
5.4.3	getVArrayClass ( VARRAY 型プロパティを扱うクラスのクラスディスクリプションの取得 )	405

## 6

例外クラス詳細	407	
6.1	例外クラスの詳細	410
6.2	DbjAccessControlException クラス	411
6.3	DbjAccessControlNotSupportedException クラス	412
6.4	DbjACEOperationException クラス	413
6.5	DbjACLOutOfRangeException クラス	414
6.6	DbjAlreadyCheckOutException クラス	415
6.7	DbjCheckOutException クラス	416
6.8	DbjContentNotRegisteredException クラス	417
6.9	DbjContentTypeMismatchException クラス	418
6.10	DbjCORBAException クラス	419
6.11	DbjDBDeadLockException クラス	420
6.12	DbjDBException クラス	421
6.13	DbjDBLockTimeoutException クラス	422
6.14	DbjDisconnectedSessionException クラス	423
6.15	DbjError クラス	424
6.16	DbjException クラス	425
6.17	DbjFileAccessException クラス	426
6.18	DbjFileNotFoundException クラス	427
6.19	DbjFileReferenceCurrentContentNotfoundException クラス	428
6.20	DbjFileReferenceMismatchStatusException クラス	429
6.21	DbjFileReferenceOperationFailedException クラス	430
6.22	DbjIllegalCacheStartIndexException クラス	431

6.23	DbjIllegalDocSpaceIdException クラス	432
6.24	DbjIllegalObjectTypeException クラス	433
6.25	DbjIllegalPropValException クラス	434
6.26	DbjInitializeError クラス	435
6.27	DbjInternalError クラス	436
6.28	DbjIOException クラス	437
6.29	DbjIsMasterRenditionException クラス	438
6.30	DbjLastVersionException クラス	439
6.31	DbjMasterRenditionNotSetException クラス	440
6.32	DbjNotAuthenticatedException クラス	441
6.33	DbjNotCheckoutException クラス	442
6.34	DbjNotLoginException クラス	443
6.35	DbjObjectNotFoundException クラス	444
6.36	DbjOutOfMemoryError クラス	445
6.37	DbjPublicACLAlreadyBoundException クラス	446
6.38	DbjPublicACLNotBoundException クラス	447
6.39	DbjPublicACLNotFoundExcepion クラス	448
6.40	DbjPublicACLOperationException クラス	449
6.41	DbjPublicACLOutOfRangeException クラス	450
6.42	DbjReferenceTypeMismatchException クラス	451
6.43	DbjRenditionCountOutOfRangeExcepion クラス	452
6.44	DbjRenditionIsEmptyException クラス	453
6.45	DbjRenditionNotConvertedException クラス	454
6.46	DbjRenditionNotFoundExcepion クラス	455
6.47	DbjRenditionTypeDuplicatedException クラス	456
6.48	DbjSessionException クラス	457
6.49	DbjSessionNotConnectException クラス	458
6.50	DbjSessionOverflowException クラス	459
6.51	DbjSubjectLengthOutOfRangeExcepion クラス	460
6.52	DbjTargetContentPathNotSetException クラス	461
6.53	DbjUnknownError クラス	462
6.54	DbjXmlParseException クラス	463

## 7

定数定義クラス詳細		465
7.1	DbjDef クラス	466
7.1.1	Category : ACL	467
7.1.2	Category : DATATYPE	468
7.1.3	Category : DMA BOOLEAN	469
7.1.4	Category : INDEXTYPE	470
7.1.5	Category : LINK	471



7.1.6	Category : LOCK	472
7.1.7	Category : OBJTYPE	473
7.1.8	Category : OPERATEMODE	474
7.1.9	Category : ORDER	475
7.1.10	Category : PERM	476
7.1.11	Category : PRIV	478
7.1.12	Category : REFERENCETYPE	479
7.1.13	Category : RELATIONEND	480
7.1.14	Category : RENDSTATUS	481
7.1.15	Category : SUBJECTTYPE	482
7.1.16	Category : SYSSUBJECT	483
7.1.17	Category : XMLPARSE	484
7.1.18	Category : Others	485
7.2	DbjTraceDef クラス	486
7.2.1	Category : TRACELEVEL	487
7.2.2	Category : TRACEOUTPUT	488

8	ライブラリ情報取得クラス詳細	489
8.1	DbjLibInfo クラス	490
8.1.1	getVersion (バージョンの取得)	491

9	トレースクラス詳細	493
9.1	DbjTrace クラス	494
9.1.1	arg (パラメタ情報の出力)	495
9.1.2	call (外部 API の呼び出し情報の出力)	497
9.1.3	DbjTrace (コンストラクタ)	499
9.1.4	enter (メソッドの入り口情報の出力)	500
9.1.5	error (エラー情報の出力)	502
9.1.6	exit (メソッドの出口情報の出力)	505
9.1.7	hint (下位のメソッドのエラー情報の出力)	506
9.1.8	init (トレースクラスの初期化)	508
9.1.9	msg (メッセージの出力)	510
9.1.10	returned (外部 API からのリターン情報の出力)	513

10	メッセージの一覧	515
10.1	メッセージの形式と見方	516
10.2	メッセージの詳細	518

<b>付録</b>	547
付録 A 文書空間で使用する文字コード種別が UTF-8 の場合の機能差異	548
付録 B 用語解説	550

<b>索引</b>	569
-----------	-----

# 1

## Java クラスライブラリのクラス， インターフェース，およびメソッド

この章では，Java クラスライブラリで使用するクラス，インターフェース，およびメソッドについて説明します。文書空間で使用する文字コード種別が UTF-8 の場合のクラス，インターフェースおよびメソッドの機能差異については，「付録 A 文書空間で使用する文字コード種別が UTF-8 の場合の機能差異」を参照してください。

- 
- 1.1 ファクトリクラスのインターフェース，およびメソッド一覧
  - 1.2 パラメタクラスのインターフェース，およびメソッド一覧
  - 1.3 文書管理クラスのインターフェース，およびメソッド一覧
  - 1.4 メタクラスのインターフェース，およびメソッド一覧
  - 1.5 例外クラスのクラス一覧，スーパークラス，およびコンストラクタ
  - 1.6 定数定義クラスのカテゴリ一覧
  - 1.7 ライブラリ情報取得クラスのクラス，およびメソッド一覧
  - 1.8 トレースクラスのクラス，およびメソッド一覧
  - 1.9 クラス，インターフェースおよびメソッドの説明形式
-

## 1.1 ファクトリクラスのインターフェース、およびメソッド一覧

ファクトリクラスのクラス、およびメソッドを一覧形式で説明します。

### 1.1.1 ファクトリクラスのクラスおよびインターフェース

ファクトリクラスのクラスおよびインターフェースを表 1-1 に示します。なお、それぞれのクラスおよびインターフェースの使用方法については、マニュアル「DocumentBroker Version 3 クラスライブラリ Java 解説」の「6.4 ファクトリクラス」を参照してください。

表 1-1 ファクトリクラスのクラスおよびインターフェース一覧

クラス・インターフェース	説明
ファクトリインターフェースを取得するためのクラス	
DbjFactory0200 クラス	Java クラスライブラリのファクトリインターフェースを取得するためのクラス
ファクトリインターフェース	
DbjFactory インターフェース	Java クラスライブラリのオブジェクトを生成し、そのインターフェースを返却するファクトリインターフェース

### 1.1.2 ファクトリクラスのメソッド一覧

ファクトリクラスのクラス、およびインターフェースのメソッドの一覧を示します。

#### (1) DbjFactory0200 クラス (Java クラスライブラリの DbjFactory インターフェースを取得するためのクラス)

DbjFactory0200 クラスのメソッド一覧を表 1-2 に示します。

表 1-2 DbjFactory0200 クラスのメソッド一覧

メソッド	機能
getFactory	ファクトリインターフェースの取得
getMetaManager	メタマネージャーインターフェースの取得

#### (2) DbjFactory インターフェース (オブジェクトを生成するためのインターフェース)

DbjFactory インターフェースのメソッド一覧を表 1-3 に示します。

表 1-3 DbjFactory インターフェースのメソッド一覧

メソッド	機能
Java クラスライブラリ上のオブジェクトを作成するメソッド	
createACE	ACE の作成
createBooleanQParam	BOOL 型の ? パラメタの作成
createFetchInfo	検索結果取得情報オブジェクトの作成
createInteger32QParam	INT 型の ? パラメタの作成

1. Java クラスライブラリのクラス、インターフェース、およびメソッド

メソッド	機能
createOIIDQParam	OIID の ? パラメタの作成
createObjQParam	Object 型の ? パラメタの作成
createPropSet	プロパティ値集合オブジェクトの作成
createPublicACLIdElm	パブリック ACL の OIID エLEMENTの作成
createReferencePathInfo	リファレンスファイル文書のパス情報オブジェクトの作成
createReferenceUploadInfo	リファレンスファイル文書のアップロード情報オブジェクトの作成
createSeedDocQParam	種文章の ? パラメタの作成
createSession	Java クラスライブラリのセッションオブジェクトの作成
createSetDCRLinkInfo	直接型リンク設定情報の作成
createSetRCRLinkInfo	参照型リンク設定情報の作成
createSetRelLinkInfo	文書間リンク設定情報の作成
createSetVTFixLinkInfo	構成管理モードが FIX モードの構成管理型リンク設定情報の作成
createSetVTFloatLinkInfo	構成管理モードが FLOATING モードの構成管理型リンク設定情報の作成
createStringQParam	STR 型の ? パラメタの作成
createUploadInfo	文書のアップロード情報オブジェクトの作成
createVArray	可変長配列オブジェクトの作成
createXmlTranslator	XML トランスレーターの取得
createXmlUploadInfo	XML 文書のアップロード情報オブジェクトの作成

## 1.2 パラメタクラスのインターフェース、およびメソッド一覧

パラメタクラスのインターフェース、およびメソッドを一覧形式で説明します。

### 1.2.1 パラメタクラスのインターフェース一覧

パラメタクラスのインターフェースの一覧を表 1-4 に示します。なお、それぞれのインターフェースの機能については、マニュアル「DocumentBroker Version 3 クラスライブラリ Java 解説」の「6.5 パラメタの操作」を参照してください。

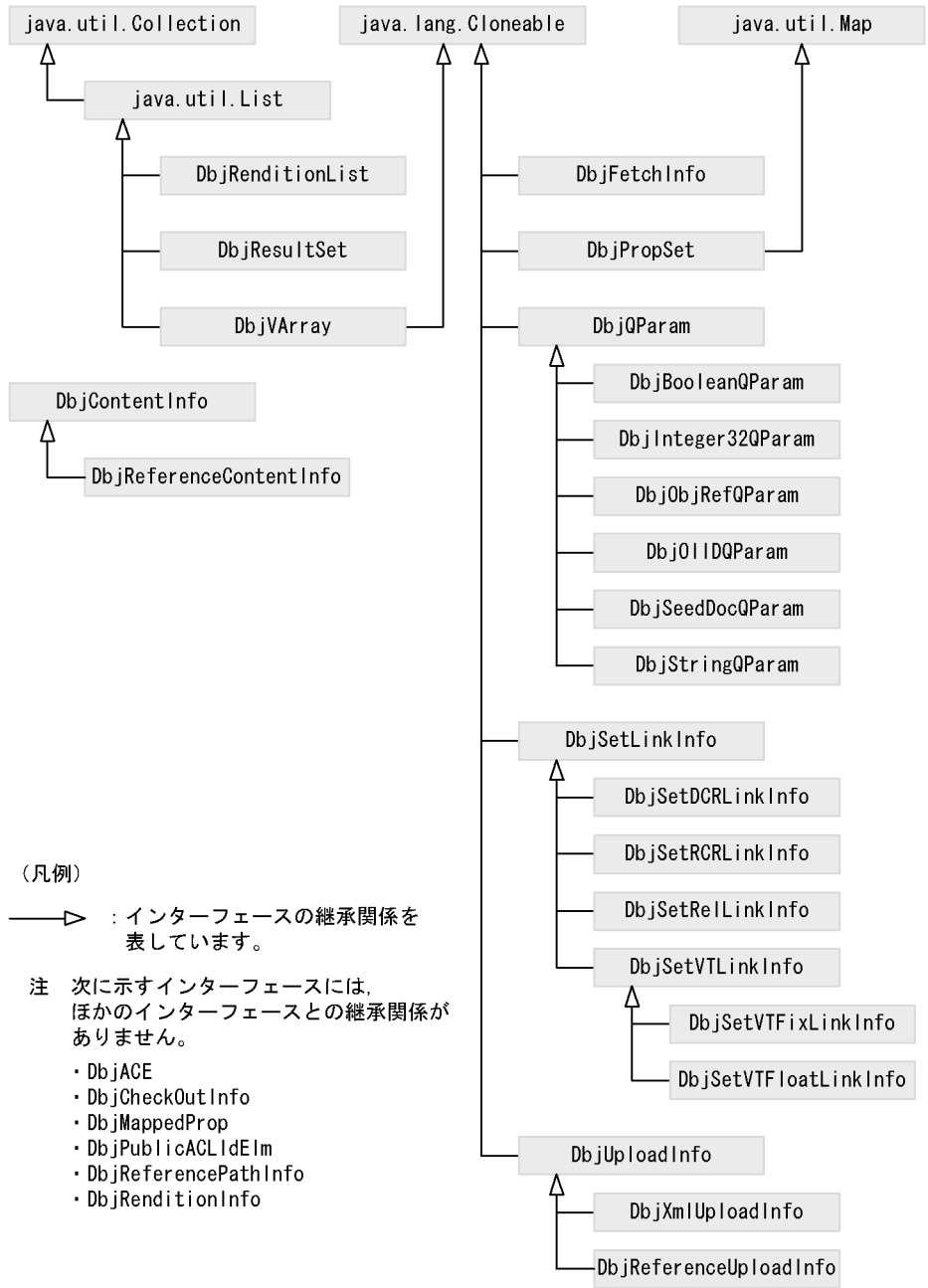
表 1-4 パラメタクラスのインターフェース一覧

インターフェース	説明
? パラメタを扱うインターフェース	
DbjBooleanQParam インターフェース	BOOL 型を表す? パラメタを扱うインターフェース
DbjInteger32QParam インターフェース	INT 型を表す? パラメタを扱うインターフェース
DbjObjQParam インターフェース	文書空間オブジェクトを表す? パラメタを扱うインターフェース
DbjOIDQParam インターフェース	OID 文字列を表す? パラメタを扱うインターフェース
DbjQParam インターフェース	検索メソッドで指定する? パラメタリストの要素となるインターフェースのスーパーインターフェース
DbjSeedDocQParam インターフェース	種文章を表す? パラメタを扱うインターフェース
DbjStringQParam インターフェース	STR 型を表す? パラメタを扱うインターフェース
XML プロパティマッピング結果情報を扱うインターフェース	
DbjMappedProp インターフェース	XML プロパティマッピング機能を実行して取得した情報を扱うインターフェース
アクセス制御用プロパティを扱うインターフェース	
DbjACE インターフェース	ACE の値を扱うインターフェース
DbjPublicACLIdElm インターフェース	文書空間オブジェクトにバインドするパブリック ACL の OID 文字列を扱うインターフェース
アップロード情報を扱うインターフェース	
DbjReferenceUploadInfo インターフェース	リファレンスファイル文書のアップロード情報を扱うインターフェース
DbjUploadInfo インターフェース	文書のアップロード情報を扱うインターフェース
DbjXmlUploadInfo インターフェース	XML 文書のアップロード情報を扱うインターフェース
可変長配列オブジェクトを扱うインターフェース	
DbjVArray インターフェース	VARRAY 型プロパティの値である可変長配列オブジェクトを扱うインターフェース
検索結果集合を扱うインターフェース	
DbjResultSet インターフェース	検索結果集合を扱うインターフェース
検索結果取得情報を扱うインターフェース	
DbjFetchInfo インターフェース	検索結果取得情報を扱うインターフェース
チェックアウト情報を扱うインターフェース	

インターフェース	説明
DbjCheckOutInfo インターフェース	バージョン付きオブジェクトのチェックアウト情報を扱うインターフェース
プロパティ値集合を扱うインターフェース	
DbjPropSet インターフェース	プロパティ値集合を扱うインターフェース
文書のコンテンツ情報を扱うインターフェース	
DbjContentInfo インターフェース	文書のコンテンツ情報を扱うインターフェース
DbjReferenceContentInfo インターフェース	リファレンスファイル文書のコンテンツ情報を扱うインターフェース
文書のパス情報を扱うインターフェース	
DbjReferencePathInfo インターフェース	リファレンスファイル文書のパス情報を扱うインターフェース
リンクの設定情報を扱うインターフェース	
DbjSetDCRLinkInfo インターフェース	直接型リンクの設定情報を扱うインターフェース
DbjSetLinkInfo インターフェース	リンクの設定情報を扱うインターフェース
DbjSetRCRLinkInfo インターフェース	参照型リンクの設定情報を扱うインターフェース
DbjSetRelLinkInfo インターフェース	文書間リンクの設定情報を扱うインターフェース
DbjSetVTFFixLinkInfo インターフェース	構成管理型リンク（構成管理モードは FIX モード）の設定情報を扱うインターフェース
DbjSetVTFLOATLinkInfo インターフェース	構成管理型リンク（構成管理モードは FLOATING モード）の設定情報を扱うインターフェース
DbjSetVTLinkInfo インターフェース	構成管理型リンクの設定情報を扱うインターフェースのスーパーインターフェース
レンディション情報を扱うインターフェース	
DbjRenditionInfo インターフェース	レンディション情報を扱うインターフェース
DbjRenditionList インターフェース	レンディション情報のリストを扱うインターフェース

Java クラスライブラリで提供しているこれらのインターフェースには継承関係があり、上位にあるインターフェースで定義したプロパティやメソッドは、下位のインターフェースに継承されます。インターフェースの継承関係を図 1-1 に示します。

図 1-1 パラメタクラスのインターフェースの継承関係



## 1.2.2 パラメタクラスのメソッド一覧

パラメタクラスの各インターフェースのメソッドの一覧を示します。

### (1) DbjACE インターフェース (ACE の値を扱うインターフェース)

DbjACE インターフェースのメソッド一覧を表 1-5 に示します。

表 1-5 DbjACE インターフェースのメソッド一覧

メソッド	機能
ACE の値を取得するメソッド	



メソッド	機能
getPermission	パーミッションの取得
getSubject	サブジェクトの取得
getSubjectType	サブジェクト種別の取得
propSet	ACEのプロパティ値集合の取得
ACEの値を設定するメソッド	
setPermission	パーミッションの設定
setPropSet	ACEのプロパティ値集合の設定
setSubject	サブジェクトの設定
setSubjectType	サブジェクト種別の設定
ACEのサブジェクト種別を設定するメソッド	
setGroupSubject	サブジェクトをグループサブジェクトとして設定
setSystemSubject	サブジェクトをシステムサブジェクトとして設定
setUserSubject	サブジェクトをユーザサブジェクトとして設定

## (2) DbjCheckOutInfo インターフェース (バージョン付きオブジェクトのチェックアウト情報を扱うインターフェース)

DbjCheckOutInfo インターフェースのメソッド一覧を表 1-6 に示します。

表 1-6 DbjCheckOutInfo インターフェースのメソッド一覧

メソッド	機能
チェックアウト情報を取得するメソッド	
getCheckOutUserId	チェックアウトしたユーザのユーザ識別子の取得
getCheckOutVersionId	仮のバージョン識別子の取得
isCheckOut	チェックアウト中かどうかの判定

## (3) DbjContentInfo インターフェース (文書のコンテンツ情報を扱うインターフェース)

DbjContentInfo インターフェースのメソッド一覧を表 1-7 に示します。

表 1-7 DbjContentInfo インターフェースのメソッド一覧

メソッド	機能
文書のコンテンツ情報を取得するメソッド	
getRenditionType	コンテンツのレンディションタイプの取得
getRetrievalName	コンテンツのファイル名の取得

## (4) DbjFetchInfo インターフェース (検索結果取得情報を扱うインターフェース)

DbjFetchInfo インターフェースのメソッド一覧を表 1-8 に示します。

表 1-8 DbjFetchInfo インターフェースのメソッド一覧

メソッド	機能
検索結果取得情報を取得するメソッド	

1. Java クラスライブラリのクラス、インターフェース、およびメソッド

メソッド	機能
getCacheKey	キャッシュキーの取得
getCacheName	キャッシュ名の取得
getCacheTotal	検索結果キャッシュの全件数の取得
getComparator	Comparator インターフェースの取得
getFetchCount	検索結果の取得件数の取得
getMaxFetchCount	検索結果の最大取得件数の取得
getStartIndex	検索結果の取得開始位置の取得
検索結果取得情報を設定するメソッド	
setCacheKey	キャッシュキーの設定
setCacheName	キャッシュ名の設定
setCacheTotal	検索結果キャッシュの全件数の設定
setComparator	Comparator インターフェースの設定
setFetchCount	検索結果の取得件数の設定
setMaxFetchCount	検索結果の最大取得件数の設定
setStartIndex	検索結果の取得開始位置の設定

(5) DbjMappedProp インターフェース (XML プロパティマッピング結果情報を扱うインターフェース)

DbjMappedProp インターフェースのメソッド一覧を表 1-9 に示します。

表 1-9 DbjMappedProp インターフェースのメソッド一覧

メソッド	機能
XML プロパティマッピング結果情報を取得するメソッド	
getClassList	マッピングされた DMA クラス名のリストの取得
propSet	マッピングされたプロパティ値集合の取得

(6) DbjPropSet インターフェース (プロパティ値集合を扱うインターフェース)

DbjPropSet インターフェースのメソッド一覧を表 1-10 に示します。

表 1-10 DbjPropSet インターフェースのメソッド一覧

メソッド	機能
プロパティ値を取得するメソッド	
getIntegerVal	Integer 型でのプロパティ値の取得
getIntVal	int 型でのプロパティ値の取得
getListRef	List 型でのプロパティ値の取得
getStringVal	String 型でのプロパティ値の取得
getVArrayRef	VARRAY 型でのプロパティ値の参照の取得
getVArrayVal	VARRAY 型でのプロパティ値の取得
isNull	プロパティ値が NULL 値かどうかの判定
プロパティ値を設定するメソッド	

メソッド	機能
setNull	NULL 値プロパティの設定
setPropRef	プロパティ値の参照の設定
setPropVal	プロパティ値の設定
プロパティ名を変更するメソッド	
changePropName	プロパティ名の変更
changePropNames	プロパティ名の一括変更

### (7) DbjPublicACLIdElm インターフェース (パブリック ACL の OIID 文字列を扱うインターフェース)

DbjPublicACLIdElm インターフェースのメソッド一覧を表 1-11 に示します。

表 1-11 DbjPublicACLIdElm インターフェースのメソッド一覧

メソッド	機能
パブリック ACL の OIID 文字列を取得するメソッド	
getId	パブリック ACL の OIID 文字列の取得
propSet	パブリック ACL の OIID 文字列のプロパティ値集合の取得
パブリック ACL の OIID 文字列を設定するメソッド	
setId	パブリック ACL の OIID 文字列の設定
setPropSet	パブリック ACL の OIID 文字列のプロパティ値集合の設定

### (8) DbjQParam インターフェース

DbjQParam インターフェースのメソッド一覧を表 1-12 に示します。

表 1-12 DbjQParam インターフェースのメソッド一覧

メソッド	機能
? パラメタ値を取得するメソッド	
getVal	? パラメタ値の取得

### (9) DbjReferenceContentInfo インターフェース (リファレンスファイル文書のコンテンツ情報を扱うインターフェース)

DbjReferenceContentInfo インターフェースのメソッド一覧を表 1-13 に示します。

表 1-13 DbjReferenceContentInfo インターフェースのメソッド一覧

メソッド	機能
リファレンスファイル文書のコンテンツ情報を取得するメソッド	
getContentLocation	コンテンツロケーションの取得
getReferenceType	リファレンス種別の取得

( 10 ) DbjReferencePathInfo インターフェース ( リファレンスファイル文書のパス情報を扱うインターフェース )

DbjReferencePathInfo インターフェースのメソッド一覧を表 1-14 に示します。

表 1-14 DbjReferencePathInfo インターフェースのメソッド一覧

メソッド	機能
リファレンスファイル文書のパス情報を取得するメソッド	
getContentOperateMode	コンテンツのパス操作モードの取得
getDeleteRootPath	削除するディレクトリのルートパスの取得
getEntry	登録するコンテンツのパスまたはダウンロード先のパスを取得
getTargetPath	コンテンツ格納先パスの取得
リファレンスファイル文書のパス情報を設定するメソッド	
setContentOperateMode	コンテンツのパス操作モードの設定
setDeleteRootPath	削除するディレクトリのルートパスの設定
setEntry	登録するコンテンツのパスまたはダウンロード先のパスの設定
setTargetPath	コンテンツ格納先パスの設定

( 11 ) DbjReferenceUploadInfo インターフェース ( リファレンスファイル文書のアップロード情報を扱うインターフェース )

DbjReferenceUploadInfo インターフェースのメソッド一覧を表 1-15 に示します。

表 1-15 DbjReferenceUploadInfo インターフェースのメソッド一覧

メソッド	機能
リファレンスファイル文書のアップロード情報を取得するメソッド	
getReferencePathInfo	リファレンスファイル文書のパス情報の取得
リファレンスファイル文書のアップロード情報を設定するメソッド	
setReferencePathInfo	リファレンスファイル文書のパス情報の設定

( 12 ) DbjRenditionInfo インターフェース ( レンディション情報を扱うインターフェース )

DbjRenditionInfo インターフェースのメソッド一覧を表 1-16 に示します。

表 1-16 DbjRenditionInfo インターフェースのメソッド一覧

メソッド	機能
レンディションタイプを取得するメソッド	
getRenditionType	レンディションタイプの取得
propSet	レンディションタイプのプロパティ値集合の取得

( 13 ) DbjRenditionList インターフェース ( レンディション情報のリストを扱うインターフェース )

DbjRenditionList インターフェースのメソッド一覧を表 1-17 に示します。

表 1-17 DbjRenditionList インターフェースのメソッド一覧

メソッド	機能
レンディションタイプのリストを取得するメソッド	
getRenditionInfo	レンディション情報の取得
getRenditionTypeList	レンディションタイプのリストの取得

## ( 14 ) DbjResultSet インターフェース ( 検索結果集合を扱うインターフェース )

DbjResultSet インターフェースのメソッド一覧を表 1-18 に示します。

表 1-18 DbjResultSet インターフェースのメソッド一覧

メソッド	機能
カーソルを操作するメソッド	
absolute	カーソルの絶対指定行への移動
afterLast	カーソルの最終行の後ろへの移動
beforeFirst	カーソルの先頭行の前への移動
first	カーソルの先頭行への移動
getRow	カーソル行インデックスの取得
isAfterLast	カーソルが最終行の後ろかどうかの判定
isBeforeFirst	カーソルが先頭行の前かどうかの判定
isFirst	カーソルが先頭行かどうかの判定
isLast	カーソルが最終行かどうかの判定
last	カーソルの最終行への移動
next	カーソルの次の行への移動
previous	カーソルの前の行への移動
検索結果集合のメタ情報を扱うメソッド	
findColumnName	列インデックスの取得
getColumnMetaName	列名の取得 ( 表名を含む )
columnName	列名の取得
getColumnType	列のデータ型の取得
getTableName	表名の取得
isNamed	名前付き検索結果かどうかの判定
setColumnMetaName	列名の設定
検索結果集合のサイズを扱うメソッド	
getColumnCount	列数の取得
getRowCount	行数の取得
指定した行または列のデータを取得するメソッド	
getColumnVals	列データの取得
getIntegerVal	Integer 型での列データの取得
getIntVal	int 型での列データの取得
getObjectRef	データの参照の取得
getObjectVal	データの取得

1. Java クラスライブラリのクラス、インターフェース、およびメソッド

メソッド	機能
getPropSet	行データをプロパティ値集合として取得
getRowVals	行データの取得
getStringVal	String 型での列データの取得
getVArrayRef	VARRAY 型でのデータの参照の取得
getVArrayVal	VARRAY 型でのデータの取得
isNull	NULL 値かどうかの判定
列を削除するメソッド	
distinct	列の重複の排除
reduct	名前なし列の削除

( 15 ) DbjSetLinkInfo インターフェース ( リンクの設定情報を扱うインターフェース )

DbjSetLinkInfo インターフェースのメソッド一覧を表 1-19 に示します。

表 1-19 DbjSetLinkInfo インターフェースのメソッド一覧

メソッド	機能
リンク設定情報を取得するメソッド	
getLinkType	リンク種別の取得
getTargetObj	リンク先オブジェクトの取得
propSet	リンクオブジェクトのプロパティ値集合の取得
リンク設定情報を設定するメソッド	
setPropSet	リンクオブジェクトのプロパティ値集合の設定
setTargetObj	リンク先オブジェクトの設定

( 16 ) DbjUploadInfo インターフェース ( 文書のアップロード情報を扱うインターフェース )

DbjUploadInfo インターフェースのメソッド一覧を表 1-20 に示します。

表 1-20 DbjUploadInfo インターフェースのメソッド一覧

メソッド	機能
文書のアップロード情報を取得するメソッド	
getFilePath	登録する文書のフルパスの取得
getIndexPath	全文検索インデックス作成用ファイルのフルパスの取得
getRenditionType	登録する文書のレンディションタイプの取得
getRetrievalName	登録する文書のファイル名の取得
renditionPropSet	レンディションプロパティ値集合の取得
文書のアップロード情報を設定するメソッド	
setFilePath	登録する文書のフルパスの設定
setIndexPath	全文検索インデックス作成用ファイルのフルパスの設定
setRenditionPropSet	レンディションプロパティ値集合の設定
setRenditionType	登録する文書のレンディションタイプの設定
setRetrievalName	登録する文書のファイル名の設定

## ( 17 ) DbjVArray インターフェース ( 可変長配列オブジェクトを扱うインターフェース )

DbjVArray インターフェースのメソッド一覧を表 1-21 に示します。

表 1-21 DbjVArray インターフェースのメソッド一覧

メソッド	機能
可変長配列オブジェクトのデータを取得するメソッド	
getPropCount	メタプロパティ数の取得
getPropNameSet	メタプロパティの取得
getPropVals	可変長配列オブジェクトのプロパティ値のリストの取得
propSet	プロパティ値集合の取得
可変長配列オブジェクトのデータを設定するメソッド	
addPropSet	プロパティ値集合 ( 行 ) の追加
addPropVals	プロパティ値 ( 列 ) の追加
removeProp	プロパティの削除
可変長配列オブジェクトのプロパティ名を変更するメソッド	
changePropName	可変長配列オブジェクトのプロパティ名の変更
changePropNames	可変長配列オブジェクトのプロパティ名の一括変更

## ( 18 ) DbjXmlUploadInfo インターフェース ( XML 文書のアップロード情報を扱うインターフェース )

DbjXmlUploadInfo インターフェースのメソッド一覧を表 1-22 に示します。

表 1-22 DbjXmlUploadInfo インターフェースのメソッド一覧

メソッド	機能
XML 文書のアップロード情報を取得するメソッド	
getFilterFilePath	フィルタリング定義ファイルのフルパスの取得
getIndexType	全文検索インデクスの種別の取得
getMappingId	マッピング定義名の取得
getParseLevel	構文解析レベルの取得
XML 文書のアップロード情報を設定するメソッド	
setFilterFilePath	フィルタリング定義ファイルのフルパスの設定
setIndexType	全文検索インデクスの種別の設定
setMappingId	マッピング定義名の設定
setParseLevel	構文解析レベルの設定
XML 文書の構文解析結果メッセージを扱うメソッド	
clearParseMessage	構文解析結果メッセージのクリア
getParseMessage	構文解析結果メッセージの取得

## 1.3 文書管理クラスのインターフェース、およびメソッド一覧

文書管理クラスのインターフェース、およびメソッドを一覧形式で説明します。

### 1.3.1 文書管理クラスのインターフェース一覧

文書管理クラスのインターフェースの一覧を表 1-23 に示します。なお、それぞれのインターフェースの使用方法については、表の参照先欄に示す箇所（マニュアル「DocumentBroker Version 3 クラスライブラリ Java 解説」の章・節番号と標題）を参照してください。

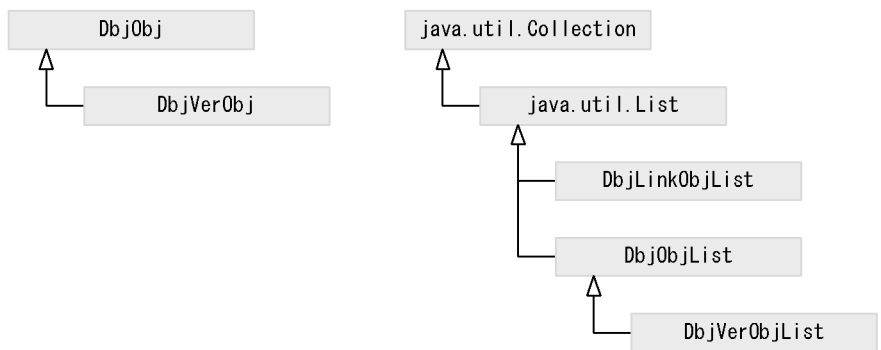
表 1-23 文書管理クラスのインターフェース一覧

インターフェース	説明	参照先
文書管理セッションを扱うインターフェース		
DbjSession インターフェース	Java クラスライブラリのセッションを管理するインターフェース	6.6 セッションとトランザクションの制御
文書空間操作を扱うインターフェース		
DbjDocSpace インターフェース	文書空間へのアクセスを扱うインターフェース	6.7 文書空間へのアクセス
文書空間オブジェクト操作を扱うインターフェース		
DbjObj インターフェース	一つの文書空間オブジェクトへのアクセスを扱うインターフェース	6.8 文書空間オブジェクトの操作
DbjVerObj インターフェース	一つの文書空間オブジェクトにアクセスするインターフェースに加えて、バージョンオブジェクトのバージョン識別子を扱うインターフェース	6.8 文書空間オブジェクトの操作
複数の文書空間オブジェクト操作を扱うインターフェース		
DbjObjList インターフェース	複数の文書空間オブジェクトに一括してアクセスするインターフェース	6.8 文書空間オブジェクトの操作
DbjVerObjList インターフェース	複数の文書空間オブジェクトに一括してアクセスするインターフェースに加えて、リストの要素すべてのバージョン識別子を扱うインターフェース	6.8 文書空間オブジェクトの操作
リンクオブジェクト操作を扱うインターフェース		
DbjLinkObj インターフェース	一つのリンクオブジェクトにアクセスするインターフェース	6.8 文書空間オブジェクトの操作
複数のリンクオブジェクト操作を扱うインターフェース		
DbjLinkObjList インターフェース	複数のリンクオブジェクトに一括してアクセスするインターフェース	6.8 文書空間オブジェクトの操作
XML 文書プロパティマッピングを扱うインターフェース		
DbjXmlTranslator インターフェース	XML 文書のプロパティマッピングを扱うインターフェース	6.10 XML 文書を管理するための操作

Java クラスライブラリで提供しているこれらのインターフェースには継承関係があり、上位にあるインターフェースで定義したメソッドは、下位のインターフェースに継承されます。インターフェースの継承関係を図 1-2 に示します。



図 1-2 文書管理クラスのインターフェースの継承関係



(凡例)

—▷ : インターフェースの継承関係を表しています。

注 次に示すインターフェースには、ほかのインターフェースとの継承関係がありません。

- DbjDocSpace
- DbjFactory
- DbjLinkObj
- DbjSession
- DbjXmlTranslator

### 1.3.2 文書管理クラスのメソッド一覧

文書管理クラスの各インターフェースのメソッドの一覧を示します。

#### (1) DbjDocSpace インターフェース (文書空間へのアクセスを扱うインターフェース)

DbjDocSpace インターフェースのメソッド一覧を表 1-24 に示します。

表 1-24 DbjDocSpace インターフェースのメソッド一覧

メソッド	機能
Proxy オブジェクトを生成するメソッド	
createLinkObjList	複数のリンクオブジェクトアクセスインターフェースの取得
createObjConnection	文書空間オブジェクトアクセスインターフェースの取得
createObjList	複数の文書空間オブジェクトアクセスインターフェースの取得
アクセス制御モードを取得するメソッド	
isAccessControlMode	アクセス制御モードの取得
オブジェクトを削除するメソッド	
removeObjects	検索条件を指定したオブジェクトの削除
検索を扱うメソッド	
changeSearchACLMode	検索実行時のアクセス制御モードの変更
executeSearch	検索の実行
getSearchACLMode	検索実行時のアクセス制御モードの取得
メタ情報を取得するメソッド	
getMeta	文書空間のメタ情報の取得

1. Java クラスライブラリのクラス、インターフェース、およびメソッド

メソッド	機能
文書空間オブジェクトを生成するメソッド	
createDocument	バージョンなし文書の作成
createFolder	バージョンなしフォルダの作成
createIndependentData	独立データの作成
createPublicACL	パブリック ACL の作成
createVrDocument	バージョン付き文書の作成
createVrFolder	バージョン付きフォルダの作成

(2) DbjLinkObj インターフェース (一つのリンクオブジェクトにアクセスするインターフェース)

DbjLinkObj インターフェースのメソッド一覧を表 1-25 に示します。

表 1-25 DbjLinkObj インターフェースのメソッド一覧

メソッド	機能
構成管理型リンクを変更するメソッド	
changeToVTFix	構成管理型リンクの構成管理モードを FIX モードに変更
changeToVTFloat	構成管理型リンクの構成管理モードを FLOATING モードに変更
リンク Proxy オブジェクトのプロパティを取得するメソッド	
getLinkId	リンク識別子の取得
getLinkType	リンク種別の取得
getOwnerObj	リンク元オブジェクトの取得
getTargetObj	リンク先オブジェクトの取得
リンク Proxy オブジェクトのプロパティ値集合を取得または設定するメソッド	
propSet	リンク Proxy オブジェクトのプロパティ値集合インターフェースの取得
setPropSet	リンク Proxy オブジェクトのプロパティ値集合の設定
リンクオブジェクトのプロパティを取得または設定するメソッド	
readProperties	リンクプロパティの取得
writeProperties	リンクプロパティ値の設定
リンクオブジェクトを削除するメソッド	
removeObject	リンクオブジェクトの削除

(3) DbjLinkObjList インターフェース (DbjLinkObj インターフェースを要素とするリストインターフェースを扱うインターフェース)

DbjLinkObjList インターフェースのメソッド一覧を表 1-26 に示します。

表 1-26 DbjLinkObjList インターフェースのメソッド一覧

メソッド	機能
構成管理型リンクを一括して変更するメソッド	

メソッド	機能
changeToVTFix	構成管理型リンクの構成管理モードを一括して FIX モードに変更
changeToVTFloat	構成管理型リンクの構成管理モードを一括して FLOATING モードに変更
リンク Proxy オブジェクトのプロパティを一括して取得するメソッド	
getLinkIdList	ターゲットリンク識別子プロパティのリストの取得
getLinkObj	リスト要素の DbjLinkObj インターフェースの取得
getOwnerObjList	リンク元オブジェクトのリストの取得
getTargetObjList	リンク先オブジェクトのリストの取得
リンクオブジェクトのプロパティを一括して取得または設定するメソッド	
readProperties	リンクプロパティ値の一括取得
writeProperties	リンクプロパティ値の一括設定
リンクオブジェクトを一括して削除するメソッド	
removeObjects	リンクオブジェクトの一括削除

#### (4) DbjObj インターフェース (一つの文書空間オブジェクトのアクセスを扱うインターフェース)

DbjObj インターフェースのメソッド一覧を表 1-27 に示します。

表 1-27 DbjObj インターフェースのメソッド一覧

メソッド	機能
アクセス対象のターゲットバージョンを変更するメソッド	
getTargetVersion	アクセス対象ターゲットバージョンのバージョン識別子の取得
setTargetVersion	アクセス対象ターゲットバージョンのバージョン識別子の変更
文書空間オブジェクトのプロパティを取得するメソッド	
getClassName	アクセス対象文書空間オブジェクトを構成する DMA クラス名の取得
getLockType	Proxy オブジェクトのアクセスロック種別の取得
getObjType	アクセス対象文書空間オブジェクトのオブジェクト種別の取得
getOiid	アクセス対象文書空間オブジェクトの OIID の取得
コンテンツにアクセスするメソッド	
downloadContents	文書のコンテンツのダウンロード
uploadContents	文書のコンテンツのアップロード
バージョンを扱うメソッド	
cancelCheckOut	バージョン付きオブジェクトのチェックアウトの取り消し
checkIn	バージョン付きオブジェクトのチェックイン
checkOut	バージョン付きオブジェクトのチェックアウト
deleteVersion	バージョンの削除
getCheckOutStatus	バージョン付きオブジェクトのチェックアウト状態の取得

1. Java クラスライブラリのクラス、インターフェース、およびメソッド

メソッド	機能
getVersionId	バージョンオブジェクトのバージョン識別子の取得
getVersioningInfo	バージョン付きオブジェクトのバージョンングオブジェクトの取得
getVersionObjList	バージョン付きオブジェクトのバージョン一覧の取得
パブリック ACL を扱うメソッド	
bindPublicACL	パブリック ACL のバインド
getBindObjectList	パブリック ACL にバインドしている文書空間オブジェクト一覧の取得
getPublicACLList	バインドしているパブリック ACL 一覧の取得
unbindPublicACL	パブリック ACL のアンバインド
プロパティ 値集合を取得または設定するメソッド	
propSet	Proxy オブジェクトのプロパティ値集合インターフェースの取得
setPropSet	Proxy オブジェクトのプロパティ値集合の設定
プロパティ 値を取得または設定するメソッド	
readProperties	文書空間オブジェクトのプロパティ 値の読み込み
writeProperties	文書空間オブジェクトのプロパティ 値の設定
文書空間オブジェクトを削除するメソッド	
removeObject	文書空間オブジェクトの削除
リンクをたどって文書空間オブジェクトを取得するメソッド	
getChildList	フォルダのリンク先オブジェクト (下位オブジェクト) の取得
getDCRParent	直接型リンクによるリンク元オブジェクトの取得
getParentList	フォルダのリンク元オブジェクトの取得
getRelList	文書間リンク一覧の取得
リンクおよびリンクを解除するメソッド	
changeToVTFix	構成管理型リンクの構成管理モードを FIX モードに変更
changeToVTFloat	構成管理型リンクの構成管理モードを FLOATING モードに変更
link	リンク先オブジェクトとのリンク
move	直接型リンクが設定されている文書空間オブジェクトの移動
unlink	リンク先オブジェクトとのリンクの解除
unlinkByLinkId	リンク識別子の指定によるリンク先オブジェクトとのリンクの解除
レンディションを扱うメソッド	
addRendition	レンディションの追加
changeMasterRendition	マスタレンディションの変更
deleteRendition	レンディションの削除
getRenditionList	レンディション情報一覧の取得
writeRenditionProperties	レンディションプロパティの設定
アクセスロックが異なる文書空間オブジェクトインターフェースを取得するメソッド	

メソッド	機能
lock	アクセスロック種別の異なる文書空間オブジェクトインターフェースの取得

#### (5) DbjObjList インターフェース (複数の文書空間オブジェクトに一括してアクセスするためのインターフェース)

DbjObjList インターフェースのメソッド一覧を表 1-28 に示します。

表 1-28 DbjObjList インターフェースのメソッド一覧

メソッド	機能
オブジェクトを一括して移動するメソッド	
move	直接型リンクが設定されている文書空間オブジェクトの一括移動
プロパティ値集合を一括して設定するメソッド	
setPropSet	要素のプロパティ値集合の設定
プロパティ値を一括して取得または設定するメソッド	
readProperties	複数の文書空間オブジェクトプロパティ値の一括取得
writeProperties	複数の文書空間オブジェクトプロパティ値の一括設定
文書空間オブジェクトを一括して削除するメソッド	
removeObjects	文書空間オブジェクトの一括削除
要素を一括して取得するメソッド	
getObj	リスト要素の取得
ロックを一括して指定するメソッド	
lock	アクセスロック種別の異なる複数文書空間オブジェクトインターフェースの取得

#### (6) DbjSession インターフェース (Java クラスライブラリのセッションを管理するインターフェース)

DbjSession インターフェースのメソッド一覧を表 1-29 に示します。

表 1-29 DbjSession インターフェースのメソッド一覧

メソッド	機能
セッションを扱うメソッド	
checkSession	セッションが有効かどうかのチェック
getLoginUserInfo	ユーザ情報の取得
login	ログイン
logout	ログアウト
トランザクションを扱うメソッド	
begin	トランザクションの開始
commit	トランザクションの確定
rollback	トランザクションの取り消し
コンテンツ格納先ベースパスを扱うメソッド	
getReferencePath	コンテンツ格納先ベースパスの取得

1. Java クラスライブラリのクラス、インターフェース、およびメソッド

メソッド	機能
setReferencePath	コンテンツ格納先ベースパスの設定

(7) DbjVerObj インターフェース (一つの文書空間オブジェクトにアクセスするインターフェースに加え、バージョンオブジェクトのバージョン識別子を扱うインターフェース)

DbjVerObj インターフェースのメソッド一覧を表 1-30 に示します。

表 1-30 DbjVerObj インターフェースのメソッド一覧

メソッド	機能
バージョンオブジェクトのバージョン識別子を取得するメソッド	
getVersionId	バージョンオブジェクトのバージョン識別子の取得

(8) DbjVerObjList インターフェース (DbjVerObj インターフェースを要素とするリストインターフェースを扱うインターフェース)

DbjVerObjList インターフェースのメソッド一覧を表 1-31 に示します。

表 1-31 DbjVerObjList インターフェースのメソッド一覧

メソッド	機能
要素のバージョンオブジェクトのバージョン識別子を一括して取得するメソッド	
getVerObj	要素の DbjVerObj インターフェースの取得
getVersionIdList	オブジェクトのバージョン識別子リストの取得

(9) DbjXmlTranslator インターフェース (XML 文書のプロパティマッピングを扱うインターフェース)

DbjXmlTranslator インターフェースのメソッド一覧を表 1-32 に示します。

表 1-32 DbjXmlTranslator インターフェースのメソッド一覧

メソッド	機能
getMappedProperties	XML 文書の XML プロパティマッピング結果情報の作成

## 1.4 メタクラスのインターフェース、およびメソッド一覧

メタクラスのインターフェース、およびメソッドを一覧形式で説明します。

### 1.4.1 メタクラスのインターフェース一覧

メタクラスのインターフェースの一覧を表 1-33 に示します。なお、それぞれのインターフェースの使用方法については、マニュアル「DocumentBroker Version 3 クラスライブラリ Java 解説」の「6.11 メタ情報の取得」を参照してください。

表 1-33 メタクラスのインターフェース一覧

インターフェース	説明
DbjClassDesc インターフェース	一つのクラスのクラスディスクリプションを扱うインターフェース
DbjMeta インターフェース	一つの文書空間のメタ情報を扱うインターフェース
DbjMetaManager インターフェース	文書空間のメタ情報を管理するインターフェース
DbjPropDesc インターフェース	一つのプロパティのプロパティディスクリプションを扱うインターフェース

なお、Java クラスライブラリで提供しているメタクラスのインターフェースには、継承関係がありません。

### 1.4.2 メタクラスのメソッド一覧

メタクラスの各インターフェースでのメソッドの一覧を示します。

#### (1) DbjClassDesc インターフェース (クラスディスクリプションを扱うインターフェース)

DbjClassDesc インターフェースのメソッド一覧を表 1-34 に示します。

表 1-34 DbjClassDesc インターフェースのメソッド一覧

メソッド	機能
getName	クラス名の取得
getProperties	プロパティディスクリプションの取得
getSubClasses	サブクラスのクラスディスクリプションの取得
getSuperClass	スーパークラスのクラスディスクリプションの取得

#### (2) DbjMeta インターフェース (文書空間のメタ情報を扱うインターフェース)

DbjMeta インターフェースのメソッド一覧を表 1-35 に示します。

表 1-35 DbjMeta インターフェースのメソッド一覧

メソッド	機能
getClassDesc	指定したクラスのクラスディスクリプションの取得
getDocSpaceId	文書空間識別子の取得

## 1. Java クラスライブラリのクラス、インターフェース、およびメソッド

メソッド	機能
getExtFromRenditionType	レンディションタイプに対応する拡張子の取得
getPropDataType	指定したプロパティのデータ型の取得
getPropDesc	指定したプロパティのプロパティディスクリプションの取得
getRenditionType	拡張子に対応するレンディションタイプの取得

### (3) DbjMetaManager インターフェース ( 文書空間のメタ情報を管理するインターフェース )

DbjMetaManager インターフェースのメソッド一覧を表 1-36 に示します。

表 1-36 DbjMetaManager インターフェースのメソッド一覧

メソッド	機能
getMeta	メタ情報の取得

### (4) DbjPropDesc インターフェース ( プロパティディスクリプションを扱うインターフェース )

DbjPropDesc インターフェースのメソッド一覧を表 1-37 に示します。

表 1-37 DbjPropDesc インターフェースのメソッド一覧

メソッド	機能
getDataType	プロパティのデータ型の取得
getName	プロパティ名の取得
getVArrayClass	VARRAY 型プロパティを扱うクラスのクラスディスクリプションの取得



## 1.5 例外クラスのクラス一覧，スーパークラス，およびコンストラクタ

例外クラスのクラスの一覧，スーパークラス，およびコンストラクタについて説明します。

### 1.5.1 例外クラスのクラス一覧

例外クラスのクラス一覧を表 1-38 に示します。なお，例外処理の方法については，マニュアル「DocumentBroker Version 3 クラスライブラリ Java 解説」の「6.12 例外処理」を参照してください。

表 1-38 例外クラスのクラス一覧

クラス	説明
DbjAccessControlException クラス	ユーザがメソッドを実行する際にその操作に対してアクセス権がなかった場合のエラーを表す例外クラス
DbjAccessControlNotSupportedException クラス	文書空間がアクセス制御機能対応していない場合のエラーを表す例外クラス
DbjACEOperationException クラス	ACE の操作エラーを表すスーパークラス。特定の ACE 操作のエラーはサブクラスがスローされる例外クラス
DbjACLOutOfRangeException クラス	文書空間オブジェクトの ACE の個数が制限値を超えた場合のエラーを表す例外クラス
DbjAlreadyCheckOutException クラス	すでにチェックアウトされている場合のエラーを表す例外クラス
DbjCheckOutException クラス	チェックアウトについてのエラーを表す例外クラスのスーパークラス
DbjContentNotRegisteredException クラス	リファレンスファイル管理機能を使用したオブジェクトをローカルファイルにダウンロードする場合に，オブジェクトにコンテンツが登録されていないときのエラーを表す例外クラス
DbjContentTypeMismatchException クラス	リファレンスファイル管理機能で，オブジェクトのコンテンツ種別とメソッドが要求するコンテンツ種別が一致しない場合のエラーを表す例外クラス
DbjCORBAException クラス	CORBA エラー（TPBroker でのエラー）を表す例外クラス
DbjDBDeadLockException クラス	DB のデッドロックエラーを表す例外クラス
DbjDBException クラス	DB エラーを表す例外クラス
DbjDBLockTimeoutException クラス	DB のロックタイムアウトエラーを表す例外クラス
DbjDisconnectedSessionException クラス	文書空間との接続が切断された場合のエラーを表す例外クラス
DbjError クラス	java.lang.Error クラスを継承し，Java クラスライブラリ固有の致命的エラーを表す例外クラスのスーパークラス
DbjException クラス	java.lang.Exception クラスを継承し，Java クラスライブラリ固有のエラーを表す例外クラスのスーパークラス
DbjFileAccessException クラス	ファイルのアクセス権がない場合のエラーを表す例外クラス

1. Java クラスライブラリのクラス、インターフェース、およびメソッド

クラス	説明
DbjFileNotFoundException クラス	指定ファイルが存在しない場合のエラーを表す例外クラス
DbjFileReferenceCurrentContentNotfoundException クラス	リファレンスファイル管理機能を使用したバージョン付き文書オブジェクトで、カレントバージョンにコンテンツが存在しない場合のエラーを表す例外クラス
DbjFileReferenceMismatchStatusException クラス	リファレンスファイル管理機能を使用するサーバでのコンテンツ操作でエラーが発生し、オブジェクトとコンテンツが不整合な状態になった場合のエラーを表す例外クラス
DbjFileReferenceOperationFailedException クラス	リファレンスファイル管理機能を使用するサーバでのコンテンツ操作が失敗した場合のエラーを表す例外クラス
DbjIllegalDocSpaceIdException クラス	不正な文書空間識別子が指定された場合のエラーを表す例外クラス
DbjIllegalObjectTypeException クラス	メソッドによる操作に不適切な文書空間オブジェクトの種別が指定された場合のエラーを表す例外クラス
DbjIllegalPropValException クラス	指定プロパティ値の不正エラーを表す例外クラス
DbjInitializeError クラス	Java クラスライブラリ初期化時のエラーを表す例外クラス
DbjInternalError クラス	Java クラスライブラリの内部エラーを表す例外クラス
DbjIOException クラス	Java クラスライブラリの IO エラーを表す例外クラスのスーパークラス
DbjIsMasterRenditionException クラス	マスタレンディションが指定された場合のエラーを表す例外クラス
DbjLastVersionException クラス	操作対象となるバージョン付きオブジェクトの最後の一つのバージョンを削除しようとした場合のエラーを表す例外クラス
DbjMasterRenditionNotSetException クラス	マスタレンディションのレンディションタイプが設定されていない場合のエラーを表す例外クラス
DbjNotAuthenticatedException クラス	ユーザ認証エラーを表す例外クラス
DbjNotCheckOutException クラス	チェックアウトされていない場合のエラーを表す例外クラス
DbjNotLoginException クラス	文書空間と接続 (ログイン) されていない場合のエラーを表す例外クラス
DbjObjectNotFoundException クラス	操作対象となる文書空間オブジェクトが、メソッド実行時にすでに削除されていた場合、または存在しない場合のエラーを表す例外クラス
DbjOutOfMemoryError クラス	メモリ不足が発生したことを表す例外クラス
DbjPublicACLAlreadyBoundException クラス	指定したパブリック ACL がすでにバインドされている場合のエラーを表す例外クラス
DbjPublicACLNotBoundException クラス	指定したパブリック ACL がバインドされていない場合のエラーを表す例外クラス
DbjPublicACLNotFoundException クラス	指定したパブリック ACL が存在しない場合のエラーを表す例外クラス
DbjPublicACLOperationException クラス	パブリック ACL の操作エラーを表すスーパークラス。特定のパブリック ACL 操作のエラーにはサブクラスがスローされる例外クラス

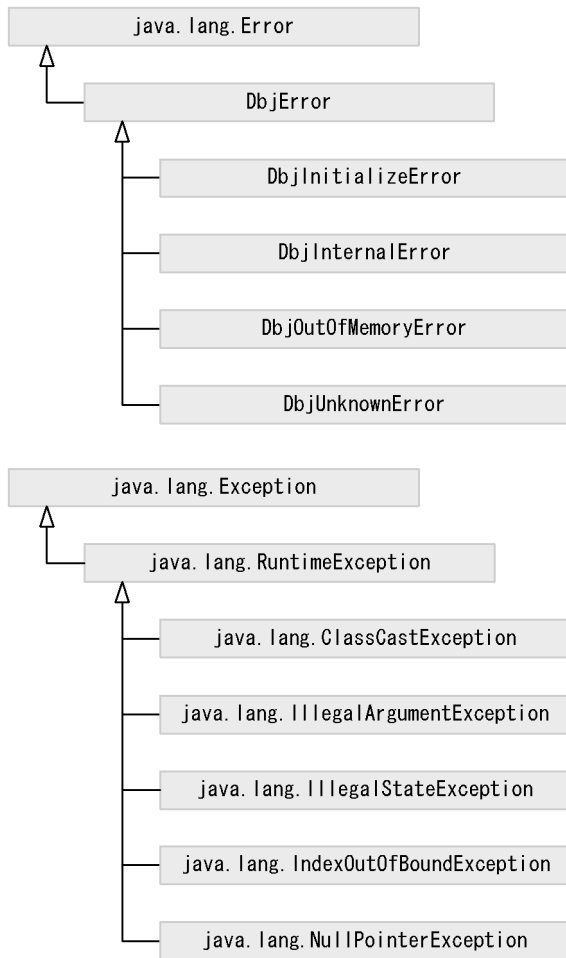
1. Java クラスライブラリのクラス、インターフェース、およびメソッド

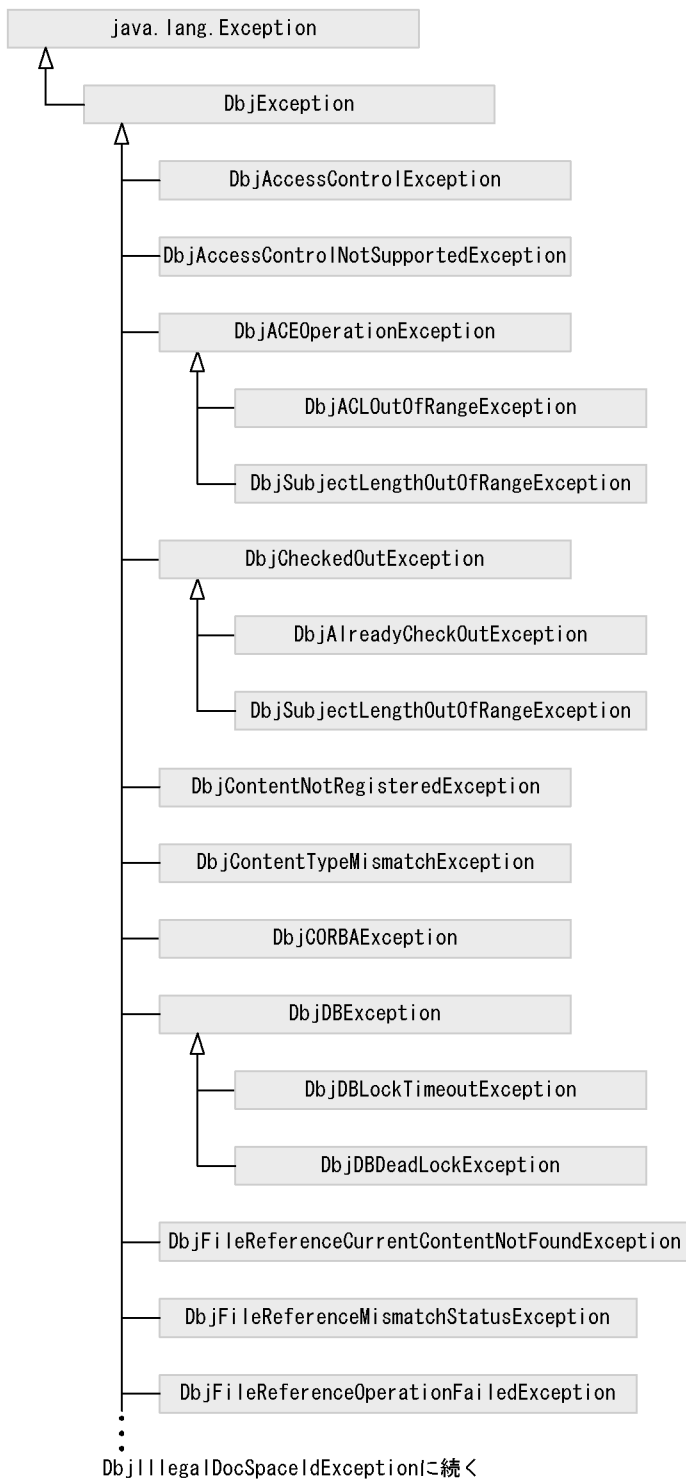
クラス	説明
DbjPublicACLOutOfRangeException クラス	オブジェクトに追加できるパブリック ACL の数が制限値を超えた場合のエラーを表す例外クラス
DbjReferenceTypeMismatchException クラス	リファレンスファイル管理機能で、オブジェクトのリファレンス種別とメソッドが要求するリファレンス種別が一致しない場合のエラーを表す例外クラス
DbjRenditionCountOutOfRangeException クラス	レンディション数が制限値を超えた場合のエラーを表す例外クラス
DbjRenditionIsEmptyException クラス	レンディションが空の場合のエラーを表す例外クラス
DbjRenditionNotConvertedException クラス	レンディションが更新処理されていない場合のエラーを表す例外クラス
DbjRenditionNotFoundException クラス	指定レンディションが存在しない場合のエラーを表す例外クラス
DbjRenditionTypeDuplicatedException クラス	レンディションタイプが重複している場合のエラーを表す例外クラス
DbjSessionException クラス	セッションに関するエラーを表す例外クラスのスーパークラス
DbjSessionNotConnectException クラス	文書空間との接続が行われていない場合のエラーを表す例外クラスのスーパークラス
DbjSessionOverflowException クラス	文書空間同時接続数が制限値を超えた場合のエラーを表す例外クラス
DbjSubjectLengthOutOfRangeException クラス	ACE のサブジェクト長が制限値を超えた場合のエラーを表す例外クラス
DbjTargetContentPathNotSetException クラス	リファレンスファイル管理機能で、コンテンツ格納先ベースパスが設定されていない場合のエラーを表す例外クラス
DbjUnknownError クラス	Java クラスライブラリ内での未知のエラーの発生を表す例外クラス
DbjXmlParseException クラス	XML 文書の構文解析エラーを表す例外クラス

Java クラスライブラリで提供しているこれらのクラスには継承関係があり、上位にあるクラスで定義したメソッドは、下位のクラスに継承されます。クラスの継承関係を図 1-3 に示します。

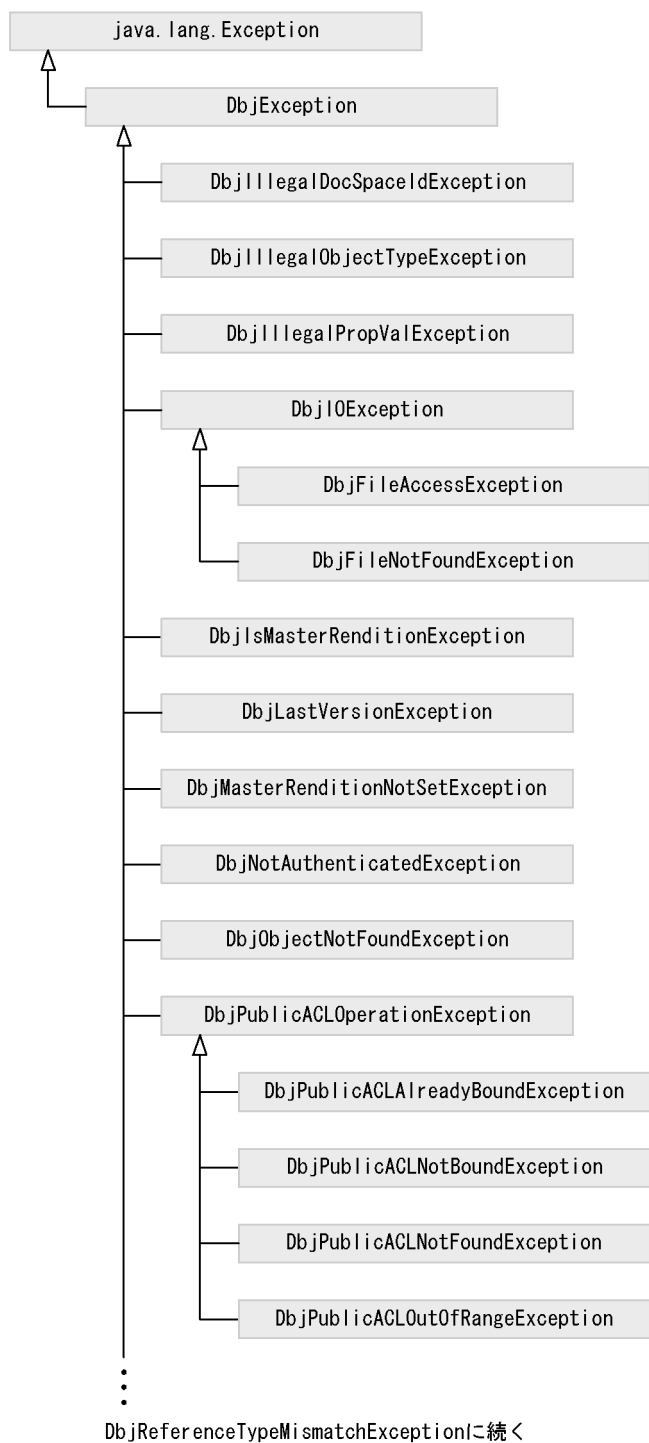
1. Java クラスライブラリのクラス、インターフェース、およびメソッド

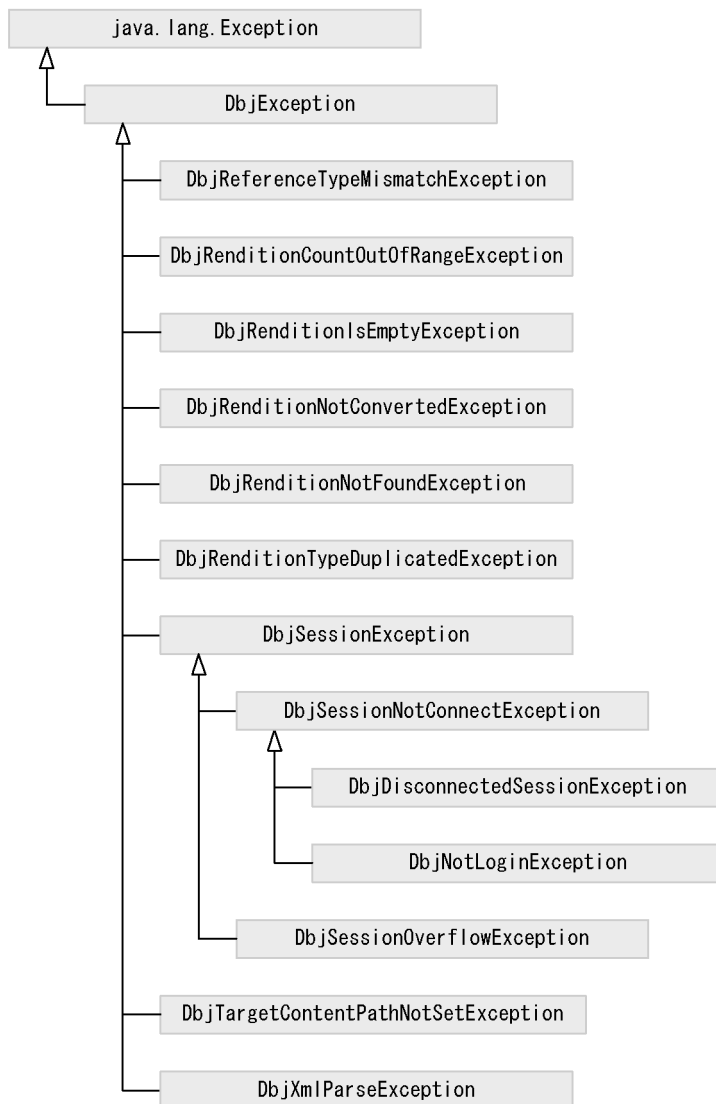
図 1-3 例外クラスのインターフェースの継承関係





1. Java クラスライブラリのクラス、インターフェース、およびメソッド





(凡例)

—▷ : インターフェースの継承関係を表しています。

## 1.5.2 例外クラスのスーパークラス

Java クラスライブラリの例外クラスのスーパークラスについて次に説明します。

### (1) DbjError クラス

DbjError クラスは `java.lang.Error` クラスを継承し、Java クラスライブラリ固有の致命的なエラーを表すスーパークラスです。メモリ不足、Java クラスライブラリの初期化エラー、Java クラスライブラリ内で予期しない内部エラーが発生した時のエラーなど、ユーザアプリケーションプログラムで対処できないエラーを表します。

### (2) DbjException クラス

DbjException クラスは `java.lang.Exception` クラスを継承し、Java クラスライブラリ固有のエラーを表す

スーパークラスです。

### 1.5.3 例外クラスのコンストラクタ

DbjError クラスおよび DbjException クラスは、二つの public 属性のコンストラクタを保持します。一つは、引数を取らないコンストラクタで、もう一つは、String 型の引数を取るコンストラクタです。引数を取らないコンストラクタは、エラーメッセージに null が関連付けられた例外クラスのオブジェクトを生成します。また、String 型の引数を取るコンストラクタは、引数に指定されたエラーメッセージが関連付けられた例外クラスのオブジェクトを生成します。



## 1.6 定数定義クラスのカテゴリー一覧

定数定義クラスのカテゴリーを表 1-39 に示します。

表 1-39 定数定義クラスのカテゴリー一覧

カテゴリ	説明
DbjDef クラス	
ACL	検索実行時のアクセス制御モードを表す定数
DATATYPE	プロパティのデータ型を表す定数
DMA BOOLEAN	BOOL 型を表す定数
INDEXTYPE	全文検索インデックスの種別を表す定数
LINK	リンク種別を表す定数
LOCK	ロック種別を表す定数
OBJTYPE	オブジェクト種別を表す定数
OPERATEMODE	コンテンツのパス操作モードを表す定数
ORDER	バージョン情報の取得順序を表す定数
PERM	パーミッションを表す定数
PRIV	ユーザの特権を表す定数
REFERENCETYPE	リファレンス種別を表す定数
RELATIONEND	文書間リンク設定情報一覧の取得条件を表す定数
RENDSTATUS	レンディションステータスを表す定数
SUBJECTTYPE	サブジェクト種別を表す定数
SYSSUBJECT	システムサブジェクトを表す定数
XMLPARSE	XML 文書の構文解析レベルを表す定数
Others	そのほか（上記以外）の内容を表す定数
DbjTraceDef クラス	
TRACELEVEL	トレースの出力レベルを表す定数
TRACEOUTPUT	トレースの出力先を表す定数

## 1.7 ライブラリ情報取得クラスのクラス、およびメソッド一覧

ライブラリ情報取得クラスのクラス、およびメソッドを一覧形式で説明します。

### 1.7.1 ライブラリ情報取得クラスのクラス

ライブラリ情報取得クラスのクラスを表 1-40 に示します。なお、クラスの使用方法については、マニュアル「DocumentBroker Version 3 クラスライブラリ Java 解説」の「6.13 ライブラリ情報の取得」を参照してください。

表 1-40 ライブラリ情報取得クラスのクラス

クラス	説明
ライブラリ情報を扱うクラス	
DbjLibInfo クラス	DocumentBroker Development Kit および DocumentBroker Runtime のバージョン情報を扱うクラス

### 1.7.2 ライブラリ情報取得クラスのメソッド一覧

ライブラリ情報取得クラスの DbjLibInfo クラス ( DocumentBroker Development Kit および DocumentBroker Runtime のバージョン情報を扱うクラス ) のメソッドを表 1-41 に示します。

表 1-41 DbjLibInfo クラスのメソッド

メソッド	機能
バージョンを取得するメソッド	
getVersion	DocumentBroker Development Kit および DocumentBroker Runtime のバージョンの取得

## 1.8 トレースクラスのクラス、およびメソッド一覧

トレースクラスのクラス、およびメソッドを一覧形式で説明します。

### 1.8.1 トレースクラスのクラス

トレースクラスのクラスを表 1-42 に示します。なお、クラスの使用方法については、マニュアル「DocumentBroker Version 3 クラスライブラリ Java 解説」の「6.14 ユーザアプリケーションプログラムのトレース情報の出力」を参照してください。

表 1-42 トレースクラスのクラス

クラス	説明
トレース情報を扱うクラス	
DbjTrace クラス	トレース情報を扱うクラス

### 1.8.2 トレースクラスのメソッド一覧

トレースクラスの DbjTrace クラス (トレース情報を扱うクラス) のメソッドを表 1-43 に示します。

表 1-43 DbjTrace クラスのメソッド

メソッド	機能
トレース情報を出力するメソッド	
arg	パラメタ情報の出力
call	外部 API の呼び出し情報の出力
DbjTrace	コンストラクタ
enter	メソッドの入口情報の出力
error	エラー情報の出力
exit	メソッドの出口情報の出力
hint	下位のメソッドのエラー情報の出力
init	トレースクラスの初期化
msg	メッセージの出力
returned	外部 API からのリターン情報の出力

## 1.9 クラス、インターフェースおよびメソッドの説明形式

---

Java クラスライブラリのクラス、インターフェースおよびメソッドについての説明形式について示します。各クラスまたは各インターフェースの説明では、そのクラスまたはインターフェースに定義されるメソッドについてだけ説明しています。上位のクラスまたはインターフェースで定義されているメソッドについては、定義元のクラスまたはインターフェースを参照してください。

### 1.9.1 メソッドで説明する項目

クラスおよびインターフェースの説明に続けて、そのクラスおよびインターフェースのメソッドを説明しています。上位のインターフェースから継承されるメソッドの詳細については、継承元のインターフェースのメソッドの説明を参照してください。

#### (1) 機能

メソッドの機能の詳細を説明しています。

#### (2) 形式

メソッドの形式を説明しています。複数の形式がある場合、「(a) 形式 1」、「(b) 形式 2」、「(c) 形式 3」などと表記します。

#### (3) 引数

引数について次のように説明しています。

引数（入力）

指定する値について説明しています。

引数（入出力）

指定した領域に設定される値について説明しています。

- 引数は「形式」で記述した文字列です。
- 省略できることを明記していない引数は、すべて指定する必要があります。
- (入力) は、値を指定することを示しています。
- (入出力) は、メソッドによって出力値が設定されるオブジェクトを指定することを示しています。

#### (4) 戻り値

メソッドの戻り値を説明しています。戻り値を返却しないメソッドでは、「なし」と記述します。

#### (5) 例外

発生する例外について説明しています。

### 1.9.2 パラメタクラスのインターフェースで説明する項目

パラメタクラスのインターフェースの説明では、次の項目について説明しています。

#### (1) 直接のスーパーインターフェース

該当するインターフェースの、直接のスーパーインターフェース名を記述しています。

なお、パラメタクラスのインターフェースは、`java.util.Collection` インターフェース、`java.util.List` イン

ターフェース、java.util.Map インターフェース、および java.lang.Cloneable インターフェースなどを継承しています。

## (2) プロパティ一覧

クラスおよびインターフェースで定義されているプロパティ一覧を記述しています。プロパティ一覧の例を表 1-44 に示します。

表 1-44 プロパティ一覧の例

プロパティ名	データ型	getter メソッド	setter メソッド
aaa	xxx 型	getAaa	setAaa
bbb	yyy 型	getBbb	setBbb

### プロパティ名

プロパティの名称を記述しています。

### データ型

プロパティの値のデータ型を記述しています。インターフェースを扱う場合は、インターフェース名を記述しています。

### getter メソッド

プロパティを取得するメソッド名を記述しています。

### setter メソッド

プロパティを設定するメソッド名を記述しています。

## 1.9.3 文書空間オブジェクトのプロパティのデータ型

メソッドの説明で使用する文書空間オブジェクトのプロパティのデータ型と、DocumentBroker サーバで扱うデータ型の対応を表 1-45 に示します。

表 1-45 文書空間オブジェクトのプロパティのデータ型と DocumentBroker サーバで扱うデータ型の対応

プロパティのデータ型	DocumentBroker サーバで扱うデータ型
BOOL	Boolean 型
INT	Integer32 型
VARRAY	Object 型
STR	String 型
STRLIST	-

### (凡例)

- : 該当するデータ型がないことを示します。

DocumentBroker サーバで扱うデータ型については、マニュアル「DocumentBroker Version 3 システム導入・運用ガイド」を参照してください。



# 2

## ファクトリクラス詳細

この章では、ファクトリクラスのインターフェース、およびメソッドについて説明します。

---

2.1 DbjFactory0200 クラス

---

2.2 DbjFactory インターフェース

---

## 2.1 DbjFactory0200 クラス

---

DbjFactory0200 クラスは、Java クラスライブラリのファクトリインターフェース、およびメタマネージャインターフェースを取得するクラスです。なお、このクラスはインスタンスを生成できません（コンストラクタは、`private` 属性です）。

Java クラスライブラリは、このクラスのメソッドが実行される初回時に初期化されます。初期化によって、Java クラスライブラリは動作環境の定義に従った内容で設定されます。ただし、初期化の実行時にエラーがあると、例外クラス `DbjInitializeError` がスローされるおそれがあります。

以降、DbjFactory0200 クラスのメソッドについて説明します。



## 2.1.1 getFactory (ファクトリインターフェースの取得)

### (1) 機能

ファクトリインターフェースを取得します。

### (2) 形式

```
static DbjFactory getFactory()
```

### (3) 引数

なし

### (4) 戻り値

ファクトリインターフェース (DbjFactory インターフェース)

### (5) 例外

なし

## 2.1.2 getMetaManager (メタマネージャーインターフェースの取得)

### (1) 機能

メタマネージャーインターフェースを取得します。

### (2) 形式

```
static DbjMetaManager getMetaManager()
```

### (3) 引数

なし

### (4) 戻り値

メタマネージャーインターフェース (DbjMetaManager インターフェース)

### (5) 例外

なし

## 2.2 DbjFactory インターフェース

DbjFactory インターフェースは、Java クラスライブラリのオブジェクトを生成して、そのインターフェースを取得する機能を持つファクトリインターフェースです。DbjFactory インターフェースは、表 2-1 に示すオブジェクトを作成してインターフェースを取得するメソッドを提供します。ユーザアプリケーションプログラムは、DbjFactory インターフェースを使用して Java クラスライブラリのセッションオブジェクトを生成して、そのインターフェースを取得したり、パラメタクラスのオブジェクトを生成して、そのインターフェースを取得したりできます。

DbjFactory インターフェースで作成できるオブジェクト、そのオブジェクトを作成するメソッド、および取得するインターフェース名を表 2-1 に示します。

表 2-1 DbjFactory インターフェースで作成できるオブジェクト、オブジェクトを作成するメソッド、および取得するインターフェース

オブジェクト	オブジェクトを作成するメソッド	取得するインターフェース
?パラメタ (BOOL 型) オブジェクト	createBooleanQParam	DbjBooleanQParam
?パラメタ (INT 型) オブジェクト	createInteger32QParam	DbjInteger32QParam
?パラメタ (Object 型) オブジェクト	createObjQParam	DbjObjQParam
?パラメタ (OID) オブジェクト	createOIDQParam	DbjOIDQParam
?パラメタ (String 型) オブジェクト	createStringQParam	DbjStringQParam
?パラメタ (概念検索種文章) オブジェクト	createSeedDocQParam	DbjSeedDocQParam
ACE オブジェクト	createACE	DbjACE
XML トランスレーターオブジェクト	createXmlTranslator	DbjXmlTranslator
XML 文書アップロード情報オブジェクト	createXmlUploadInfo	DbjXmlUploadInfo
アップロード情報オブジェクト	createUploadInfo	DbjUploadInfo
可変長配列オブジェクト	createVArray	DbjVArray
Java クラスライブラリのセッションオブジェクト	createSession	DbjSession
検索結果取得情報オブジェクト	createFetchInfo	DbjFetchInfo
構成管理モードが FIX モードの構成管理型リンク設定情報オブジェクト	createSetVTFixLinkInfo	DbjSetVTFixLinkInfo
構成管理モードが FLOATING モードの構成管理型リンク設定情報オブジェクト	createSetVTFloatLinkInfo	DbjSetVTFloatLinkInfo
参照型リンク設定情報オブジェクト	createSetRCRLinkInfo	DbjSetRCRLinkInfo
直接型リンク設定情報オブジェクト	createSetDCRLinkInfo	DbjSetDCRLinkInfo
パブリック ACL の OID エレメントオブジェクト	createPublicACLIdElm	DbjPublicACLIdElm

## 2. ファクトリクラス詳細

オブジェクト	オブジェクトを作成するメソッド	取得するインターフェース
プロパティ値集合オブジェクト	createPropSet	DbjPropSet
文書間リンク設定情報オブジェクト	createSetRelLinkInfo	DbjSetRelLinkInfo
リファレンスファイル文書のアップロード情報オブジェクト	createReferenceUploadInfo	DbjReferenceUploadInfo
リファレンスファイル文書のパス情報オブジェクト	createReferencePathInfo	DbjReferencePathInfo

以降、DbjFactory インターフェースのメソッドについて説明します。

## 2.2.1 createACE ( ACE の作成 )

### (1) 機能

ACE を作成し、そのインターフェース ( DbjACE インターフェース ) を取得します。

形式 1 では、初期値状態の ACE を作成します。形式 2 では、指定されたプロパティ値集合を ACE とみなして作成して propSet プロパティに設定します。この場合、値はコピーされません。形式 3 では、指定された引数を初期値として ACE を作成します。なお、形式 2 および形式 3 の引数に指定するプロパティ値集合の妥当性は、検証されません。

### (2) 形式

#### (a) 形式 1

```
DbjACE createACE ()
```

#### (b) 形式 2

```
DbjACE createACE (
    DbjPropSet src
)
```

#### (c) 形式 3

```
DbjACE createACE (
    String      subject,
    int         subjectType,
    int         permission
)
```

### (3) 引数

src ( 入力 )

ACE として扱うプロパティ値集合を指定します。null を指定すると例外がスローされます。

subject ( 入力 )

ACE のサブジェクトを指定します。

subjectType ( 入力 )

ACE のサブジェクト種別を指定します。

permission ( 入力 )

ACE のパーミッションを指定します。

### (4) 戻り値

ACE インターフェース ( DbjACE インターフェース )

### (5) 例外

NullPointerException

引数 src に null を指定した場合

## 2.2.2 createBooleanQParam ( BOOL 型の ? パラメタの作成 )

### (1) 機能

BOOL 型の ? パラメタを作成し、そのインターフェース ( DbjBooleanQParam インターフェース ) を取得します。なお、引数に指定する値の妥当性は検証されません。

### (2) 形式

```
DbjBooleanQParam createBooleanQParam(  
    int      param  
)
```

### (3) 引数

param ( 入力 )

パラメタの値を指定します。次に示す値のどれかを指定できます。

- DbjDef.DMA\_TRUE
- DbjDef.DMA\_FALSE
- DbjDef.DMA\_UNKNOWN

### (4) 戻り値

BOOL 型の ? パラメタインターフェース ( DbjBooleanQParam インターフェース )

### (5) 例外

なし

## 2.2.3 createFetchInfo ( 検索結果取得情報オブジェクトの作成 )

### (1) 機能

検索結果取得情報オブジェクトを作成し、そのインターフェース ( DbjFetchInfo インターフェース ) を取得します。

形式 1 では、オブジェクトのメンバにデフォルト値が入ります。形式 2 では、初期値を引数で指定できます。形式 3 では、形式 2 で指定できる初期値に加えて、キャッシュ検索実行時に取得した結果のソートを制御するための引数 `comp` を指定できます。オブジェクトの初期値として設定できるパラメタの詳細については、「3.5 DbjFetchInfo インターフェース」を参照してください。なお、引数に指定する値の妥当性は検証されません。

### (2) 形式

#### (a) 形式 1

```
DbjFetchInfo createFetchInfo()
```

#### (b) 形式 2

```
DbjFetchInfo createFetchInfo(
    int          startIndex,
    int          fetchCount,
    int          maxFetchCount,
    String       cacheName,
    int          cacheKey
)
```

#### (c) 形式 3

```
DbjFetchInfo createFetchInfo(
    int          startIndex,
    int          fetchCount,
    int          maxFetchCount,
    Comparator   comp,
    String       cacheName,
    int          cacheKey
)
```

### (3) 引数

#### startIndex ( 入力 )

startIndex プロパティの初期値を指定します。

#### fetchCount ( 入力 )

fetchCount プロパティの初期値を指定します。

#### maxFetchCount ( 入力 )

maxFetchCount プロパティの初期値を指定します。

#### comp ( 入力 )

comparator プロパティの初期値を指定します。

#### cacheName ( 入力 )

cacheName プロパティの初期値を指定します。

#### cacheKey ( 入力 )

cacheKey プロパティの初期値を指定します。

## 2. ファクトリクラス詳細

### (4) 戻り値

検索結果取得情報インターフェース (DbjFetchInfo インターフェース)

### (5) 例外

なし



## 2.2.4 createInteger32QParam ( INT 型の ? パラメタの作成 )

### (1) 機能

INT 型の ? パラメタを作成し、そのインターフェース ( DbjInteger32QParam インターフェース ) を取得します。なお、引数に指定する値の妥当性は検証されません。

### (2) 形式

```
DbjInteger32QParam createInteger32QParam(  
    int      param  
)
```

### (3) 引数

param ( 入力 )  
パラメタの値を指定します。

### (4) 戻り値

INT 型の ? パラメタインターフェース ( DbjInteger32QParam インターフェース )

### (5) 例外

なし

## 2.2.5 createOIIDQParam ( OIID の ? パラメタの作成 )

### (1) 機能

OIID の ? パラメタを作成し, そのインターフェース ( DbjOIIDQParam インターフェース ) を取得します。なお, 引数に指定する値の妥当性は検証されません。

### (2) 形式

```
DbjOIIDQParam createOIIDQParam(  
    DbjObj      param  
)
```

### (3) 引数

param ( 入力 )

パラメタの値を指定します。param.getOiid() の値が, ? パラメタとして指定されます。

### (4) 戻り値

OIID の ? パラメタインターフェース ( DbjOIIDQParam インターフェース )

### (5) 例外

なし

## 2.2.6 createObjQParam ( Object 型の ? パラメタの作成 )

### (1) 機能

Object 型の ? パラメタを作成し、そのインターフェース ( DbjObjQParam インターフェース ) を取得します。なお、引数に指定する値の妥当性は検証されません。

### (2) 形式

```
DbjObjQParam createObjQParam(  
    DbjObj    param  
)
```

### (3) 引数

param ( 入力 )  
パラメタの値を指定します。

### (4) 戻り値

Object 型の ? パラメタインターフェース ( DbjObjQParam インターフェース )

### (5) 例外

なし

## 2.2.7 createPropSet ( プロパティ値集合オブジェクトの作成 )

### (1) 機能

プロパティ値集合オブジェクトを作成し、そのインターフェース ( DbjPropSet インターフェース ) を取得します。

形式 1 では、空のプロパティ値集合オブジェクトを作成します。形式 2 では、指定されたマップをプロパティ値集合オブジェクトにマッピングして DbjPropSet インターフェースを取得します。

### (2) 形式

#### (a) 形式 1

```
DbjPropSet createPropSet ()
```

#### (b) 形式 2

```
DbjPropSet createPropSet (  
    Map      src  
)
```

### (3) 引数

src ( 入力 )

マッピングの元になるマップを指定します。要素キーにプロパティ名、要素の値にプロパティ値を設定しておきます。null を指定すると、要素数 0 の空のオブジェクトが仮定されます。

### (4) 戻り値

プロパティ値集合インターフェース ( DbjPropSet インターフェース )

### (5) 例外

なし

## 2.2.8 createPublicACLIdElm (パブリック ACL の OIID エLEMENTの作成)

### (1) 機能

パブリック ACL の OIID エLEMENTを作成し、そのインターフェース (DbjPublicACLIdElm インターフェース) を取得します。

形式 1 では、初期値状態のパブリック ACL の OIID エLEMENTを作成します。形式 2 では、指定されたプロパティ値集合をパブリック ACL の OIID エLEMENTとみなして取得して propSet プロパティに設定します (値はコピーされません)。形式 3 では、初期値に指定されたパブリック ACL の OIID を取得します。なお、形式 2 および形式 3 の引数に指定するプロパティ値集合の妥当性は検証されません。

### (2) 形式

#### (a) 形式 1

```
DbjPublicACLIdElm createPublicACLIdElm()
```

#### (b) 形式 2

```
DbjPublicACLIdElm createPublicACLIdElm(
    DbjPropSet      src
)
```

#### (c) 形式 3

```
DbjPublicACLIdElm createPublicACLIdElm(
    String          publicACLId
)
```

### (3) 引数

src (入力)

パブリック ACL の OIID エLEMENTとして扱うプロパティ値集合を指定します。null を指定した場合は、例外がスローされます。

publicACLId (入力)

パブリック ACL の OIID を指定します。null を指定した場合は、例外がスローされます。

### (4) 戻り値

パブリック ACL の OIID エLEMENTインターフェース (DbjPublicACLIdElm インターフェース)

### (5) 例外

NullPointerException

引数 src に null を指定した場合

## 2.2.9 createReferencePathInfo ( リファレンスファイル文書のパス情報オブジェクトの作成 )

### (1) 機能

リファレンスファイル文書のパス情報オブジェクトを作成し、そのインターフェース ( DbjReferencePathInfo インターフェース ) を取得します。

形式 1 では、オブジェクトのメンバにデフォルト値が入ります。形式 2 では、オブジェクトの初期値を引数で指定できます。オブジェクトの初期値として設定できるパラメタの詳細については、「3.14 DbjReferencePathInfo インターフェース」を参照してください。なお、引数に指定する値の妥当性は検証されません。

### (2) 形式

#### (a) 形式 1

```
DbjReferencePathInfo createReferencePathInfo()
```

#### (b) 形式 2

```
DbjReferencePathInfo createReferencePathInfo(  
    int          contentOperateMode,  
    String       entry,  
    String       targetPath,  
    String       deleteRootPath )
```

### (3) 引数

contentOperateMode ( 入力 )

contentOperateMode プロパティの初期値を指定します。

entry ( 入力 )

entry プロパティの初期値を指定します。

targetPath ( 入力 )

targetPath プロパティの初期値を指定します。なお、文書空間で使用する文字コード種別が UTF-8 の場合に指定できる値は、印刷が可能な ASCII コードだけです。

deleteRootPath ( 入力 )

deleteRootPath プロパティの初期値を指定します。なお、文書空間で使用する文字コード種別が UTF-8 の場合に指定できる値は、印刷が可能な ASCII コードだけです。

### (4) 戻り値

リファレンスファイル文書のパス情報インターフェース ( DbjReferencePathInfo インターフェース )

### (5) 例外

なし

## 2.2.10 createReferenceUploadInfo (リファレンスファイル文書のアップロード情報オブジェクトの作成)

### (1) 機能

リファレンスファイル文書のアップロード情報オブジェクトを作成し、そのインターフェース (DbjReferenceUploadInfo インターフェース) を取得します。

形式 1 では、オブジェクトのメンバにデフォルト値が入ります。形式 2 では、オブジェクトの初期値を引数で指定できます。オブジェクトの初期値として設定できるパラメタの詳細については、「3.15 DbjReferenceUploadInfo インターフェース」を参照してください。なお、引数に指定する値の妥当性は検証されません。

### (2) 形式

#### (a) 形式 1

```
DbjReferenceUploadInfo createReferenceUploadInfo ()
```

#### (b) 形式 2

```
DbjReferenceUploadInfo createReferenceUploadInfo (
    String          filePath,
    String          retrievalName,
    String          renditionType,
    DbjPropSet      renditionPropSet,
    String          indexPath,
    DbjReferencePathInfo referencePathInfo )
```

### (3) 引数

filePath (入力)

filePath プロパティの初期値を指定します。

retrievalName (入力)

retrievalName プロパティの初期値を指定します。

renditionType (入力)

renditionType プロパティの初期値を指定します。

renditionPropSet (入力)

renditionPropSet プロパティの初期値を指定します。

indexPath (入力)

indexPath プロパティの初期値を指定します。

referencePathInfo (入力)

referencePathInfo プロパティの初期値を指定します。

### (4) 戻り値

リファレンスファイル文書のアップロード情報インターフェース (DbjReferenceUploadInfo インターフェース)

### (5) 例外

なし

## 2.2.11 createSeedDocQParam (種文章の?パラメタの作成)

### (1) 機能

概念検索で指定する種文章の?パラメタを作成し、そのインターフェース (DbjSeedDocQParam インターフェース) を取得します。なお、引数に指定する値の妥当性は検証されません。

### (2) 形式

```
DbjSeedDocQParam createSeedDocQParam(  
    String param  
)
```

### (3) 引数

param (入力)  
パラメタの値を指定します。

### (4) 戻り値

概念検索の種文章の?パラメタインターフェース (DbjSeedDocQParam インターフェース)

### (5) 例外

なし



## 2.2.12 createSession ( Java クラスライブラリのセッションオブジェクトの作成 )

### (1) 機能

指定した文書空間に対する Java クラスライブラリのセッションオブジェクトを作成し、そのインターフェース ( DbjSession インターフェース ) を取得します。ただし、DbjSession インターフェースを取得した時点では文書空間にアクセスしません。

形式 1 では、セッションを開始する文書空間の文書空間識別子を指定します。形式 2 では、動作環境定義ファイルに指定したデフォルト文書空間 ( コンフィギュレーションキーは DefaultDocSpaceId ) に接続するためのセッションオブジェクトを作成し、そのインターフェース ( DbjSession インターフェース ) を取得します。

### (2) 形式

#### (a) 形式 1

```
DbjSession createSession(
    String      docspaceId
)
```

#### (b) 形式 2

```
DbjSession createSession()
```

### (3) 引数

docspaceId ( 入力 )

文書空間識別子を GUID 文字列で指定します。GUID 文字列は、16 進数「X」によって、「XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX」( 8 けた -4 けた -4 けた -4 けた -12 けた ) の形式で表されます。「X」は、0 ~ 9, a ~ f ( 小文字 ), および A ~ F ( 大文字 ) のどれかです。null を指定した場合は、例外がスローされます。

### (4) 戻り値

Java クラスライブラリのセッションインターフェース ( DbjSession インターフェース )

### (5) 例外

NullPointerException

引数 docspaceId に null を指定した場合

## 2.2.13 createSetDCRLinkInfo ( 直接型リンク設定情報の作成 )

### (1) 機能

直接型リンク設定情報オブジェクトを作成し、そのインターフェース ( DbjSetDCRLinkInfo インターフェース ) を取得します。

形式 1 では、オブジェクトのメンバにデフォルト値が入ります。形式 2 では、オブジェクトの初期値を引数で指定できます。オブジェクトの初期値として設定できるパラメタの詳細については、「3.20 DbjSetDCRLinkInfo インターフェース」を参照してください。

### (2) 形式

#### (a) 形式 1

```
DbjSetDCRLinkInfo createSetDCRLinkInfo()
```

#### (b) 形式 2

```
DbjSetDCRLinkInfo createSetDCRLinkInfo(  
    DbjObj          obj,  
    DbjPropSet     linkProps  
)
```

### (3) 引数

obj ( 入力 )

リンク対象の文書空間オブジェクトを指定します。文書空間オブジェクトを未設定にする場合は、null を指定できます。初期値のデフォルト値は null です。

linkProps ( 入力 )

リンクオブジェクトに設定するプロパティのプロパティ値集合を指定します。プロパティを未設定にする場合は、null を指定できます。初期値のデフォルト値は null です。

### (4) 戻り値

直接型リンク設定情報インターフェース ( DbjSetDCRLinkInfo インターフェース )

### (5) 例外

なし

## 2.2.14 createSetRCRLinkInfo ( 参照型リンク設定情報の作成 )

### (1) 機能

参照型リンク設定情報オブジェクトを作成し、そのインターフェース ( DbjSetRCRLinkInfo インターフェース ) を取得します。

形式 1 では、オブジェクトのメンバにデフォルト値が入ります。形式 2 では、オブジェクトの初期値を引数で指定できます。オブジェクトの初期値として設定できるパラメタの詳細については、「3.22 DbjSetRCRLinkInfo インターフェース」を参照してください。

### (2) 形式

#### (a) 形式 1

```
DbjSetRCRLinkInfo createSetRCRLinkInfo()
```

#### (b) 形式 2

```
DbjSetRCRLinkInfo createSetRCRLinkInfo(
    DbjObj          obj,
    DbjPropSet     linkProps
)
```

### (3) 引数

obj ( 入力 )

リンク対象の文書空間オブジェクトを指定します。文書空間オブジェクトを未設定にする場合は、null を指定できます。初期値のデフォルト値は null です。

linkProps ( 入力 )

リンクオブジェクトに設定するプロパティのプロパティ値集合を指定します。プロパティを未設定にする場合は、null を指定できます。初期値のデフォルト値は null です。

### (4) 戻り値

参照型リンク設定情報インターフェース ( DbjSetRCRLinkInfo インターフェース )

### (5) 例外

なし

## 2.2.15 createSetRelLinkInfo ( 文書間リンク設定情報の作成 )

### (1) 機能

文書間リンク設定情報オブジェクトを作成し、そのインターフェース ( DbjSetRelLinkInfo インターフェース ) を取得します。

形式 1 では、オブジェクトのメンバにデフォルト値が入ります。形式 2 では、オブジェクトの初期値を引数で指定できます。オブジェクトの初期値として設定できるパラメタの詳細については、「3.23 DbjSetRelLinkInfo インターフェース」を参照してください。

### (2) 形式

#### (a) 形式 1

```
DbjSetRelLinkInfo createSetRelLinkInfo()
```

#### (b) 形式 2

```
DbjSetRelLinkInfo createSetRelLinkInfo (  
    DbjObj          obj,  
    DbjPropSet     linkProps  
)
```

### (3) 引数

obj (入力)

リンク対象の文書空間オブジェクト ( バージョン付きオブジェクト ) を指定します。文書空間オブジェクトを未設定にする場合は、null を指定できます。初期値のデフォルト値は null です。

linkProps (入力)

リンクオブジェクトに設定するプロパティのプロパティ値集合を指定します。プロパティを未設定にする場合は、null を指定できます。初期値のデフォルト値は null です。

### (4) 戻り値

文書間リンク設定情報インターフェース ( DbjSetRelLinkInfo インターフェース )

### (5) 例外

なし

## 2.2.16 createSetVTFixLinkInfo ( 構成管理モードが FIX モードの構成管理型リンク設定情報の作成 )

### (1) 機能

構成管理型リンク ( 構成管理モードは FIX モード ) 設定情報オブジェクトを作成し、そのインターフェース ( DbjSetVTFixLinkInfo インターフェース ) を取得します。

形式 1 では、オブジェクトのメンバにデフォルト値が入ります。形式 2 では、オブジェクトの初期値を引数で指定できます。オブジェクトの初期値として設定できるパラメタの詳細については、「3.24 DbjSetVTFixLinkInfo インターフェース」を参照してください。

### (2) 形式

#### (a) 形式 1

```
DbjSetVTFixLinkInfo createSetVTFixLinkInfo()
```

#### (b) 形式 2

```
DbjSetVTFixLinkInfo createSetVTFixLinkInfo(
    DbjObj          obj,
    DbjPropSet     linkProps
)
```

### (3) 引数

#### obj ( 入力 )

リンク対象の文書空間オブジェクト ( バージョン付きオブジェクト ) を指定します。文書空間オブジェクトを未設定にする場合は、null を指定できます。初期値のデフォルト値は null です。

DbjObj#getTargetVersion メソッドで取得できるバージョンに固定した構成管理型リンク情報 ( 構成管理モードは FIX モード ) が設定されます。これらの情報はコピーされてリンク設定情報に保持されます。

#### linkProps ( 入力 )

リンクオブジェクトに設定するプロパティのプロパティ値集合を指定します。プロパティを未設定にする場合は、null を指定できます。初期値のデフォルト値は null です。

### (4) 戻り値

構成管理モードが FIX モードの構成管理型リンク設定情報インターフェース ( DbjSetVTFixLinkInfo インターフェース )

### (5) 例外

なし

## 2.2.17 createSetVTFloatLinkInfo ( 構成管理モードが FLOATING モードの構成管理型リンク設定情報の作成 )

### (1) 機能

構成管理型リンク ( 構成管理モードは FLOATING モード ) 設定情報オブジェクトを作成し、そのインターフェース ( DbjSetVTFloatLinkInfo インターフェース ) を取得します。

形式 1 では、オブジェクトのメンバにデフォルト値が入ります。形式 2 では、オブジェクトの初期値を引数で指定できます。オブジェクトの初期値として設定できるパラメタの詳細については、「3.25 DbjSetVTFloatLinkInfo インターフェース」を参照してください。

### (2) 形式

#### (a) 形式 1

```
DbjSetVTFloatLinkInfo createSetVTFloatLinkInfo()
```

#### (b) 形式 2

```
DbjSetVTFloatLinkInfo createSetVTFloatLinkInfo(  
    DbjObj          obj,  
    DbjPropSet     linkProps  
)
```

### (3) 引数

#### obj ( 入力 )

リンク対象の文書空間オブジェクト ( バージョン付きオブジェクト ) を指定します。文書空間オブジェクトを未設定にする場合は、null を指定できます。初期値のデフォルト値は null です。

DbjObj#getTargetVersion メソッドで取得できるバージョン以降の最新バージョンを指す構成管理型リンク情報 ( 構成管理モードは FLOATING モード ) が設定されます。

#### linkProps ( 入力 )

リンクオブジェクトに設定するプロパティのプロパティ値集合を指定します。プロパティを未設定にする場合は、null を指定できます。初期値のデフォルト値は null です。

### (4) 戻り値

構成管理モードが FLOATING モードの構成管理型リンク設定情報インターフェース ( DbjSetVTFloatLinkInfo インターフェース )

### (5) 例外

なし

## 2.2.18 createStringQParam (String 型の ? パラメタの作成)

### (1) 機能

String 型の ? パラメタを作成し、そのインターフェース (DbjStringQParam インターフェース) を取得します。なお、引数に指定する値の妥当性は検証されません。

### (2) 形式

```
DbjStringQParam createStringQParam(  
    String param  
)
```

### (3) 引数

param (入力)  
? パラメタの値を指定します。

### (4) 戻り値

String 型の ? パラメタ (DbjStringQParam インターフェース)

### (5) 例外

なし

## 2.2.19 createUploadInfo ( 文書のアップロード情報オブジェクトの作成 )

### (1) 機能

文書アップロード情報オブジェクトを作成し、そのインターフェース ( DbjUploadInfo インターフェース ) を取得します。

形式 1 では、オブジェクトのメンバにデフォルト値が入ります。形式 2 では、オブジェクトの初期値を引数で指定できます。オブジェクトの初期値として設定できるパラメタの詳細については、「3.28 DbjUploadInfo インターフェース」を参照してください。なお、引数に指定する値の妥当性は検証されません。

### (2) 形式

#### (a) 形式 1

```
DbjUploadInfo createUploadInfo()
```

#### (b) 形式 2

```
DbjUploadInfo createUploadInfo(  
    String          filePath,  
    String          retrievalName,  
    String          renditionType,  
    DbjPropSet      renditionPropSet,  
    String          indexPath  
)
```

### (3) 引数

filePath ( 入力 )

filePath プロパティの初期値を指定します。

retrievalName ( 入力 )

retrievalName プロパティの初期値を指定します。

renditionType ( 入力 )

renditionType プロパティの初期値を指定します。

renditionPropSet ( 入力 )

renditionPropSet プロパティの初期値を指定します。

indexPath ( 入力 )

indexPath プロパティの初期値を指定します。

### (4) 戻り値

アップロード情報インターフェース ( DbjUploadInfo インターフェース )

### (5) 例外

なし



## 2.2.20 createVArray ( 可変長配列オブジェクトの作成 )

### (1) 機能

可変長配列オブジェクトを作成し、そのインターフェース ( DbjVArray インターフェース ) を取得します。引数に指定されたメタプロパティを保持する要素数 0 の空の可変長配列オブジェクトを作成します。

### (2) 形式

```
DbjVArray createVArray(  
    Collection      propDefs  
)
```

### (3) 引数

propDefs ( 入力 )

各要素は、String 型のプロパティになります。可変長配列のメタプロパティのコレクションを指定します。指定を省略する場合は、各要素に null を指定できます。この場合、要素数 0 の空のオブジェクトが仮定されます。

### (4) 戻り値

可変長配列インターフェース ( DbjVArray インターフェース )

### (5) 例外

なし

## 2.2.21 createXmlTranslator (XML トランスレーターの取得)

### (1) 機能

XML トランスレーターインターフェース (DbjXmlTranslator インターフェース) を取得します。なお、同じ引数の指定値で再度実行してもエラーになりません。

### (2) 形式

```
DbjXmlTranslator createXmlTranslator(  
    String      docspaceId,  
    String      xmsFileName  
)
```

### (3) 引数

docspaceId (入力)

文書空間識別子を GUID 文字列で指定します。GUID 文字列は、16 進数「X」によって「XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX」(8 けた -4 けた -4 けた -4 けた -12 けた)の形式で表されます。「X」は、0 ~ 9, a ~ f (小文字), および A ~ F (大文字) のどれかです。null を指定した場合は、例外が出力されます。

xmsFileName (入力)

XMS ファイル名を指定します。null を指定した場合は、デフォルトの XMS ファイルが仮定されません。

### (4) 戻り値

XML トランスレーターインターフェース (DbjXmlTranslator インターフェース)

### (5) 例外

DbjException

Java クラスライブラリ固有のエラーの場合

IllegalArgumentException

指定した引数が不正の場合

NullPointerException

引数 docspaceId に null を指定した場合

## 2.2.22 createXmlUploadInfo (XML 文書のアップロード情報オブジェクトの作成)

### (1) 機能

XML 文書アップロード情報オブジェクトを作成し、そのインターフェース (DbjXmlUploadInfo インターフェース) を取得します。

形式 1 では、オブジェクトのメンバにデフォルト値が入ります。形式 2 では、初期値を引数で指定できます。オブジェクトの初期値として設定できるパラメタの詳細については、「3.30 DbjXmlUploadInfo インターフェース」を参照してください。なお、引数に指定する値の妥当性は検証されません。

### (2) 形式

#### (a) 形式 1

```
DbjXmlUploadInfo createXmlUploadInfo()
```

#### (b) 形式 2

```
DbjXmlUploadInfo createXmlUploadInfo(
    String          filePath,
    String          retrievalName,
    String          renditionType,
    DbjPropSet      renditionPropSet,
    int             parseLevel,
    String          mappingId,
    int             indexType,
    String          filterFilePath
)
```

### (3) 引数

filePath (入力)

filePath プロパティの初期値を指定します。

retrievalName (入力)

retrievalName プロパティの初期値を指定します。

renditionType (入力)

renditionType プロパティの初期値を指定します。

renditionPropSet (入力)

renditionPropSet プロパティの初期値を指定します。

parseLevel (入力)

parseLevel プロパティの初期値を指定します。

mappingId (入力)

mappingId プロパティの初期値を指定します。

indexType (入力)

indexType プロパティの初期値を指定します。

filterFilePath (入力)

filterFilePath プロパティの初期値を指定します。

## 2. ファクトリクラス詳細

### (4) 戻り値

XML 文書アップロード情報インターフェース (DbjXmlUploadInfo インターフェース)

### (5) 例外

なし

# 3

## パラメタクラス詳細

この章では，パラメタクラスのインターフェース，およびメソッドについて説明します。

- 
- 3.1 DbjACE インターフェース

---

  - 3.2 DbjBooleanQParam インターフェース

---

  - 3.3 DbjCheckOutInfo インターフェース

---

  - 3.4 DbjContentInfo インターフェース

---

  - 3.5 DbjFetchInfo インターフェース

---

  - 3.6 DbjInteger32QParam インターフェース

---

  - 3.7 DbjMappedProp インターフェース

---

  - 3.8 DbjObjQParam インターフェース

---

  - 3.9 DbjOIDQParam インターフェース

---

  - 3.10 DbjPropSet インターフェース

---

  - 3.11 DbjPublicACLIdElm インターフェース

---

  - 3.12 DbjQParam インターフェース

---

  - 3.13 DbjReferenceContentInfo インターフェース

---

  - 3.14 DbjReferencePathInfo インターフェース

---

  - 3.15 DbjReferenceUploadInfo インターフェース

---

  - 3.16 DbjRenditionInfo インターフェース

---

  - 3.17 DbjRenditionList インターフェース

---

  - 3.18 DbjResultSet インターフェース

---

  - 3.19 DbjSeedDocQParam インターフェース

---

  - 3.20 DbjSetDCRLinkInfo インターフェース

---

  - 3.21 DbjSetLinkInfo インターフェース

---

3.22 DbjSetRCRLinkInfo インターフェース

---

3.23 DbjSetRelLinkInfo インターフェース

---

3.24 DbjSetVTFixLinkInfo インターフェース

---

3.25 DbjSetVTFloatLinkInfo インターフェース

---

3.26 DbjSetVTLinkInfo インターフェース

---

3.27 DbjStringQParam インターフェース

---

3.28 DbjUploadInfo インターフェース

---

3.29 DbjVArray インターフェース

---

3.30 DbjXmlUploadInfo インターフェース

---

## 3.1 DbjACE インターフェース

DbjACE インターフェースは、ACE の値を扱うインターフェースです。ACE の値を取得、または設定するメソッドを提供します。また、サブジェクトと同時にサブジェクト種別を設定するメソッドも提供します。

なお、Java クラスライブラリでは、ACL を可変長配列として扱い、ACE を可変長配列の要素として扱います。このため、ACL は DbjVArray インターフェースでも操作でき、ACE は DbjPropSet インターフェースでも操作できます。

### プロパティ一覧

DbjACE インターフェースで扱うプロパティの一覧と、各プロパティに対する getter メソッドおよび setter メソッドを表 3-1 に示します。

表 3-1 DbjACE インターフェースで扱うプロパティ

プロパティ名	データ型	getter メソッド	setter メソッド
permission	int 型	getPermission	setPermission
propSet	DbjPropSet	propSet	setPropSet
subject	String 型	getSubject	setSubject
subjectType	int 型	getSubjectType	setSubjectType

### プロパティ詳細

- permission プロパティ  
ACE のパーミッションを表します。  
このプロパティに対する操作は、プロパティ値集合の dbrProp\_Permission プロパティに反映されます。  
初期値は DbjDef.PERM\_NONE (アクセス権なし) です。
- propSet プロパティ  
DbjACE インターフェースで扱うプロパティ値の集合です。  
このプロパティ値集合に対する操作は、集合に含まれる各プロパティ値に反映されます。また、各プロパティ値に対する操作もこのプロパティ値集合に反映されます。
- subject プロパティ  
ACE のサブジェクトを表します。  
このプロパティに対する操作は、プロパティ値集合の dbrProp\_Subject プロパティに反映されます。  
初期値は null です。
- subjectType プロパティ  
ACE のサブジェクト種別を表します。  
このプロパティに対する操作は、プロパティ値集合の dbrProp\_SubjectType プロパティに反映されます。  
初期値は 0 です。

以降、DbjACE インターフェースのメソッドについて説明します。

### 3.1.1 getPermission ( パーミッションの取得 )

#### (1) 機能

ACE のパーミッションを取得します。

permission プロパティに値が設定されていない場合は、0 が返却されます。

#### (2) 形式

```
int getPermission()
```

#### (3) 引数

なし

#### (4) 戻り値

ACE のパーミッションが、次に示す定数で返却されます。

- DbjDef.PERM\_NONE  
アクセス権なし
- DbjDef.PERM\_PRIM\_READ\_PROPS  
基本プロパティ参照権
- DbjDef.PERM\_PRIM\_WRITE\_PROPS  
基本プロパティ更新権
- DbjDef.PERM\_PRIM\_READ\_CONTENTS  
基本コンテンツ参照権
- DbjDef.PERM\_PRIM\_WRITE\_CONTENTS  
基本コンテンツ更新権
- DbjDef.PERM\_PRIM\_LINK  
基本リンク権
- DbjDef.PERM\_PRIM\_VERSION  
基本バージョン管理権
- DbjDef.PERM\_PRIM\_DELETE  
基本削除権
- DbjDef.PERM\_CHANGE\_PERM  
アクセス制御情報変更権
- DbjDef.PERM\_CREATE  
オブジェクト作成権
- DbjDef.PERM\_READ\_PROPS  
プロパティ参照権
- DbjDef.PERM\_READ  
参照権
- DbjDef.PERM\_WRITE\_PROPS  
プロパティ更新権
- DbjDef.PERM\_READ\_WRITE  
参照更新権
- DbjDef.PERM\_DELETE  
削除権
- DbjDef.PERM\_LINK  
リンク権



- DbjDef.PERM\_VERSION  
バージョン権
- DbjDef.PERM\_FULL\_CONTROL  
フルコントロール

(5) 例外

なし

### 3.1.2 getSubject ( サブジェクトの取得 )

#### (1) 機能

ACE のサブジェクトを取得します。

subject プロパティに値が設定されていない場合は、null が返却されます。

#### (2) 形式

```
String getSubject ()
```

#### (3) 引数

なし

#### (4) 戻り値

subject プロパティ

ACE のサブジェクトが返却されます。

#### (5) 例外

なし

### 3.1.3 getSubjectType ( サブジェクト種別の取得 )

#### (1) 機能

ACE のサブジェクト種別を取得します。

subjectType プロパティに値が設定されていない場合は、0 が返却されます。

#### (2) 形式

```
int getSubjectType()
```

#### (3) 引数

なし

#### (4) 戻り値

ACE のサブジェクト種別が、次に示す定数で返却されます。

- DbjDef.SUBJECTTYPE\_USR  
ユーザサブジェクト
- DbjDef.SUBJECTTYPE\_GRP  
グループサブジェクト
- DbjDef.SUBJECTTYPE\_SYS  
システムサブジェクト

#### (5) 例外

なし

### 3.1.4 propSet ( ACE のプロパティ値集合の取得 )

#### (1) 機能

ACE の値を要素としたプロパティ値集合を取得します。

取得するプロパティ値集合の各要素は、プロパティ値の参照です。なお、値が設定されていない場合、`null` は返却されません。

#### (2) 形式

```
DbjPropSet propSet()
```

#### (3) 引数

なし

#### (4) 戻り値

propSet プロパティ ( DbjPropSet インターフェース )

#### (5) 例外

なし

### 3.1.5 setGroupSubject ( サブジェクトをグループサブジェクトとして設定 )

#### (1) 機能

指定したサブジェクトをグループサブジェクトとして設定します。subjectType プロパティに、DbjDef.SUBJECTTYPE\_GRP (グループサブジェクト) が自動的に設定されます。

なお、指定した値の妥当性は検証されません。

#### (2) 形式

```
void setGroupSubject (  
    String      grpSubject  
)
```

#### (3) 引数

grpSubject ( 入力 )

グループサブジェクトとして設定するサブジェクトを指定します。

#### (4) 戻り値

なし

#### (5) 例外

なし

### 3.1.6 setPermission (パーミッションの設定)

#### (1) 機能

ACE のパーミッションを設定します。

なお、指定した値の妥当性は検証されません。

#### (2) 形式

```
void setPermission(  
    int      permission  
)
```

#### (3) 引数

permission (入力)

ACE のパーミッションを指定します。次に示す定数のどれかを指定できます。

- DbjDef.PERM\_NONE  
アクセス権なし
- DbjDef.PERM\_PRIM\_READ\_PROPS  
基本プロパティ参照権
- DbjDef.PERM\_PRIM\_WRITE\_PROPS  
基本プロパティ更新権
- DbjDef.PERM\_PRIM\_READ\_CONTENTS  
基本コンテンツ参照権
- DbjDef.PERM\_PRIM\_WRITE\_CONTENTS  
基本コンテンツ更新権
- DbjDef.PERM\_PRIM\_LINK  
基本リンク権
- DbjDef.PERM\_PRIM\_VERSION  
基本バージョン管理権
- DbjDef.PERM\_PRIM\_DELETE  
基本削除権
- DbjDef.PERM\_CHANGE\_PERM  
アクセス制御情報変更権
- DbjDef.PERM\_CREATE  
オブジェクト作成権
- DbjDef.PERM\_READ\_PROPS  
プロパティ参照権
- DbjDef.PERM\_READ  
参照権
- DbjDef.PERM\_WRITE\_PROPS  
プロパティ更新権
- DbjDef.PERM\_READ\_WRITE  
参照更新権
- DbjDef.PERM\_DELETE  
削除権
- DbjDef.PERM\_LINK  
リンク権
- DbjDef.PERM\_VERSION

バージョン権

- DbjDef.PERM\_FULL\_CONTROL

フルコントロール

(4) 戻り値

なし

(5) 例外

なし

### 3.1.7 setPropSet ( ACE のプロパティ値集合の設定 )

#### (1) 機能

ACE の値に、指定したプロパティ値集合の内容をコピーして設定します。

#### (2) 形式

```
void setPropSet (  
    DbjPropSet propSet  
)
```

#### (3) 引数

propSet ( 入力 )

設定するプロパティ値集合を指定します。

#### (4) 戻り値

なし

#### (5) 例外

NullPointerException

引数 propSet に null を指定した場合



### 3.1.8 setSubject (サブジェクトの設定)

#### (1) 機能

ACE のサブジェクトを設定します。

なお、指定した値の妥当性は検証されません。

#### (2) 形式

```
void setSubject (  
    String      subject  
)
```

#### (3) 引数

subject (入力)

ACE のサブジェクトを指定します。

#### (4) 戻り値

なし

#### (5) 例外

なし

### 3.1.9 setSubjectType (サブジェクト種別の設定)

#### (1) 機能

ACE のサブジェクト種別を設定します。

なお、指定した値の妥当性は検証されません。

#### (2) 形式

```
void setSubjectType(  
    int      subjectType  
)
```

#### (3) 引数

subjectType (入力)

ACE のサブジェクト種別を指定します。次に示す定数のどれかを指定できます。

- DbjDef.SUBJECTTYPE\_USR  
ユーザサブジェクト
- DbjDef.SUBJECTTYPE\_GRP  
グループサブジェクト
- DbjDef.SUBJECTTYPE\_SYS  
システムサブジェクト

#### (4) 戻り値

なし

#### (5) 例外

なし

### 3.1.10 setSystemSubject (サブジェクトをシステムサブジェクトとして設定)

#### (1) 機能

指定したサブジェクトをシステムサブジェクトとして設定します。subjectType プロパティに、DbjDef.SUBJECTTYPE\_SYS が自動的に設定されます。

なお、指定した値の妥当性は検証されません。

#### (2) 形式

```
void setSystemSubject(  
    String sysSubject  
)
```

#### (3) 引数

sysSubject (入力)

システムサブジェクトとして設定するサブジェクトを指定します。次に示す定数のどちらかを指定できます。

- DbjDef.SYSSUBJECT\_SELF  
対象オブジェクトの所有者
- DbjDef.SYSSUBJECT\_EVERYONE  
すべてのユーザ

#### (4) 戻り値

なし

#### (5) 例外

なし

### 3.1.11 setUserSubject ( サブジェクトをユーザサブジェクトとして設定 )

#### (1) 機能

指定したサブジェクトをユーザサブジェクトとして設定します。subjectType プロパティに、DbjDef.SUBJECTTYPE\_USR が自動的に設定されます。

なお、指定した値の妥当性は検証されません。

#### (2) 形式

```
void setUserSubject (  
    String      usrSubject  
)
```

#### (3) 引数

usrSubject ( 入力 )

ユーザサブジェクトとして設定するサブジェクトを指定します。

#### (4) 戻り値

なし

#### (5) 例外

なし

## 3.2 DbjBooleanQParam インターフェース

---

DbjBooleanQParam インターフェースは、BOOL 型を表す?パラメタを扱うインターフェースです。

直接のスーパーインターフェース

- DbjQParam

## 3.3 DbjCheckOutInfo インターフェース

DbjCheckOutInfo インターフェースは、バージョン付きオブジェクトのチェックアウト情報を扱うインターフェースです。バージョン付きオブジェクトのチェックアウト情報を取得するメソッドを提供します。チェックアウト中の場合は、チェックアウトしたユーザのユーザ識別子および仮のバージョン識別子を取得できます。

### プロパティ一覧

DbjCheckOutInfo インターフェースで扱うプロパティの一覧と、各プロパティに対する getter メソッドを表 3-2 に示します。

表 3-2 DbjCheckOutInfo インターフェースで扱うプロパティ

プロパティ名	データ型	getter メソッド
checkOut	boolean 型	isCheckOut
checkOutUserId	String 型	getCheckOutUserId
checkOutVersionId	String 型	getCheckOutVersionId

### プロパティ詳細

- checkOut プロパティ  
バージョン付きオブジェクトがチェックアウト中かどうかを表します。  
チェックアウト中の場合は true となり、チェックアウトしていない場合は false となります。
- checkOutUserId プロパティ  
チェックアウトしたユーザのユーザ識別子を表します。  
チェックアウト中の場合にだけ有効です。チェックアウトしていない場合は null となります。
- checkOutVersionId プロパティ  
チェックアウト中の仮のバージョン識別子を表します。  
チェックアウト中の場合にだけ有効です。チェックアウトしていない場合は null となります。

以降、DbjCheckOutInfo インターフェースのメソッドについて説明します。

### 3.3.1 getCheckOutUserId (チェックアウトしたユーザのユーザ識別子の取得)

#### (1) 機能

チェックアウトしたユーザのユーザ識別子を取得します。

チェックアウトしていない場合は、null が返却されます。

#### (2) 形式

```
String getCheckOutUserId()
```

#### (3) 引数

なし

#### (4) 戻り値

checkOutUserId プロパティ

#### (5) 例外

なし

### 3.3.2 getCheckoutVersionId ( 仮のバージョン識別子の取得 )

#### (1) 機能

チェックアウト中の仮のバージョン識別子を取得します。

チェックアウトしていない場合は、null が返却されます。

#### (2) 形式

```
String getCheckoutVersionId()
```

#### (3) 引数

なし

#### (4) 戻り値

checkoutVersionId プロパティ

#### (5) 例外

なし



### 3.3.3 isCheckOut (チェックアウト中かどうかの判定)

#### (1) 機能

チェックアウト中かどうかを判定します。

#### (2) 形式

```
boolean isCheckOut()
```

#### (3) 引数

なし

#### (4) 戻り値

true

チェックアウト中です。

false

チェックアウトしていません。

#### (5) 例外

なし

## 3.4 DbjContentInfo インターフェース

DbjContentInfo インターフェースは、文書のコンテンツ情報を扱うインターフェースです。コンテンツのレンディションタイプおよびファイル名を取得するメソッドを提供します。

### プロパティ一覧

DbjContentInfo インターフェースで扱うプロパティの一覧と、各プロパティに対する getter メソッドを表 3-3 に示します。

表 3-3 DbjContentInfo インターフェースで扱うプロパティ

プロパティ名	データ型	getter メソッド
renditionType	String 型	getRenditionType
retrievalName	String 型	getRetrievalName

### プロパティ詳細

- renditionType プロパティ  
コンテンツのレンディションタイプを表します。文書オブジェクトの dbrProp\_RenditionType プロパティと同じ値です。
- retrievalName プロパティ  
コンテンツのファイル名を表します。文書オブジェクトの dbrProp\_RetrievalName プロパティと同じ値です。

以降、DbjContentInfo インターフェースのメソッドについて説明します。

### 3.4.1 getRenditionType (コンテンツのレンディションタイプの取得)

#### (1) 機能

コンテンツのレンディションタイプを取得します。

#### (2) 形式

```
String getRenditionType()
```

#### (3) 引数

なし

#### (4) 戻り値

renditionType プロパティ

#### (5) 例外

なし

### 3.4.2 getRetrievalName (コンテンツのファイル名の取得)

(1) 機能

コンテンツのファイル名を取得します。

(2) 形式

`String getRetrievalName()`

(3) 引数

なし

(4) 戻り値

retrievalName プロパティ

(5) 例外

なし

## 3.5 DbjFetchInfo インターフェース

DbjFetchInfo インターフェースは、検索結果取得情報を扱うインターフェースです。検索結果の取得開始位置、取得件数および最大取得件数などを取得、設定するメソッドを提供します。また、キャッシュ検索の実行時に使用するキャッシュ名およびキャッシュキーを取得、設定できます。

キャッシュ検索を実行すると検索結果がキャッシュされ、以降の検索では検索結果キャッシュから検索結果を取得できます。検索結果キャッシュは、同一セッションの同一キャッシュ名に対して有効です。

直接のスーパーインターフェース

- java.lang.Cloneable

DbjFetchInfo インターフェースの clone メソッドは、ディープコピーを実行します。

プロパティ一覧

DbjFetchInfo インターフェースで扱うプロパティの一覧と、各プロパティに対する getter メソッドおよび setter メソッドを表 3-4 に示します。

表 3-4 DbjFetchInfo インターフェースで扱うプロパティ

プロパティ名	データ型	getter メソッド	setter メソッド
cacheKey	int 型	getCacheKey	setCacheKey
cacheName	String 型	getCacheName	setCacheName
cacheTotal	int 型	getCacheTotal	setCacheTotal
comparator	Comparator	getComparator	setComparator
fetchCount	int 型	getFetchCount	setFetchCount
maxFetchCount	int 型	getMaxFetchCount	setMaxFetchCount
startIndex	int 型	getStartIndex	setStartIndex

プロパティ詳細

- cacheKey プロパティ

キャッシュ検索の実行時に、検索結果キャッシュの整合性をチェックするためのキーです。キャッシュキーは INT 型の値で、再検索を実行して検索結果キャッシュが更新されるごとにインクリメントされます。キャッシュ検索を実行しない時、またはキャッシュ検索の初回実行時は無効です。指定したキャッシュキーと検索結果キャッシュに保持されているキーが一致した場合は、データベースへはアクセスしないで、検索結果キャッシュから検索結果を取得します。キャッシュキーが不一致の場合は、データベースを再検索して検索結果を取得します。このとき、検索結果キャッシュは更新されます。

キャッシュ検索を実行しない場合（データベースを再検索する場合）は、キャッシュキーに DbjDef.INITIAL\_KEY を指定してください。

初期値は DbjDef.INITIAL\_KEY です。

- cacheName プロパティ

検索結果のキャッシュ名を表します。キャッシュ検索を実行する場合に指定します。null または「」（空文字列）を指定すると、キャッシュ検索を実行しません（検索結果はキャッシュされません）。

初期値は null です。

- cacheTotal プロパティ

キャッシュ検索の実行時に設定される検索結果キャッシュの全件数を表します。

初期値は 0 です。

- comparator プロパティ

検索結果取得時に検索結果キャッシュの内容をソートする場合に使用する、`java.util.Comparator` インターフェースを表します。null を指定すると、検索結果キャッシュの内容はソートされません。

初期値は null です。

検索結果キャッシュの内容をソートするには、`java.util.Comparator#compare` メソッドを実装して、検索結果の要素を比較する必要があります。また、`java.util.Comparator#equals` メソッドを実装して、検索結果キャッシュに保持されているコンパレータと、今回の検索結果のコンパレータが等しいかどうかを判定する必要があります。コンパレータが等しいかどうかを判定する対象には、異なるインスタンスのコンパレータを指定してください。

コンパレータが等しいかどうかを判定した結果、等しい場合は、検索結果キャッシュはソートされません。等しくない場合は、検索結果キャッシュがソートされます。なお、キャッシュキーが一致している場合で、コンパレータが等しくないときは、データベースへの再検索は実行されないで、検索結果キャッシュのソートだけが実行されます。

- fetchCount プロパティ

検索結果の取得件数（検索結果を実際に何件取得するか）を表します。最大件数を取得したい場合は、`DbjDef.MAX_NUM` を指定します。

キャッシュ検索では、検索結果キャッシュに保持している件数以上は取得できません。

初期値は `DbjDef.MAX_NUM` です。

- maxFetchCount プロパティ

キャッシュ検索を実行する場合に、検索結果の最大取得件数（検索結果を最大で何件まで取得するか）を指定します。最大件数を取得したい場合は、`DbjDef.MAX_NUM` を指定します。

初期値は `DbjDef.MAX_NUM` です。

- startIndex プロパティ

検索結果の取得開始位置を表します。開始位置は 0 から始まるインデクスで指定します。

キャッシュ検索では、検索結果キャッシュ上の開始位置を表します。

初期値は 0 です。

以降、`DbjFetchInfo` インターフェースのメソッドについて説明します。

### 3.5.1 getCacheKey ( キャッシュキーの取得 )

#### (1) 機能

キャッシュキーを取得します。

#### (2) 形式

```
int getCacheKey()
```

#### (3) 引数

なし

#### (4) 戻り値

cacheKey プロパティ

#### (5) 例外

なし

### 3.5.2 getCacheName ( キャッシュ名の取得 )

(1) 機能

キャッシュ名を取得します。

(2) 形式

`String getCacheName()`

(3) 引数

なし

(4) 戻り値

cacheName プロパティ

(5) 例外

なし



### 3.5.3 getCacheTotal ( 検索結果キャッシュの全件数の取得 )

#### (1) 機能

検索結果キャッシュの全件数を取得します。

#### (2) 形式

```
int getCacheTotal()
```

#### (3) 引数

なし

#### (4) 戻り値

cacheTotal プロパティ

#### (5) 例外

なし

### 3.5.4 getComparator ( Comparator インターフェースの取得 )

#### (1) 機能

java.util.Comparator インターフェースを取得します。

#### (2) 形式

```
Comparator getComparator()
```

#### (3) 引数

なし

#### (4) 戻り値

java.util.Comparator インターフェース

#### (5) 例外

なし

### 3.5.5 getFetchCount ( 検索結果の取得件数の取得 )

#### (1) 機能

検索結果の取得件数 ( 検索結果を実際に何件取得するか ) を取得します。

#### (2) 形式

```
int getFetchCount ()
```

#### (3) 引数

なし

#### (4) 戻り値

fetchCount プロパティ

#### (5) 例外

なし

### 3.5.6 getMaxFetchCount ( 検索結果の最大取得件数の取得 )

(1) 機能

検索結果の最大取得件数 ( 検索結果を最大で何件まで取得するか ) を取得します。

(2) 形式

```
int getMaxFetchCount()
```

(3) 引数

なし

(4) 戻り値

maxFetchCount プロパティ

(5) 例外

なし

### 3.5.7 `getStartIndex` ( 検索結果の取得開始位置の取得 )

#### (1) 機能

検索結果の取得開始位置を取得します。

#### (2) 形式

```
int getStartIndex()
```

#### (3) 引数

なし

#### (4) 戻り値

`startIndex` プロパティ

#### (5) 例外

なし

### 3.5.8 setCacheKey ( キャッシュキーの設定 )

#### (1) 機能

キャッシュキーを設定します。

なお、指定した値の妥当性は検証されません。

#### (2) 形式

```
void setCacheKey(  
    int      cacheKey  
)
```

#### (3) 引数

cacheKey ( 入力 )

キャッシュキーを指定します。

キャッシュ検索を実行しない場合は、DbjDef.INITIAL\_KEY を指定してください。

#### (4) 戻り値

なし

#### (5) 例外

なし

### 3.5.9 setCacheName ( キャッシュ名の設定 )

#### (1) 機能

キャッシュ名を設定します。

なお、指定した値の妥当性は検証されません。

#### (2) 形式

```
void setCacheName (  
    String      cacheName  
)
```

#### (3) 引数

cacheName ( 入力 )

キャッシュ名を指定します。

キャッシュ検索を実行しない場合は、null または「」( 空文字列 ) を指定してください。

#### (4) 戻り値

なし

#### (5) 例外

なし

### 3.5.10 setCacheTotal ( 検索結果キャッシュの全件数の設定 )

#### (1) 機能

検索結果キャッシュの全件数を表す cacheTotal プロパティを設定します。

#### (2) 形式

```
void setCacheTotal(  
    int cacheTotal  
)
```

#### (3) 引数

cacheTotal ( 入力 )

cacheTotal プロパティに設定する値を指定します。

#### (4) 戻り値

なし

#### (5) 例外

なし



### 3.5.11 setComparator ( Comparator インターフェースの設定 )

#### (1) 機能

java.util.Comparator インターフェースを設定します。

なお、指定した値の妥当性は検証されません。

#### (2) 形式

```
void setComparator(  
    Comparator comp  
)
```

#### (3) 引数

comp ( 入力 )

java.util.Comparator インターフェースを指定します。

#### (4) 戻り値

なし

#### (5) 例外

なし

### 3.5.12 setFetchCount ( 検索結果の取得件数の設定 )

#### (1) 機能

検索結果の取得件数 ( 検索結果を実際に何件取得するか ) を設定します。

なお、指定した値の妥当性は検証されません。

#### (2) 形式

```
void setFetchCount (
    int      fetchCount
)
```

#### (3) 引数

fetchCount ( 入力 )

検索結果の取得件数を指定します。

最大件数取得したい場合は、DbjDef.MAX\_NUM を指定してください。

#### (4) 戻り値

なし

#### (5) 例外

なし

### 3.5.13 setMaxFetchCount ( 検索結果の最大取得件数の設定 )

#### (1) 機能

検索結果の最大取得件数 ( 検索結果を最大で何件まで取得するか ) を設定します。

なお、指定した値の妥当性は検証されません。

#### (2) 形式

```
void setMaxFetchCount (
    int      maxFetchCount
)
```

#### (3) 引数

maxFetchCount ( 入力 )

検索結果の最大取得件数を指定します。

全件取得したい場合は、DbjDef.MAX\_NUM を指定してください。

#### (4) 戻り値

なし

#### (5) 例外

なし

### 3.5.14 setStartIndex ( 検索結果の取得開始位置の設定 )

#### (1) 機能

検索結果の取得を開始する位置を設定します。

なお、指定した値の妥当性は検証されません。

#### (2) 形式

```
void setStartIndex(  
    int      startIndex  
)
```

#### (3) 引数

startIndex ( 入力 )

検索結果の取得開始位置を指定します。

0 から始まるインデクスで指定してください。

#### (4) 戻り値

なし

#### (5) 例外

なし

## 3.6 DbjInteger32QParam インターフェース

---

DbjInteger32QParam インターフェースは、INT 型を表す ? パラメタを扱うインターフェースです。

直接のスーパーインターフェース

- DbjQParam

## 3.7 DbjMappedProp インターフェース

DbjMappedProp インターフェースは、XML プロパティマッピング結果情報を扱うインターフェースです。XML プロパティマッピング機能を実行して取得した情報を取得するメソッドを提供します。

### プロパティ一覧

DbjMappedProp インターフェースで扱うプロパティの一覧と、各プロパティに対する getter メソッドを表 3-5 に示します。

表 3-5 DbjMappedProp インターフェースで扱うプロパティ

プロパティ名	データ型	getter メソッド
classList	List 型 (各要素は String 型)	getClassList
propSet	DbjPropSet	propSet

### プロパティ詳細

- classList プロパティ

マッピングされた DMA クラス名のリストです。

バージョン付きオブジェクトの DMA クラスがマッピングされた場合、リストとして取得する DMA クラス名の順序は、「バージョン付きオブジェクトの DMA クラス名」、「バージョンオブジェクトの DMA クラス名」となります。バージョン付きオブジェクトの DMA クラス以外がマッピングされた場合、要素は一つとなります。

- propSet プロパティ

マッピングされたプロパティ値集合です。

なお、バージョン付きオブジェクトの DMA クラスがマッピングされた場合は、バージョンオブジェクトのプロパティ名に@プレフィックスが付けられます。

以降、DbjMappedProp インターフェースのメソッドについて説明します。

### 3.7.1 getClassList ( マッピングされた DMA クラス名のリストの取得 )

#### (1) 機能

プロパティマッピングされた DMA クラス名のリストを取得します。

#### (2) 形式

```
List getClassList()
```

#### (3) 引数

なし

#### (4) 戻り値

classList プロパティ

#### (5) 例外

なし

### 3.7.2 propSet ( マッピングされたプロパティ値集合の取得 )

(1) 機能

プロパティマッピングされたプロパティ値集合を取得します。

(2) 形式

```
DbjPropSet propSet()
```

(3) 引数

なし

(4) 戻り値

propSet プロパティ ( DbjPropSet インターフェース )

(5) 例外

なし



## 3.8 DbjObjQParam インターフェース

---

DbjObjQParam インターフェースは、文書空間オブジェクトを表す?パラメタを扱うインターフェースです。

直接のスーパーインターフェース

- DbjQParam

## 3.9 DbjOIDQParam インターフェース

---

DbjOIDQParam インターフェースは、OID 文字列を表す?パラメタを扱うインターフェースです。

直接のスーパーインターフェース

- DbjQParam

## 3.10 DbjPropSet インターフェース

DbjPropSet インターフェースは、プロパティ値集合を扱うインターフェースです。プロパティ値を取得、設定するメソッド、およびプロパティ名を変更するメソッドを提供します。

プロパティ値集合とは、文書空間オブジェクトのプロパティ名とそのプロパティ値を一組にした要素の集合オブジェクトです。要素のキーを文書空間オブジェクトのプロパティ名とし、要素の値をそのプロパティ値とするマップで表します。

直接のスーパーインターフェース

- java.lang.Cloneable  
DbjPropSet インターフェースの clone メソッドは、ディープコピーを実行します。
- java.util.Map  
DbjPropSet インターフェースは、java.util.Map インターフェースを継承します。したがって、プロパティ値集合にプロパティ間（要素間）の順序性はなく、プロパティ名（要素のキー）は重複しません。また、マップに対する一般的な操作を使用して、プロパティ値集合を操作することもできます。

コーディング例

```
// ユーザアプリケーションプログラムのコーディング例
// mdmProp_Nameプロパティ (STR型),
// mdmProp_Numberプロパティ (INT型) について,
// name, numberで与えられた値でプロパティ値集合を作成します。

DbjPropSet MakeProps(String name, int number)
{
// ファクトリから空プロパティ値集合を取得
  DbjPropSet propset = DbjFactory0200.getFactory().createPropSet();

// mdmProp_Nameプロパティの設定
  propset.setPropVal("mdmProp_Name", name);

// mdmProp_Numberプロパティの設定
  propset.setPropVal("mdmProp_Number", number);

  return propset;
}
```

以降、DbjPropSet インターフェースのメソッドについて説明します。

### 3.10.1 changePropName ( プロパティ名の変更 )

#### (1) 機能

指定したプロパティのプロパティ名を変更します。

変更後のプロパティ名がすでに存在していても、例外はスローされません。ただし、プロパティ名が変更前、変更後のどちらになるかは保証できません。

なお、指定したプロパティ名がプロパティ値集合にない場合は、`false` が返却されます。

#### (2) 形式

```
boolean changePropName(  
    String      oldName,  
    String      newName  
)
```

#### (3) 引数

`oldName` ( 入力 )

変更前のプロパティ名を指定します。

`null` を指定した場合は、例外がスローされます。

`newName` ( 入力 )

変更後のプロパティ名を指定します。

`null` を指定した場合は、例外がスローされます。

#### (4) 戻り値

`true`

変更されました。

`false`

変更されませんでした。

#### (5) 例外

`NullPointerException`

引数 `oldName` または `newName` に `null` を指定した場合

## 3.10.2 changePropNames ( プロパティ名の一括変更 )

### (1) 機能

指定したマップの内容に従って、プロパティ名を一括変更します。マップは、要素のキーに変更前のプロパティ名を、要素の値に変更後のプロパティ名を設定します。

変更後のプロパティ名がすでに存在していても、例外はスローされません。ただし、プロパティ名が変更前、変更後のどちらになるかは保証できません。

なお、変更前として指定したプロパティ名がプロパティ値集合にない場合は、`false` が返却されます。

### (2) 形式

```
boolean changePropNames (  
    Map      changeMap  
)
```

### (3) 引数

`changeMap` ( 入力 )

変更前と変更後のプロパティ名を設定したマップを指定します。

### (4) 戻り値

`true`

一つ以上変更されました。

`false`

変更されませんでした。

### (5) 例外

`NullPointerException`

指定したマップの任意のキーまたは値に `null` を指定した場合

### 3.10.3 getIntegerVal ( Integer 型でのプロパティ値の取得 )

#### (1) 機能

指定したプロパティの値を Integer 型で取得します。

指定したプロパティがない場合、またはプロパティの値が NULL 値の場合は、null が返却されます。プロパティの値が INT 型でない場合は、例外がスローされます。

#### (2) 形式

```
Integer getIntegerVal(  
    String      propName  
)
```

#### (3) 引数

propName (入力)

プロパティ名を指定します。

#### (4) 戻り値

プロパティ値 ( Integer 型 )

#### (5) 例外

ClassCastException

プロパティ値が INT 型でなかった場合

NullPointerException

引数 propName に null を指定した場合

### 3.10.4 getIntVal ( int 型でのプロパティ値の取得 )

#### (1) 機能

指定したプロパティの値を int 型で取得します。

指定したプロパティがない場合、またはプロパティの値が NULL 値の場合は、0 が返却されます。プロパティの値が INT 型でない場合は、java.lang.Object#toString メソッドの実行結果がさらに int 型の値に変換されて返却されます。

#### (2) 形式

```
int getIntVal(  
    String    propName  
)
```

#### (3) 引数

propName ( 入力 )  
プロパティ名を指定します。

#### (4) 戻り値

プロパティ値 ( int 型 )

#### (5) 例外

NullPointerException  
引数 propName に null を指定した場合

NumberFormatException  
数値変換に失敗した場合

### 3.10.5 getListRef ( List 型でのプロパティ値の取得 )

#### (1) 機能

指定したプロパティの値を List 型で取得します。このメソッドは、ログインユーザのユーザ情報 ( dbrProp\_GroupList プロパティ ) の参照を取得するのに使用します。

指定したプロパティがない場合、またはプロパティの値が NULL 値の場合は、null が返却されます。プロパティの値が STRLIST 型でない場合は、例外がスローされます。

#### (2) 形式

```
List getListRef (
    String      propName
)
```

#### (3) 引数

propName ( 入力 )  
プロパティ名を指定します。

#### (4) 戻り値

プロパティ値のリスト ( List インターフェース )

#### (5) 例外

ClassCastException  
プロパティ値が STRLIST 型でなかった場合

NullPointerException  
引数 propName に null を指定した場合



### 3.10.6 getStringVal (String 型でのプロパティ値の取得)

#### (1) 機能

指定したプロパティの値を String 型で取得します。

指定したプロパティがない場合、またはプロパティの値が NULL 値の場合は、null が返却されます。プロパティの値が STR 型でない場合は、java.lang.Object#toString メソッドを実行した結果が返却されます。

#### (2) 形式

```
String getStringVal(  
    String      propName  
)
```

#### (3) 引数

propName (入力)  
プロパティ名を指定します。

#### (4) 戻り値

プロパティ値 (String 型)

#### (5) 例外

NullPointerException  
引数 propName に null を指定した場合

### 3.10.7 getVArrayRef ( VARRAY 型でのプロパティ値の参照の取得 )

#### (1) 機能

指定した VARRAY 型プロパティの値の参照を取得します。

指定したプロパティがない場合、またはプロパティの値が NULL 値の場合は、null が返却されます。プロパティの値が VARRAY 型でない場合は、例外がスローされます。

#### (2) 形式

```
DbjVArray getVArrayRef (  
    String      propName  
)
```

#### (3) 引数

propName (入力)

プロパティ名を指定します。

#### (4) 戻り値

VARRAY 型プロパティ値の参照 ( DbjVArray インターフェース )

#### (5) 例外

ClassCastException

プロパティ値が VARRAY 型でなかった場合

NullPointerException

引数 propName に null を指定した場合

### 3.10.8 getVArrayVal ( VARRAY 型でのプロパティ値の取得 )

#### (1) 機能

指定した VARRAY 型プロパティの値のコピーを取得します。

指定したプロパティがない場合、またはプロパティの値が NULL 値の場合は、null が返却されます。プロパティの値が VARRAY 型でない場合は、例外がスローされます。

なお、取得するプロパティ値は、次に示すメソッドの実行結果と同じです。ただし、getVArrayVal メソッドでは、プロパティ値が NULL 値の場合に null が返却されます。

```
DbjVArray getVArrayRef(propName).clone()
```

#### (2) 形式

```
DbjVArray getVArrayVal(  
    String      propName  
)
```

#### (3) 引数

propName ( 入力 )  
プロパティ名を指定します。

#### (4) 戻り値

VARRAY 型プロパティ値 ( DbjVArray インターフェース )

#### (5) 例外

ClassCastException  
プロパティ値が VARRAY 型でなかった場合

NullPointerException  
引数 propName に null を指定した場合

### 3.10.9 isNull (プロパティ値が NULL 値かどうかの判定)

#### (1) 機能

指定したプロパティの値が NULL 値かどうかを判定します。

指定したプロパティがない場合は、true が返却されます。

#### (2) 形式

```
boolean isNull(  
    String    propName  
)
```

#### (3) 引数

propName (入力)

プロパティ名を指定します。

#### (4) 戻り値

true

NULL 値です。または指定したプロパティがありません。

false

NULL 値以外です。

#### (5) 例外

NullPointerException

引数 propName に null を指定した場合

### 3.10.10 setNull ( NULL 値プロパティの設定 )

#### (1) 機能

指定したプロパティの値に NULL 値を設定します。

同一プロパティがある場合は上書きします。

#### (2) 形式

```
void setNull(  
    String    propName  
)
```

#### (3) 引数

propName ( 入力 )

プロパティ名を指定します。

#### (4) 戻り値

なし

#### (5) 例外

NullPointerException

引数 propName に null を指定した場合

### 3.10.11 setPropRef ( プロパティ値の参照の設定 )

#### (1) 機能

指定したプロパティの値の参照を設定します。

同一プロパティがある場合は上書きします。

#### (2) 形式

```
void setPropRef (
    String      propName,
    DbjVArray   ref
)
```

#### (3) 引数

propName ( 入力 )

プロパティ名を指定します。

ref ( 入力 )

プロパティ値を指定します。null を指定できます。

#### (4) 戻り値

なし

#### (5) 例外

NullPointerException

引数 propName に null を指定した場合

### 3.10.12 setPropVal ( プロパティ値の設定 )

#### (1) 機能

指定したプロパティの値を設定します。

同一プロパティがある場合は上書きします。

形式 2 では、プロパティ値が INT 型のオブジェクトにラップされて設定されます。

形式 4 では、指定した VARRAY 型の値がコピーされて設定されます。実行結果は次に示すメソッドの実行結果と同じです。

```
setVArrayRef(propName, (DbjVArray) val.clone())
```

ただし、setPropVal メソッドの形式 4 では、val に null を指定した場合、次に示すメソッドの実行結果と同じになります。

```
setVArrayRef(propName, null)
```

#### (2) 形式

##### (a) 形式 1

```
void setPropVal (
    String      propName,
    String      val
)
```

##### (b) 形式 2

```
void setPropVal (
    String      propName,
    int         val
)
```

##### (c) 形式 3

```
void setPropVal (
    String      propName,
    Integer     val
)
```

##### (d) 形式 4

```
void setPropVal (
    String      propName,
    DbjVArray   val
)
```

#### (3) 引数

propName (入力)

プロパティ名を指定します。

val (入力)

プロパティ値を指定します。形式 1、形式 3 または形式 4 の場合は、null を指定するとプロパティに NULL 値を設定できます。

### 3. パラメタクラス詳細

#### (4) 戻り値

なし

#### (5) 例外

NullPointerException

引数 propName に null を指定した場合



## 3.11 DbjPublicACLIdeIm インターフェース

DbjPublicACLIdeIm インターフェースは、文書空間オブジェクトにバインドするパブリック ACL の OIID 文字列を扱うインターフェースです。パブリック ACL の OIID 文字列を取得、設定するメソッドを提供します。

なお、Java クラスライブラリでは、文書空間オブジェクトにバインドするパブリック ACL の OIID 文字列のリストを、DbjVArray インターフェースで操作できます。また、その要素であるパブリック ACL の OIID 文字列は DbjPropSet インターフェースでも操作できます。

### プロパティ一覧

DbjPublicACLIdeIm インターフェースで扱うプロパティの一覧と、各プロパティに対する getter メソッドおよび setter メソッドを表 3-6 に示します。

表 3-6 DbjPublicACLIdeIm インターフェースで扱うプロパティ

プロパティ名	データ型	getter メソッド	setter メソッド
Id	String 型	getId	setId
propSet	DbjPropSet	propSet	setPropSet

### プロパティ詳細

- Id プロパティ

パブリック ACL の OIID 文字列を表します。

このプロパティへのアクセスは、プロパティ値集合の dbrProp\_PublicACLIdeIm プロパティに反映されます。

初期値は null です。

- propSet プロパティ

DbjPublicACLIdeIm インターフェースで扱うプロパティ値の集合です。

このプロパティ値集合に対する操作は、集合に含まれる各プロパティ値に反映されます。また、各プロパティ値に対する操作もこのプロパティ値集合に反映されます。

以降、DbjPublicACLIdeIm インターフェースのメソッドについて説明します。

### 3.11.1 getId (パブリック ACL の OIID 文字列の取得)

#### (1) 機能

パブリック ACL の OIID 文字列を取得します。

id プロパティに値が設定されていない場合は、null が返却されます。

#### (2) 形式

```
String getId()
```

#### (3) 引数

なし

#### (4) 戻り値

Id プロパティ

#### (5) 例外

なし

### 3.11.2 propSet (パブリック ACL の OIID 文字列のプロパティ値集合の取得)

#### (1) 機能

パブリック ACL の OIID 文字列を要素としたプロパティ値集合を取得します。

取得するプロパティ値集合の各要素は、プロパティ値の参照です。なお、値が設定されていなくても、null は返却されません。

#### (2) 形式

```
DbjPropSet propSet()
```

#### (3) 引数

なし

#### (4) 戻り値

propSet プロパティ (DbjPropSet インターフェース)

#### (5) 例外

なし

### 3.11.3 setId (パブリック ACL の OIID 文字列の設定)

#### (1) 機能

パブリック ACL の OIID 文字列を設定します。

#### (2) 形式

```
void setId(  
    String    id  
)
```

#### (3) 引数

id (入力)

パブリック ACL の OIID 文字列を指定します。null を指定できます。

#### (4) 戻り値

なし

#### (5) 例外

なし

### 3.11.4 setPropSet (パブリック ACL の OIID 文字列のプロパティ値集合の設定)

#### (1) 機能

パブリック ACL の OIID 文字列を要素としたプロパティ値集合に、指定したプロパティ値集合の内容をコピーして設定します。

#### (2) 形式

```
void setPropSet (
    DbjPropSet    propSet
)
```

#### (3) 引数

propSet (入力)

パブリック ACL の OIID 文字列のプロパティ値集合を指定します。

#### (4) 戻り値

なし

#### (5) 例外

NullPointerException

引数 propSet に null を指定した場合

## 3.12 DbjQParam インターフェース

---

DbjQParam インターフェースは、検索メソッドで指定する?パラメタリストの要素となるインターフェースのスーパーインターフェースです。

直接のスーパーインターフェース

- `java.lang.Cloneable`

### 3.12.1 getVal (? パラメタ値の取得)

#### (1) 機能

オブジェクトに設定されている ? パラメタの値を Object 型で取得します。

? パラメタの値が INT 型の場合は, INT 型オブジェクトの値が返却されます。値が設定されていない場合, または NULL 値の場合は, null が返却されます。

#### (2) 形式

```
Object getVal()
```

#### (3) 引数

なし

#### (4) 戻り値

? パラメタ値 (Object 型)

#### (5) 例外

なし

## 3.13 DbjReferenceContentInfo インターフェース

DbjReferenceContentInfo インターフェースは、リファレンスファイル文書のコンテンツ情報を扱うインターフェースです。リファレンスファイル文書のコンテンツロケーションおよびリファレンス種別を取得するメソッドを提供します。

直接のスーパーインターフェース

- DbjContentInfo

プロパティ一覧

DbjReferenceContentInfo インターフェースで扱うプロパティ一覧と、各プロパティに対する getter メソッドを表 3-7 に示します。

表 3-7 DbjReferenceContentInfo インターフェースで扱うプロパティ

プロパティ名	データ型	getter メソッド
contentLocation	String 型	getContentLocation
referenceType	int 型	getReferenceType

プロパティ詳細

- contentLocation プロパティ  
コンテンツロケーションを表します。  
文書オブジェクトの dbrProp\_ContentLocation プロパティと同じ値です。
- referenceType プロパティ  
コンテンツのリファレンス種別を表します。  
文書オブジェクトの dbrProp\_ReferenceType プロパティと同じ値です。

以降、DbjReferenceContentInfo インターフェースのメソッドについて説明します。



### 3.13.1 getContentLocation (コンテンツロケーションの取得)

#### (1) 機能

コンテンツロケーションを取得します。

#### (2) 形式

```
String getContentLocation()
```

#### (3) 引数

なし

#### (4) 戻り値

contentLocation プロパティ

#### (5) 例外

なし

### 3.13.2 getReferenceType ( リファレンス種別の取得 )

(1) 機能

リファレンス種別を取得します。

(2) 形式

```
int getReferenceType()
```

(3) 引数

なし

(4) 戻り値

referenceType プロパティ

(5) 例外

なし

## 3.14 DbjReferencePathInfo インターフェース

DbjReferencePathInfo インターフェースは、リファレンスファイル文書のパス情報を扱うインターフェースです。リファレンスファイル文書のパス情報を取得、設定するメソッドを提供します。

### プロパティ一覧

DbjReferencePathInfo インターフェースで扱うプロパティの一覧と、各プロパティに対する getter メソッドおよび setter メソッドを表 3-8 に示します。

表 3-8 DbjReferencePathInfo インターフェースで扱うプロパティ

プロパティ名	データ型	getter メソッド	setter メソッド
contentOperateMode	int 型	getContentOperateMode	setContentOperateMode
deleteRootPath	String 型	getDeleteRootPath	setDeleteRootPath
entry	String 型	getEntry	setEntry
targetPath	String 型	getTargetPath	setTargetPath

### プロパティ詳細

- contentOperateMode プロパティ**  
 コンテンツのパス操作モードを表します。  
 初期値は DbjDef.OPERATEMODE\_NONE です。コンテンツロケーションをコンテンツの相対パスで管理する場合は、DbjDef.OPERATEMODE\_USER\_RELATIVE\_CONTENT を指定します。コンテンツがないオブジェクトを作成する場合は、DbjDef.OPERATEMODE\_NONE を指定します。
- deleteRootPath プロパティ**  
 リファレンスファイル文書を削除する場合に、コンテンツの削除と同時にコンテンツを格納したディレクトリを削除するときの、削除するディレクトリのルートパスを表します。  
 削除の対象となるディレクトリは、コンテンツロケーション (dbrProp\_ContentLocation プロパティ) で示されるパス上の格納ディレクトリから、deleteRootPath プロパティで指定したディレクトリの下位までの間にあるディレクトリです。最下位層のディレクトリから順に削除し、エラーとなった時点で処理を終了します。  
 deleteRootPath プロパティに指定したパスがコンテンツロケーションのパスと一致していない場合はエラーになります。  
 初期値は null です。null を指定した場合、登録されているコンテンツ、および DocumentBroker がコンテンツを管理するためのディレクトリが削除されます。  
 なお、deleteRootPath プロパティの指定は、コンテンツが登録されているオブジェクトに対してだけ有効になります。  
 また、deleteRootPath プロパティに指定できるのは、DbjSession#setReferencePath メソッドで指定したコンテンツ格納先ベースパスから、コンテンツ格納先ベースパスと targetPath プロパティで示されるコンテンツ格納先パスの間のパスです。コンテンツ格納先ベースパスより上位のパス、および DocumentBroker がコンテンツを管理するためのディレクトリのパスは指定できません。
- entry プロパティ**  
 登録するコンテンツのパスまたはダウンロード先のパスを表します。  
 初期値は null です。
- targetPath プロパティ**  
 サーバ上のコンテンツの格納先となるディレクトリのパス (コンテンツ格納先パス) を表します。  
 初期値は null です。

### 3. パラメタクラス詳細

コンテンツロケーション (dbrProp\_ContentLocation プロパティ) には, targetPath プロパティで指定したパス, DocumentBroker がコンテンツを管理するためのディレクトリのパス, および entry プロパティで指定したファイル名を結合した値が格納されます。このとき, 区切り文字には半角の「/」が設定されます。

以降, DbjReferencePathInfo インターフェースのメソッドについて説明します。

### 3.14.1 getContentOperateMode (コンテンツのパス操作モードの取得)

#### (1) 機能

コンテンツのパス操作モードを取得します。

#### (2) 形式

```
int getContentOperateMode()
```

#### (3) 引数

なし

#### (4) 戻り値

contentOperateMode プロパティ

#### (5) 例外

なし

### 3.14.2 getDeleteRootPath ( 削除するディレクトリのルートパスの取得 )

#### (1) 機能

リファレンスファイル文書を削除する場合に、コンテンツの削除と同時にコンテンツを格納したディレクトリを削除するときの、削除するディレクトリのルートパスを取得します。

#### (2) 形式

```
String getDeleteRootPath()
```

#### (3) 引数

なし

#### (4) 戻り値

deleteRootPath プロパティ

#### (5) 例外

なし

### 3.14.3 getEntry (登録するコンテンツのパスまたはダウンロード先のパスの取得)

#### (1) 機能

登録するコンテンツのパスまたはダウンロード先のパスを取得します。

#### (2) 形式

```
String getEntry()
```

#### (3) 引数

なし

#### (4) 戻り値

entry プロパティ

#### (5) 例外

なし

### 3.14.4 getTargetPath (コンテンツ格納先パスの取得)

(1) 機能

コンテンツ格納先パスを取得します。

(2) 形式

`String getTargetPath()`

(3) 引数

なし

(4) 戻り値

targetPath プロパティ

(5) 例外

なし



### 3.14.5 setContentOperateMode (コンテンツのパス操作モードの設定)

#### (1) 機能

コンテンツのパス操作モードを設定します。なお、指定した値の妥当性は、検証されません。

#### (2) 形式

```
void setContentOperateMode(  
    int      contentOperateMode  
)
```

#### (3) 引数

contentOperateMode (入力)

コンテンツのパス操作モードを指定します。次に示す定数のどちらかを指定できます。

- OPERATEMODE\_NONE
- OPERATEMODE\_USER\_RELATIVE\_CONTENT

#### (4) 戻り値

なし

#### (5) 例外

なし

### 3.14.6 setDeleteRootPath ( 削除するディレクトリのルートパスの設定 )

#### (1) 機能

リファレンスファイル文書を削除する場合に、コンテンツの削除と同時にコンテンツを格納したディレクトリを削除するときの、削除するディレクトリのルートパスを設定します。なお、指定した値の妥当性は、検証されません。

#### (2) 形式

```
void setDeleteRootPath(  
    String deleteRootPath  
)
```

#### (3) 引数

deleteRootPath ( 入力 )

リファレンスファイル文書を削除する場合に、コンテンツの削除と同時にコンテンツを格納したディレクトリを削除するときの、削除するディレクトリのルートパスを指定します。

#### (4) 戻り値

なし

#### (5) 例外

なし

### 3.14.7 setEntry (登録するコンテンツのパスまたはダウンロード先のパスの設定)

#### (1) 機能

登録するコンテンツのパスまたはダウンロード先のパスを設定します。なお、指定した値の妥当性は、検証されません。

#### (2) 形式

```
void setEntry (
    String entry
)
```

#### (3) 引数

entry (入力)

登録するコンテンツのパスまたはダウンロード先のパスを指定します。

#### (4) 戻り値

なし

#### (5) 例外

なし

### 3.14.8 setTargetPath (コンテンツ格納先パスの設定)

#### (1) 機能

コンテンツ格納先パスを設定します。なお、指定した値の妥当性は、検証されません。

#### (2) 形式

```
void setTargetPath(  
    String targetPath  
)
```

#### (3) 引数

targetPath (入力)

コンテンツ格納先パスを指定します。

#### (4) 戻り値

なし

#### (5) 例外

なし

## 3.15 DbjReferenceUploadInfo インターフェース

DbjReferenceUploadInfo インターフェースは、リファレンスファイル文書のアップロード情報を扱うインターフェースです。リファレンスファイル文書のアップロード情報を取得、設定するメソッドを提供します。

直接のスーパーインターフェース

- DbjUploadInfo

プロパティ一覧

DbjReferenceUploadInfo インターフェースで扱うプロパティの一覧と、各プロパティに対する getter メソッドおよび setter メソッドを表 3-9 に示します。

表 3-9 DbjReferenceUploadInfo インターフェースで扱うプロパティ

プロパティ名	データ型	getter メソッド	setter メソッド
referencePathInfo	DbjReferencePathInfo	getReferencePathInfo	setReferencePathInfo

プロパティ詳細

- referencePathInfo プロパティ  
リファレンスファイル文書のパス情報を表します。

DbjUploadInfo インターフェースから継承するプロパティ詳細

- filePath プロパティ  
このインターフェースでは使用しません。設定した場合、設定値は無視されます。
- indexPath プロパティ  
登録するリファレンスファイル文書の全文検索インデックスを作成するファイルのフルパスを表します。
- renditionPropSet プロパティ  
登録するリファレンスファイル文書のレンディションプロパティ値集合です。
- renditionType プロパティ  
登録するリファレンスファイル文書のレンディションタイプを表します。
- retrievalName プロパティ  
登録するリファレンスファイル文書のクライアント側でのローカルファイル名を表します。

以降、DbjReferenceUploadInfo インターフェースのメソッドについて説明します。

### 3.15.1 getReferencePathInfo ( リファレンスファイル文書のパス情報の取得 )

(1) 機能

リファレンスファイル文書のパス情報を取得します。

(2) 形式

```
DbjReferencePathInfo getReferencePathInfo()
```

(3) 引数

なし

(4) 戻り値

referencePathInfo プロパティ

(5) 例外

なし

### 3.15.2 setReferencePathInfo (リファレンスファイル文書のパス情報の設定)

#### (1) 機能

リファレンスファイル文書のパス情報を設定します。なお、指定した値の妥当性は、検証されません。

#### (2) 形式

```
void setReferencePathInfo(  
                                DbjReferencePathInfo referencePathInfo  
)
```

#### (3) 引数

referencePathInfo (入力)

リファレンスファイル文書のパス情報を指定します。

#### (4) 戻り値

なし

#### (5) 例外

なし

## 3.16 DbjRenditionInfo インターフェース

DbjRenditionInfo インターフェースは、レンディション情報を扱うインターフェースです。レンディションタイプを取得するメソッドを提供します。

### プロパティ一覧

DbjRenditionInfo インターフェースで扱うプロパティの一覧と、各プロパティに対する getter メソッドを表 3-10 に示します。

表 3-10 DbjRenditionInfo インターフェースで扱うプロパティ

プロパティ名	データ型	getter メソッド
propSet	DbjPropSet	propSet
renditionType	String 型	getRenditionType

### プロパティ詳細

- propSet プロパティ  
オブジェクトに設定されているレンディションのプロパティ値集合です。
- renditionType プロパティ  
レンディションタイプを表します。

以降、DbjRenditionInfo インターフェースのメソッドについて説明します。



### 3.16.1 getRenditionType ( レンディションタイプの取得 )

#### (1) 機能

レンディションタイプを取得します。

#### (2) 形式

```
String getRenditionType()
```

#### (3) 引数

なし

#### (4) 戻り値

renditionType プロパティ

#### (5) 例外

なし

### 3.16.2 propSet ( レンディションのプロパティ値集合の取得 )

#### (1) 機能

レンディションのプロパティ値集合を取得します。

取得するプロパティ値集合の各要素は、プロパティ値の参照です。

#### (2) 形式

```
DbjPropSet propSet()
```

#### (3) 引数

なし

#### (4) 戻り値

propSet プロパティ ( DbjPropSet インターフェース )

#### (5) 例外

なし

## 3.17 DbjRenditionList インターフェース

---

DbjRenditionList インターフェースは、レンディション情報のリストを扱うインターフェースです。リストの要素であるレンディション情報 (DbjRenditionInfo インターフェース) を取得するメソッドを提供します。また、各レンディション情報からレンディションタイプのリストを取得するメソッドも提供します。

直接のスーパーインターフェース

- java.util.List

DbjRenditionList インターフェースは java.util.List インターフェースを継承します。リスト中の各要素は DbjRenditionInfo インターフェースです。

以降、DbjRenditionList インターフェースのメソッドについて説明します。

### 3.17.1 getRenditionInfo ( レンディション情報の取得 )

#### (1) 機能

指定した要素の DbjRenditionInfo インターフェースを取得します。

指定した要素は java.util.List#get メソッドでも取得できますが、この getRenditionInfo メソッドでは、ユーザアプリケーションプログラムがダウンキャストしないで要素を取得できます。

#### (2) 形式

```
DbjRenditionInfo getRenditionInfo(  
    int      index  
)
```

#### (3) 引数

index ( 入力 )

レンディション情報を取得する要素のインデックスを指定します。

0 から始まるインデックスで指定してください。

#### (4) 戻り値

指定した要素のレンディション情報 ( DbjRenditionInfo インターフェース )

#### (5) 例外

ClassCastException

要素が DbjRenditionInfo インターフェースでなかった場合

IndexOutOfBoundsException

インデックスが不正の場合

### 3.17.2 getRenditionTypeList ( レン디션タイプのリストの取得 )

#### (1) 機能

オブジェクトが保持するすべての要素のレン디션タイプを List 型で取得します。

リストが空の場合は、空リストが返却されます。

#### (2) 形式

```
List getRenditionTypeList ()
```

#### (3) 引数

なし

#### (4) 戻り値

レン디션タイプのリスト ( List 型 ( 各要素は String 型 ) )

#### (5) 例外

なし

## 3.18 DbjResultSet インターフェース

---

DbjResultSet インターフェースは、検索結果集合を扱うインターフェースです。検索結果集合の情報を扱うメソッド、指定した行または列のデータを取得するメソッド、カーソルを操作するメソッドなどを提供します。

検索結果集合とは、要素にプロパティ値を保持する行と列の2次元データに、メタデータを付けたものです。メタデータとは、列のプロパティ値のデータ型と列名の集合のことです。DbjResultSet インターフェースは、検索結果集合の2次元データを、各行を要素とするリストとして扱います。また、この各行データを、各列の値を要素とするリストとして扱います。

列名をメタデータに含む検索結果集合を名前付き検索結果といい、列名をメタデータに含まない検索結果集合を名前なし検索結果といいます。ユーザアプリケーションプログラムで実行する検索メソッドの形式によって、取得する検索結果集合が名前付きか名前なしかが決まります。名前付き検索結果は、検索結果集合をプロパティ値集合にマッピングする際に利用できます。また、列名を指定して名前が重複する列を削除したり、列名の付いていない列だけを削除したりできます。

検索結果集合に対するデータのアクセスには、カーソルを使用します。このカーソルの初期位置は、検索結果集合の先頭行の前です。したがって、ある行のデータを取得する際には、まず DbjResultSet#next メソッドなどを実行して、カーソルを移動する必要があります。

直接のスーパーインターフェース

- java.util.List

以降、DbjResultSet インターフェースのメソッドについて説明します。

### 3.18.1 absolute (カーソルの絶対指定行への移動)

#### (1) 機能

検索結果集合のカーソルを絶対指定行に移動します。

#### (2) 形式

```
boolean absolute(  
    int      rowIndex  
)
```

#### (3) 引数

rowIndex (入力)

カーソルの移動先を行インデクスで指定します。

指定できる範囲は、-1 ~ n (n は行数) です。-1 は先頭行の前を表し、n は最終行の後ろを表します。

つまり、行インデクスは次のように表せます。

- -1 : 先頭行の前
- 0 : 先頭行
- n-1 : 最終行
- n : 最終行の後ろ

#### (4) 戻り値

true

移動後が有効行です。

false

移動後が有効行ではありません。カーソルは移動していません。

#### (5) 例外

IndexOutOfBoundsException

行インデクスが範囲を超えた場合

### 3.18.2 afterLast (カーソルの最終行の後ろへの移動)

(1) 機能

検索結果集合のカーソルを最終行の後ろに移動します。

(2) 形式

```
void afterLast()
```

(3) 引数

なし

(4) 戻り値

なし

(5) 例外

なし



### 3.18.3 beforeFirst (カーソルの先頭行の前への移動)

#### (1) 機能

検索結果集合のカーソルを先頭行の前に移動します。

#### (2) 形式

```
void beforeFirst()
```

#### (3) 引数

なし

#### (4) 戻り値

なし

#### (5) 例外

なし

### 3.18.4 distinct (列の重複の排除)

#### (1) 機能

名前付き検索結果から、列名が重複している列を排除します。

なお、列名が重複している列のうち、どの列が残るかは保証できません。

#### (2) 形式

```
void distinct()
```

#### (3) 引数

なし

#### (4) 戻り値

なし

#### (5) 例外

`IllegalStateException`

名前なし検索結果に対してメソッドを実行した場合

### 3.18.5 findColumnName (列インデクスの取得)

#### (1) 機能

検索結果集合から、指定した列名の列インデクスを取得します。

指定した列名が検索結果集合にない場合は、-1 が返却されます。

なお、指定した列名が重複している場合、どの列インデクスを取得するかは保証できません。

#### (2) 形式

```
int findColumnName(  
    String columnName  
)
```

#### (3) 引数

columnName (入力)

列名を指定します。null を指定すると、列名が付いていない列のインデクスを取得できます。

#### (4) 戻り値

列インデクス

#### (5) 例外

なし

### 3.18.6 first (カーソルの先頭行への移動)

#### (1) 機能

検索結果集合のカーソルを先頭行に移動します。

#### (2) 形式

```
boolean first()
```

#### (3) 引数

なし

#### (4) 戻り値

true

移動後が先頭行です。

false

検索結果の件数が 0 件のため、先頭行がありません。カーソルは移動していません。

#### (5) 例外

なし

### 3.18.7 getColumnCount ( 列数の取得 )

#### (1) 機能

検索結果集合が保持する列数を取得します。

検索結果の件数が 0 件の場合は、0 が返却されます。

#### (2) 形式

```
int getColumnCount()
```

#### (3) 引数

なし

#### (4) 戻り値

列数

#### (5) 例外

なし

### 3.18.8 getColumnMetaName (列名の取得 (表名を含む))

#### (1) 機能

名前付き検索結果から、指定した列インデクスの列名を取得します。取得する列名は表名を含みます。

名前なし検索結果の場合は、null が返却されます。検索結果の件数が 0 件の場合は、メタデータも空となるため、例外がスローされます。

#### (2) 形式

```
String getColumnMetaName(  
    int    columnIndex  
)
```

#### (3) 引数

columnIndex (入力)

0 から始まる列インデクスを指定します。

#### (4) 戻り値

表名を含む列名

#### (5) 例外

IndexOutOfBoundsException

列インデクスが範囲を超えた場合

NullPointerException

検索結果の件数が 0 件の場合

### 3.18.9 getColumnName ( 列名の取得 )

#### (1) 機能

名前付き検索結果から、指定した列インデクスの列名を取得します。取得する列名は表名（または関連名）を含みません。

名前なし検索結果の場合は、null が返却されます。検索結果の件数が 0 件の場合は、メタデータも空となるため、例外がスローされます。

#### (2) 形式

```
String getColumnName (  
    int      columnIndex  
)
```

#### (3) 引数

columnIndex ( 入力 )

0 から始まる列インデクスを指定します。

#### (4) 戻り値

列名

#### (5) 例外

IndexOutOfBoundsException

列インデクスが範囲を超えた場合

NullPointerException

検索結果の件数が 0 件の場合

### 3.18.10 getColumnType ( 列のデータ型の取得 )

#### (1) 機能

検索結果集合から、指定した列インデクスの列のデータ型を取得します。

検索結果の件数が 0 件の場合は、メタデータも空となるため、例外がスローされます。

#### (2) 形式

```
int getColumnType(  
    int    columnIndex  
)
```

#### (3) 引数

columnIndex ( 入力 )

0 から始まる列インデクスを指定します。

#### (4) 戻り値

列のデータ型が、次に示す定数で返却されます。

- DbjDef.DATATYPE\_BOOL  
 BOOL 型
- DbjDef.DATATYPE\_INT  
 INT 型
- DbjDef.DATATYPE\_STR  
 STR 型
- DbjDef.DATATYPE\_VARRAY  
 VARRAY 型

#### (5) 例外

IndexOutOfBoundsException

列インデクスが範囲を超えた場合

NullPointerException

検索結果の件数が 0 件の場合



### 3.18.11 getColumnVals (列データの取得)

#### (1) 機能

検索結果集合から、指定した列データの全行をリストで取得します。取得した List オブジェクトは新規に作成され、List オブジェクトの各要素は列データ値のコピーとなります。

データを取得する列は、列インデクス (形式 1) または列名 (形式 2) で指定します。なお、列名で指定した場合で、その列名が重複しているときは、どの列データを取得するかは保証できません。

#### (2) 形式

##### (a) 形式 1

```
List getColumnVals(  
    int      columnIndex  
)
```

##### (b) 形式 2

```
List getColumnVals(  
    String   columnName  
)
```

#### (3) 引数

columnIndex (入力)

0 から始まる列インデクスを指定します。

columnName (入力)

列名を指定します。null を指定すると、列名が付いていない列のデータを取得できます。

#### (4) 戻り値

列データのリスト

#### (5) 例外

IllegalArgumentException

指定した列名がなかった場合

IndexOutOfBoundsException

列インデクスが範囲を超えた場合

### 3.18.12 getIntIntegerVal ( Integer 型での列データの取得 )

#### (1) 機能

検索結果集合のカーソル行から、指定した列データを Integer 型で取得します。

列値が NULL 値の場合は、null が返却されます。列値が INT 型でない場合は、例外がスローされます。

データを取得する列は、列インデクス (形式 1) または列名 (形式 2) で指定します。なお、列名で指定した場合で、その列名が重複しているときは、どの列データを取得するかは保証できません。

#### (2) 形式

##### (a) 形式 1

```
Integer getIntIntegerVal(  
    int      columnIndex  
)
```

##### (b) 形式 2

```
Integer getIntIntegerVal(  
    String   columnName  
)
```

#### (3) 引数

columnIndex (入力)

0 から始まる列インデクスを指定します。

columnName (入力)

列名を指定します。null を指定すると、列名が付いていない列の INT 型データを取得できます。

#### (4) 戻り値

列データ (Integer 型)

#### (5) 例外

ClassCastException

列値が INT 型でなかった場合

IllegalArgumentException

指定した列名がなかった場合

IllegalStateException

カーソル状態が不正の場合

IndexOutOfBoundsException

列インデクスが不正の場合

### 3.18.13 getIntVal ( int 型での列データの取得 )

#### (1) 機能

検索結果集合のカーソル行から、指定した列のデータを int 型で取得します。

列値が NULL 値の場合は、0 が返却されます。列値が INT 型でない場合は、java.lang.Object#toString メソッドの実行結果がさらに int 型の値に変換されて返却されます。列値が INT 型の場合は、そのまま java.lang.Integer#intValue メソッドの実行結果が返却されます。

データを取得する列は、列インデックス (形式 1) または列名 (形式 2) で指定します。なお、列名で指定した場合で、その列名が重複しているときは、どの列データを取得するかは保証できません。

#### (2) 形式

##### (a) 形式 1

```
int getIntVal(
    int    columnIndex
)
```

##### (b) 形式 2

```
int getIntVal(
    String columnName
)
```

#### (3) 引数

columnIndex (入力)

0 から始まる列インデックスを指定します。

columnName (入力)

列名を指定します。null を指定すると、列名が付いていない列のデータを int 型で取得できます。

#### (4) 戻り値

列データ (int 型)

#### (5) 例外

IllegalArgumentException

指定した列名がなかった場合

IllegalStateException

カーソル状態が不正の場合

IndexOutOfBoundsException

列インデックスが不正の場合

NumberFormatException

数値変換に失敗した場合

### 3.18.14 getObjectRef (データの参照の取得)

#### (1) 機能

検索結果集合のカーソル行から、指定した列の Object 型データの値の参照を取得します。列値が NULL 値の場合は、null が返却されます。

データの参照を取得する列は、列インデクス (形式 1) または列名 (形式 2) で指定します。なお、列名で指定した場合で、その列名が重複しているときは、どの列データの参照を取得するかは保証できません。

#### (2) 形式

##### (a) 形式 1

```
Object getObjectRef(  
    int      columnIndex  
)
```

##### (b) 形式 2

```
Object getObjectRef(  
    String   columnName  
)
```

#### (3) 引数

columnIndex (入力)

0 から始まる列インデクスを指定します。

columnName (入力)

列名を指定します。null を指定すると、列名が付いていない列のデータの参照を取得できます。

#### (4) 戻り値

データ (Object 型)

#### (5) 例外

IllegalArgumentException

指定した列名がなかった場合

IllegalStateException

カーソル状態が不正の場合

IndexOutOfBoundsException

列インデクスが不正の場合

### 3.18.15 getObjectVal (データの取得)

#### (1) 機能

検索結果集合のカーソル行から、指定した列の Object 型データの値のコピーを取得します。列値が NULL 値の場合は、null が返却されます。

データを取得する列は、列インデクス (形式 1) または列名 (形式 2) で指定します。なお、列名で指定した場合で、その列名が重複しているときは、どの列データを取得するかは保証できません。

#### (2) 形式

##### (a) 形式 1

```
Object getObjectVal(  
    int      columnIndex  
)
```

##### (b) 形式 2

```
Object getObjectVal(  
    String   columnName  
)
```

#### (3) 引数

columnIndex (入力)

0 から始まる列インデクスを指定します。

columnName (入力)

列名を指定します。null を指定すると、列名が付いていない列のデータ値のコピーを取得できます。

#### (4) 戻り値

データ (Object 型)

#### (5) 例外

IllegalArgumentException

指定した列名がなかった場合

IllegalStateException

カーソル状態が不正の場合

IndexOutOfBoundsException

列インデクスが不正の場合

### 3.18.16 getPropSet (行データをプロパティ値集合として取得)

#### (1) 機能

名前付き検索結果から、指定した行のデータをプロパティ値集合として取得します。取得したプロパティ値集合は新規に作成され、値がコピーされます。取得後のプロパティ値集合のプロパティ名は、列名の「.」(ピリオド)以前の文字列が削除されます。

なお、指定した行のデータのうち、列名が null の列のデータは取得しません。

形式 1, 3 はカーソル行が対象となり、形式 2, 4 はインデクスで行を指定します。

形式 1, 2 はすべての列が対象となりますが、形式 3, 4 は指定した表名 (または関連名) に関連する列名だけを対象とします。なお、形式 3, 4 では、列名が「表名.列名」の形式で設定されている必要があります。

#### (2) 形式

##### (a) 形式 1

```
DbjPropSet getPropSet()
```

##### (b) 形式 2

```
DbjPropSet getPropSet(  
    int      rowIndex  
)
```

##### (c) 形式 3

```
DbjPropSet getPropSet(  
    String   tableName  
)
```

##### (d) 形式 4

```
DbjPropSet getPropSet(  
    String   tableName,  
    int      rowIndex  
)
```

#### (3) 引数

rowIndex (入力)

0 から始まる行インデクスを指定します。

tableName (入力)

対象表名を指定します。null を指定すると、表指定なしとみなされます。

#### (4) 戻り値

行データ (DbjPropSet インターフェース)

#### (5) 例外

IllegalStateException

名前付き検索結果でない場合、またはカーソル状態が不正の場合

IndexOutOfBoundsException

行インデクスが不正の場合

### 3.18.17 getRow (カーソル行インデクスの取得)

#### (1) 機能

検索結果集合から、カーソル行のインデクスを取得します。

#### (2) 形式

```
int getRow()
```

#### (3) 引数

なし

#### (4) 戻り値

カーソル行のインデクス

#### (5) 例外

なし

### 3.18.18 getRowCount ( 行数の取得 )

(1) 機能

検索結果集合が保持する行数を取得します。

(2) 形式

```
int getRowCount ()
```

(3) 引数

なし

(4) 戻り値

行数

(5) 例外

なし



### 3.18.19 getRowVals ( 行データの取得 )

#### (1) 機能

検索結果集合から、指定した行のデータをリストで取得します。各要素は列データの値となります。取得したリストは新規に作成されます。

形式 1 はカーソル行のデータを取得し、形式 2 はインデクスで指定した行のデータを取得します。

#### (2) 形式

##### (a) 形式 1

```
List getRowVals()
```

##### (b) 形式 2

```
List getRowVals(  
    int      rowIndex  
)
```

#### (3) 引数

rowIndex ( 入力 )

0 から始まる行インデクスを指定します。

#### (4) 戻り値

行データのリスト

#### (5) 例外

IllegalStateException

カーソル状態が不正の場合

IndexOutOfBoundsException

行インデクスが不正の場合

## 3.18.20 getStringVal ( String 型での列データの取得 )

### (1) 機能

検索結果集合のカーソル行から、指定した列のデータを String 型で取得します。列データが NULL 値の場合は、null が返却されます。列データが STR 型でない場合は、java.lang.Object#toString メソッドを実行した結果が返却されます。

データを取得する列は、列インデクス (形式 1) または列名 (形式 2) で指定します。なお、列名で指定した場合で、その列名が重複しているときは、どの列データを取得するかは保証できません。

### (2) 形式

#### (a) 形式 1

```
String getStringVal (
    int      columnIndex
)
```

#### (b) 形式 2

```
String getStringVal (
    String   columnName
)
```

### (3) 引数

columnIndex (入力)

0 から始まる列インデクスを指定します。

columnName (入力)

列名を指定します。null を指定すると、列名が付いていない列のデータを String 型で取得できます。

### (4) 戻り値

列データ (String 型)

### (5) 例外

IllegalArgumentException

指定した列名がなかった場合

IllegalStateException

カーソル状態が不正の場合

IndexOutOfBoundsException

列インデクスが不正の場合

### 3.18.21 getTableName ( 表名の取得 )

#### (1) 機能

名前付き検索結果から、指定した列の表名 (または関連名) を取得します。

指定した列の列名が表名を含まない場合は、null が返却されます。検索結果の件数が 0 件の場合は、メタデータも空となるため、例外がスローされます。

#### (2) 形式

```
String getTableName (  
    int      columnIndex  
)
```

#### (3) 引数

columnIndex ( 入力 )

0 から始まる列インデックスを指定します。

#### (4) 戻り値

表名

#### (5) 例外

IllegalStateException

検索結果の件数が 0 件の場合

IndexOutOfBoundsException

列インデックスが範囲を超えた場合

## 3.18.22 getVArrayRef ( VARRAY 型でのデータの参照の取得 )

### (1) 機能

検索結果集合のカーソル行から、指定した列の VARRAY 型データの参照を取得します。列値が NULL 値の場合は、null が返却されます。列値が VARRAY 型でない場合は、例外がスローされます。

データの参照を取得する列は、列インデクス (形式 1) または列名 (形式 2) で指定します。なお、列名で指定した場合で、その列名が重複しているときは、どの列のデータの参照を取得するかは保証できません。

### (2) 形式

#### (a) 形式 1

```
DbjVArray getVArrayRef (
    int      columnIndex
)
```

#### (b) 形式 2

```
DbjVArray getVArrayRef (
    String   columnName
)
```

### (3) 引数

columnIndex (入力)

0 から始まる列インデクスを指定します。

columnName (入力)

列名を指定します。null を指定すると、列名が付いていない列の VARRAY 型データの参照を取得できます。

### (4) 戻り値

VARRAY 型データ (DbjVArray インターフェース)

### (5) 例外

ClassCastException

列値が VARRAY 型でなかった場合

IllegalArgumentException

指定した列名がなかった場合

IllegalStateException

カーソル状態が不正の場合

IndexOutOfBoundsException

列インデクスが不正の場合

### 3.18.23 getVArrayVal ( VARRAY 型でのデータの取得 )

#### (1) 機能

検索結果集合のカーソル行から、指定した列の VARRAY 型データのコピーを取得します。列データが NULL 値の場合は、null が返却されます。列データが VARRAY 型でない場合は、例外がスローされます。

データを取得する列は、列インデックス (形式 1) または列名 (形式 2) で指定します。なお、列名で指定した場合で、その列名が重複しているときは、どの列データを取得するかは保証できません。

#### (2) 形式

##### (a) 形式 1

```
DbjVArray getVArrayVal (
    int      columnIndex
)
```

##### (b) 形式 2

```
DbjVArray getVArrayVal (
    String   columnName
)
```

#### (3) 引数

columnIndex (入力)

0 から始まる列インデックスを指定します。

columnName (入力)

列名を指定します。null を指定すると、列名が付いていない列の VARRAY 型データのコピーを取得できません。

#### (4) 戻り値

VARRAY 型データ (DbjVArray インターフェース)

#### (5) 例外

ClassCastException

列値が VARRAY 型でなかった場合

IllegalArgumentException

指定した列名がなかった場合

IllegalStateException

カーソル状態が不正の場合

IndexOutOfBoundsException

列インデックスが不正の場合

### 3.18.24 isAfterLast (カーソルが最終行の後ろかどうかの判定)

#### (1) 機能

検索結果集合のカーソルが最終行の後ろかどうかを判定します。

#### (2) 形式

```
boolean isAfterLast()
```

#### (3) 引数

なし

#### (4) 戻り値

true

最終行の後ろです。または、検索結果の件数が0件のため最終行がありません。

false

最終行の後ろではありません。

#### (5) 例外

なし

### 3.18.25 isBeforeFirst (カーソルが先頭行の前かどうかの判定)

#### (1) 機能

検索結果集合のカーソルが先頭行の前かどうかを判定します。

#### (2) 形式

```
boolean isBeforeFirst()
```

#### (3) 引数

なし

#### (4) 戻り値

true

先頭行の前です。または、検索結果の件数が0件のため先頭行がありません。

false

先頭行の前ではありません。

#### (5) 例外

なし

### 3.18.26 isFirst (カーソルが先頭行かどうかの判定)

#### (1) 機能

検索結果集合のカーソルが先頭行かどうかを判定します。

#### (2) 形式

```
boolean isFirst()
```

#### (3) 引数

なし

#### (4) 戻り値

true

先頭行です。

false

先頭行ではありません。

#### (5) 例外

なし



### 3.18.27 isLast (カーソルが最終行かどうかの判定)

#### (1) 機能

検索結果集合のカーソルが最終行かどうかを判定します。

#### (2) 形式

```
boolean isLast()
```

#### (3) 引数

なし

#### (4) 戻り値

true

最終行です。

false

最終行ではありません。

#### (5) 例外

なし

### 3.18.28 isNamed (名前付き検索結果かどうかの判定)

#### (1) 機能

検索結果集合が名前付き検索結果かどうかを判定します。

#### (2) 形式

```
boolean isNamed()
```

#### (3) 引数

なし

#### (4) 戻り値

true

名前付き検索結果です。

false

名前付き検索結果ではありません。名前なし検索結果です。

#### (5) 例外

なし

### 3.18.29 isNull ( NULL 値かどうかの判定 )

#### (1) 機能

検索結果集合のカーソル行の、指定した列のデータが NULL 値かどうかを判定します。

NULL 値かどうか判定する列は、列インデクス (形式 1) または列名 (形式 2) で指定します。なお、列名で指定した場合で、その列名が重複しているときは、どの列データを判定するかは保証できません。

#### (2) 形式

##### (a) 形式 1

```
boolean isNull(
    int      columnIndex
)
```

##### (b) 形式 2

```
boolean isNull(
    String   columnName
)
```

#### (3) 引数

columnIndex (入力)

0 から始まる列インデクスを指定します。

columnName (入力)

列名を指定します。null を指定すると、列名が付いていない列のデータが NULL 値かどうかを判定します。

#### (4) 戻り値

true

NULL 値です。

false

NULL 値ではありません。

#### (5) 例外

IllegalArgumentException

指定した列名がなかった場合

IllegalStateException

カーソル状態が不正の場合

IndexOutOfBoundsException

列インデクスが不正の場合

### 3.18.30 last (カーソルの最終行への移動)

#### (1) 機能

検索結果集合のカーソルを最終行に移動します。

#### (2) 形式

```
boolean last()
```

#### (3) 引数

なし

#### (4) 戻り値

true

移動後が最終行です。

false

検索結果の件数が 0 件のため、最終行がありません。カーソルは移動していません。

#### (5) 例外

なし

### 3.18.31 next (カーソルの次の行への移動)

#### (1) 機能

検索結果集合のカーソルを次の行に移動します。

#### (2) 形式

```
boolean next()
```

#### (3) 引数

なし

#### (4) 戻り値

true

移動後、末尾 (EOF) を越えていません。有効行です。

false

移動後、末尾 (EOF) を越えました。カーソルは移動していません。

#### (5) 例外

なし

### 3.18.32 previous (カーソルの前の行への移動)

#### (1) 機能

検索結果集合のカーソルを前の行に移動します。

#### (2) 形式

```
boolean previous()
```

#### (3) 引数

なし

#### (4) 戻り値

true

移動後が有効行です。

false

移動後が有効行ではありません。カーソルは移動していません。

#### (5) 例外

なし

### 3.18.33 reduct (名前なし列の削除)

#### (1) 機能

名前付き検索結果の列から、名前の付いていない列 (列名が null の列) を削除します。

#### (2) 形式

```
void reduct()
```

#### (3) 引数

なし

#### (4) 戻り値

なし

#### (5) 例外

`IllegalStateException`

名前なし検索結果に対してメソッドを実行した場合

### 3.18.34 setColumnMetaName (列名の設定)

#### (1) 機能

検索結果集合の、指定した列の列名を設定します。列名には表名(または相関名)を含めてもかまいません。列名は、ほかの列の列名と重複してもかまいませんが、できるだけ重複しないように設定することをお勧めします。

なお、名前なし検索結果に対して一つでも列名を設定すると、検索結果集合は名前付き検索結果となります。また、一度名前付き検索結果となった検索結果集合は、すべての列名を null に設定しても、名前付き検索結果のままです。つまり、すべての列名が null である検索結果集合でも、名前付き検索結果の場合があります。

#### (2) 形式

```
void setColumnMetaName(  
    int         columnIndex,  
    String      columnName  
)
```

#### (3) 引数

columnIndex (入力)

0 から始まる列インデックスを指定します。

columnMetaName (入力)

列名を指定します。null を指定できます。

#### (4) 戻り値

なし

#### (5) 例外

IndexOutOfBoundsException

列インデックスが範囲を超えた場合



## 3.19 DbjSeedDocQParam インターフェース

---

DbjSeedDocQParam インターフェースは、種文章を表す?パラメタを扱うインターフェースです。

直接のスーパーインターフェース

- DbjQParam

## 3.20 DbjSetDCRLinkInfo インターフェース

---

DbjSetDCRLinkInfo インターフェースは、DbjObj#Link メソッドなどで設定する直接型リンクの設定情報を扱うインターフェースです。このインターフェースで扱うオブジェクトの linkType プロパティの値は、常に DbjDef.LINK\_DCR となります。

直接のスーパーインターフェース

- DbjSetLinkInfo

## 3.21 DbjSetLinkInfo インターフェース

DbjSetLinkInfo インターフェースは、DbjObj#Link メソッドなどで設定するリンク設定情報を扱うインターフェースです。リンク種別およびリンク先オブジェクトを取得、設定するメソッドを提供します。

直接のスーパーインターフェース

- java.lang.Cloneable

DbjSetLinkInfo インターフェースは、java.lang.Cloneable インターフェースを継承します。このインターフェースの clone メソッドは、ディープコピーを実行します。

プロパティ一覧

DbjSetLinkInfo インターフェースで扱うプロパティの一覧と、各プロパティに対する getter メソッドおよび setter メソッドを表 3-11 に示します。

表 3-11 DbjSetLinkInfo インターフェースで扱うプロパティ

プロパティ名	データ型	getter メソッド	setter メソッド
linkType	int 型	getLinkType	-
propSet	DbjPropSet	PropSet	setPropSet
targetObj	DbjObj	getTargetObj	setTargetObj

(凡例)

- : 該当するメソッドがないことを示します。

プロパティ詳細

- linkType プロパティ  
リンク種別を表します。
- propSet プロパティ  
リンクオブジェクトのプロパティ値集合です。
- targetObj プロパティ  
リンク先オブジェクトを表します。

以降、DbjSetLinkInfo インターフェースのメソッドについて説明します。

### 3.21.1 getLinkType (リンク種別の取得)

#### (1) 機能

リンク種別を取得します。

#### (2) 形式

```
int getLinkType()
```

#### (3) 引数

なし

#### (4) 戻り値

リンク種別が、次に示す定数で返却されます。

- DbjDef.LINK\_DCR  
直接型リンク
- DbjDef.LINK\_RCR  
参照型リンク
- DbjDef.LINK\_VTFIX  
構成管理型リンク (構成管理モードは FIX モード)
- DbjDef.LINK\_VTFLOAT  
構成管理型リンク (構成管理モードは FLOATING モード)
- DbjDef.LINK\_REL  
文書間リンク

#### (5) 例外

なし

### 3.21.2 getTargetObj (リンク先オブジェクトの取得)

#### (1) 機能

リンク先オブジェクトを取得します。

#### (2) 形式

```
DbjObj getTargetObj()
```

#### (3) 引数

なし

#### (4) 戻り値

targetObj プロパティ (DbjObj インターフェース)

#### (5) 例外

なし

### 3.21.3 propSet (リンクオブジェクトのプロパティ値集合の取得)

#### (1) 機能

リンクオブジェクトのプロパティ値集合を取得します。

取得するプロパティ値集合の各要素は、プロパティ値の参照です。なお、値が設定されていなくても、null は返却されません。

#### (2) 形式

```
DbjPropSet propSet()
```

#### (3) 引数

なし

#### (4) 戻り値

propSet プロパティ (DbjPropSet インターフェース)

#### (5) 例外

なし

### 3.21.4 setPropSet (リンクオブジェクトのプロパティ値集合の設定)

#### (1) 機能

リンクオブジェクトのプロパティ値を要素としたプロパティ値集合に、指定したプロパティ値集合の内容をコピーして設定します。

#### (2) 形式

```
void setPropSet (  
    DbjPropSet    propSet  
)
```

#### (3) 引数

propSet (入力)

設定するリンクのプロパティ値集合を指定します。null を指定できます。

#### (4) 戻り値

なし

#### (5) 例外

なし

### 3.21.5 setTargetObj (リンク先オブジェクトの設定)

#### (1) 機能

リンク先オブジェクトを設定します。

#### (2) 形式

```
void setTargetObj(  
    DbjObj      obj  
)
```

#### (3) 引数

obj (入力)

リンク先となるオブジェクトを指定します。null を指定できます。

#### (4) 戻り値

なし

#### (5) 例外

なし



## 3.22 DbjSetRCRLinkInfo インターフェース

---

DbjSetRCRLinkInfo インターフェースは、DbjObj#Link メソッドなどで設定する参照型リンクの設定情報を扱うインターフェースです。このインターフェースで扱うオブジェクトの linkType プロパティの値は、常に DbjDef.LINK\_RCR となります。

直接のスーパーインターフェース

- DbjSetLinkInfo

## 3.23 DbjSetRelLinkInfo インターフェース

---

DbjSetRelLinkInfo インターフェースは、DbjObj#Link メソッドなどで設定する文書間リンクの設定情報を扱うインターフェースです。このインターフェースで扱うオブジェクトの linkType プロパティの値は、常に DbjDef.LINK\_REL となります。

直接のスーパーインターフェース

- DbjSetLinkInfo

## 3.24 DbjSetVTFixLinkInfo インターフェース

---

DbjSetVTFixLinkInfo インターフェースは、DbjObj#Link メソッドなどで設定する構成管理型リンク（構成管理モードは FIX モード）の設定情報を扱うインターフェースです。このインターフェースで扱うオブジェクトの linkType プロパティの値は、常に DbjDef.LINK\_VTFIX となります。

直接のスーパーインターフェース

- DbjSetVTLinkInfo

## 3.25 DbjSetVTFloatLinkInfo インターフェース

---

DbjSetVTFloatLinkInfo インターフェースは、DbjObj#Link メソッドなどで設定する構成管理型リンク（構成管理モードは FLOATING モード）の設定情報を扱うインターフェースです。このインターフェースで扱うオブジェクトの linkType プロパティの値は、常に DbjDef.LINK\_VTFLOAT となります。

直接のスーパーインターフェース

- DbjSetVTLinkInfo

## 3.26 DbjSetVTLinkInfo インターフェース

---

DbjSetVTLinkInfo インターフェースは、構成管理型リンクの設定情報を扱うインターフェースのスーパーインターフェースです。

直接のスーパーインターフェース

- DbjSetLinkInfo

## 3.27 DbjStringQParam インターフェース

---

DbjStringQParam インターフェースは、STR 型を表す ? パラメタを扱うインターフェースです。

直接のスーパーインターフェース

- DbjQParam

## 3.28 DbjUploadInfo インターフェース

DbjUploadInfo インターフェースは、文書のアップロード情報を扱うインターフェースです。文書のアップロード情報を取得、設定するメソッドを提供します。

直接のスーパーインターフェース

- java.lang.Cloneable

DbjUploadInfo インターフェースは、java.lang.Cloneable インターフェースを継承します。このインターフェースの clone メソッドは、ディープコピーを実行します。

プロパティ一覧

DbjUploadInfo インターフェースで扱うプロパティの一覧と、各プロパティに対する getter メソッドおよび setter メソッドを表 3-12 に示します。

表 3-12 DbjUploadInfo インターフェースで扱うプロパティ

プロパティ名	データ型	getter メソッド	setter メソッド
filePath	String 型	getFilePath	setFilePath
indexPath	String 型	getIndexPath	setIndexPath
renditionPropSet	DbjPropSet	renditionPropSet	setRenditionPropSet
renditionType	String 型	getRenditionType	setRenditionType
retrievalName	String 型	getRetrievalName	setRetrievalName

プロパティ詳細

- filePath プロパティ

登録する文書のローカルパス名を表します。ファイル名を含むフルパスです。

次に示す場合には、このプロパティの値に null を指定できます。

- DocumentBroker Rendering Option のレンディション変換を利用してサブレンディションを登録する場合
- マルチレンディションを使用しない文書（サブレンディションを持たない、マスタレンディションだけの文書）を登録する場合

注

この場合、登録する文書から全文検索インデックスを作成しません。

初期値は null です。

- indexPath プロパティ

登録する文書の全文検索インデックスを作成するファイルのフルパスを表します。登録する文書がマスタレンディションでない場合は、このプロパティの値は無視されます。

filePath プロパティに指定したファイルから全文検索インデックスを作成する場合は、

DbjDef.INDEXPATH\_SAME を指定します。全文検索インデックスを作成しない場合は、null を指定します。

初期値は null です。

- renditionPropSet プロパティ

登録する文書のレンディションプロパティ値集合です。レンディションプロパティ値集合には、次のプロパティが指定できます。

- dbrProp\_RetrievalName（レンディションに登録された文書のファイル名）
- dbrProp\_RenditionStatus（レンディションの更新状態）

各レンディションプロパティの値に初期値を設定する場合は、null を指定します。

初期値は空プロパティ値集合です。

- renditionType プロパティ

登録する文書のレンディションタイプを表します。設定したレンディションタイプは登録文書オブジェクトの `dbrProp_RenditionType` プロパティに設定されます。

このプロパティ値に `null` を設定すると、`retrievalName` プロパティの値の拡張子に対応するレンディションタイプが設定されます。`retrievalName` プロパティの値も `null` の場合には、`filePath` プロパティの値の拡張子に対応するレンディションタイプが設定されます。この時、拡張子に対応するレンディションタイプは、レンディション定義ファイルの定義内容に従って取得されます。拡張子がレンディション定義ファイルに定義されていない場合は、レンディションタイプとして `null` が設定されます。なお、このプロパティ値に `null` を指定して、さらに `retrievalName` プロパティおよび `filePath` プロパティの値も `null` の場合には、レンディションタイプに `null` が設定されます。

初期値は `null` です。

- retrievalName プロパティ

登録する文書のクライアント側でのオリジナルファイル名を表します。設定したファイル名は登録文書オブジェクトの `dbrProp_RetrievalName` プロパティに設定されます。

`filePath` プロパティに指定したフルパス中のファイル名を設定したい場合は、`null` を指定します。

初期値は `null` です。

以降、`DbjUploadInfo` インターフェースのメソッドについて説明します。



### 3.28.1 getFilePath (登録する文書のフルパスの取得)

#### (1) 機能

登録する文書のフルパスを取得します。

#### (2) 形式

```
String getFilePath()
```

#### (3) 引数

なし

#### (4) 戻り値

filePath プロパティ

#### (5) 例外

なし

### 3.28.2 getIndexPath ( 全文検索インデクス作成用ファイルのフルパスの取得 )

(1) 機能

全文検索インデクス作成用ファイルのフルパスを取得します。

(2) 形式

`String getIndexPath()`

(3) 引数

なし

(4) 戻り値

indexPath プロパティ

(5) 例外

なし

### 3.28.3 getRenditionType (登録する文書のレンディションタイプの取得)

#### (1) 機能

登録する文書のレンディションタイプを取得します。

#### (2) 形式

```
String getRenditionType()
```

#### (3) 引数

なし

#### (4) 戻り値

renditionType プロパティ

#### (5) 例外

なし

### 3.28.4 getRetrievalName ( 登録する文書のファイル名の取得 )

(1) 機能

登録する文書のファイル名を取得します。

(2) 形式

`String getRetrievalName()`

(3) 引数

なし

(4) 戻り値

retrievalName プロパティ

(5) 例外

なし

### 3.28.5 renditionPropSet ( レンディションプロパティ値集合の取得 )

#### (1) 機能

レンディションプロパティ値集合を取得します。

取得するプロパティ値集合の各要素は、プロパティ値の参照です。なお、値が設定されていなくても、null は返却されません。

#### (2) 形式

```
DbjPropSet renditionPropSet()
```

#### (3) 引数

なし

#### (4) 戻り値

renditionPropSet プロパティ ( DbjPropSet インターフェース )

#### (5) 例外

なし

### 3.28.6 setFilePath (登録する文書のフルパスの設定)

#### (1) 機能

登録する文書のフルパスを設定します。

なお、指定した値の妥当性は検証されません。

#### (2) 形式

```
void setFilePath(  
    String    filePath  
)
```

#### (3) 引数

filePath (入力)

登録する文書のフルパスを指定します。

#### (4) 戻り値

なし

#### (5) 例外

なし

### 3.28.7 setIndexPath ( 全文検索インデクス作成用ファイルのフルパスの設定 )

#### (1) 機能

全文検索インデクス作成用ファイルのフルパスを設定します。

なお、指定した値の妥当性は検証されません。

#### (2) 形式

```
void setIndexPath(  
    String      indexPath  
)
```

#### (3) 引数

indexPath ( 入力 )

全文検索インデクス作成用ファイルのフルパスを指定します。

filePath プロパティに指定したファイルから全文検索インデクスを作成したい場合は、

DbjDef.INDEXPATH\_SAME を指定します。

全文検索インデクスを作成しない場合は、null を指定します。

#### (4) 戻り値

なし

#### (5) 例外

なし

### 3.28.8 setRenditionPropSet ( レンディションプロパティ値集合の設定 )

#### (1) 機能

レンディションプロパティ値集合に、指定したプロパティ値集合の内容をコピーして設定します。

なお、指定した値の妥当性は検証されません。

#### (2) 形式

```
void setRenditionPropSet (  
    DbjPropSet      renditionPropSet  
)
```

#### (3) 引数

renditionPropSet ( 入力 )

設定するレンディションプロパティ値集合を指定します。

各レンディションプロパティの値に初期値を設定する場合は、null を指定します。

レンディションプロパティ値集合に指定できるプロパティを次に示します。

- dbrProp\_RetrievalName
- dbrProp\_RenditionStatus

#### (4) 戻り値

なし

#### (5) 例外

なし



### 3.28.9 setRenditionType (登録する文書のレンディションタイプの設定)

#### (1) 機能

登録する文書のレンディションタイプを設定します。

なお、指定した値の妥当性は検証されません。

#### (2) 形式

```
void setRenditionType(  
    String    renditionType  
)
```

#### (3) 引数

renditionType (入力)

レンディションタイプを指定します。

このプロパティ値に null を設定すると、filePath プロパティの値の拡張子に対応するレンディションタイプが自動的に設定されます。

#### (4) 戻り値

なし

#### (5) 例外

なし

### 3.28.10 setRetrievalName (登録する文書のファイル名の設定)

#### (1) 機能

登録する文書のファイル名を設定します。

なお、指定した値の妥当性は検証されません。

#### (2) 形式

```
void setRetrievalName(  
    String      retrievalName  
)
```

#### (3) 引数

retrievalName (入力)

登録する文書のファイル名を指定します。

filePath プロパティに指定したフルパス中のファイル名を設定したい場合は、null を指定します。

#### (4) 戻り値

なし

#### (5) 例外

なし

## 3.29 DbjVArray インターフェース

DbjVArray インターフェースは、VARRAY 型プロパティの値である可変長配列オブジェクトを扱うインターフェースです。可変長配列オブジェクトの要素を扱うメソッド、可変長配列オブジェクトのプロパティを扱うメソッド、およびプロパティ名を変更するメソッドを提供します。

可変長配列オブジェクトは、プロパティ値集合を要素とするリストオブジェクトで、プロパティとしてメタプロパティを保持しています。可変長配列オブジェクトのメタプロパティとは、その可変長配列オブジェクトが保持するプロパティの、プロパティ名集合を表します。

可変長配列オブジェクトの任意の要素は、そのオブジェクトのメタプロパティが保持するプロパティを必ず保持します。例えば、メタプロパティがプロパティ "A", "B", "C" を保持する場合、任意の要素がプロパティ "A", "B", "C" を保持します。

また、可変長配列オブジェクトの要素は、メタプロパティが保持していないプロパティを要素ごとに保持できます。例えば、メタプロパティがプロパティ "A", "B", "C" を保持する場合、ある要素がプロパティ "A", "B", "C", "X" を保持し、同時に別の要素がプロパティ "A", "B", "C", "Y", "Z" を保持できます。

- メタプロパティとプロパティの整合性について  
前述のメタプロパティとプロパティとの整合性は、DbjVArray インターフェースを使用して可変長配列オブジェクトの要素（プロパティ値集合）を追加する場合に保証されます。  
しかし、java.util.List インターフェースを直接使用して要素を追加した場合は、単に要素が追加されるだけとなるため、整合性は保証されません。

直接のスーパーインターフェース

- java.util.List
  - java.lang.Cloneable
- DbjVArray インターフェースの clone メソッドは、ディープコピーを実行します。

コーディング例

```
// ユーザーアプリケーションプログラムのコーディング例
Set metaProps = new HashSet();
metaProps.add("mdmProp_Name");
metaProps.add("mdmProp_Number");

DbjVArray varray = dbjfactory.createVArray(metaProps);

DbjPropSet props = dbjfactory.createPropSet();
props.setPropVal("mdmProp_Name", "muraoka");
props.setPropVal("mdmProp_Number", 0);

// 1行目の追加("muraoka", 0)
varray.addPropSet(props);

props.setPropVal("mdmProp_Name", "tanaka");
props.setPropVal("mdmProp_Number", 8);

// 2行目の追加("tanaka", 8)
varray.addPropSet(props);

// 表示
System.out.println("1st display");
for(int i = 0; i < varray.size(); i ++ ) {
System.out.print("[ " + i + " ]");
System.out.print(varray.propSet(i).getStringVal("mdmProp_Name") + ", ");
System.out.println(varray.propSet(i).getIntVal("mdmProp_Number"));
}
```

### 3. パラメタクラス詳細

```
// 1行目のmdmProp_Numberプロパティ更新
props = varray.propSet(0);
props.setPropVal("mdmProp_Number", 7);

// 2行目のmdmPropNameプロパティ更新
varray.propSet(1).setPropVal("mdmProp_Name", "kanie");

// 表示
System.out.println("2nd display");
for(int i = 0; i < varray.size(); i++){
System.out.print "[" + i + " ");
System.out.print (varray.propSet(i).getStringVal("mdmProp_Name") + ", ");
System.out.println(varray.propSet(i).getIntVal("mdmProp_Number"));
}
}
```

#### コーディング例の実行結果

```
1st display
[0]muraoka, 0
[1]tanaka, 8
2nd display
[0]muraoka, 7
[1]kanie, 8
```

以降, DbjVArray インターフェースのメソッドについて説明します。

### 3.29.1 addPropSet ( プロパティ値集合 ( 行 ) の追加 )

#### (1) 機能

指定したプロパティ値集合を可変長配列オブジェクトの要素に追加します。

追加する際、メタプロパティに含まれているが指定したプロパティ値集合に含まれていないプロパティには、NULL 値が設定されます。一方、指定したプロパティ値集合に含まれているがメタプロパティに含まれていないプロパティは、値がそのまま追加されます。ただし、メタプロパティは変更されません。

#### (2) 形式

```
void addPropSet (  
    DbjPropSet    propSet  
)
```

#### (3) 引数

propSet ( 入力 )  
プロパティ値集合を指定します。

#### (4) 戻り値

なし

#### (5) 例外

NullPointerException  
引数 propSet に null を指定した場合

## 3.29.2 addPropVals ( プロパティ値 ( 列 ) の追加 )

### ( 1 ) 機能

可変長配列オブジェクトに、指定したリストの値をコピーして追加します。指定したプロパティ名がすでに存在していた場合は、そのプロパティの値を、指定したリストの値で上書きします。

指定したリストの要素が可変長配列の要素に対して不足していた場合は、リストの最後の要素が可変長配列オブジェクトの不足分に設定されます。

可変長配列オブジェクトが空の場合に限り、空のプロパティ値を追加できます。

メタプロパティは変更されません。

### ( 2 ) 形式

```
void addPropVals(  
    String      propName,  
    List        vals  
)
```

### ( 3 ) 引数

propName ( 入力 )

プロパティ名を指定します。

vals ( 入力 )

プロパティ値の List インターフェース ( 要素は可変長配列の各要素のプロパティ値 ) を指定します。

### ( 4 ) 戻り値

なし

### ( 5 ) 例外

IllegalArgumentException

可変長配列要素が空でないのに空リストを指定した場合

NullPointerException

引数 propName に null を指定した場合

### 3.29.3 changePropName ( 可変長配列オブジェクトのプロパティ名の変更 )

#### (1) 機能

プロパティ名を変更します。

変更後のプロパティ名が可変長配列オブジェクトにすでに存在していても、例外はスローされません。ただし、プロパティ名が変更前、変更後のどちらになるかは保証できません。

また、変更前として指定したプロパティ名が可変長配列オブジェクトにない場合は、`false` が返却されます。

#### (2) 形式

```
boolean changePropName(  
    String    oldName,  
    String    newName  
)
```

#### (3) 引数

`oldName` ( 入力 )

変更前のプロパティ名を指定します。

`null` を指定した場合は、例外がスローされます。

`newName` ( 入力 )

変更後のプロパティ名を指定します。

`null` を指定した場合は、例外がスローされます。

#### (4) 戻り値

`true`

変更されました。

`false`

変更されませんでした。

#### (5) 例外

`NullPointerException`

引数 `oldName` または `newName` に `null` を指定した場合

### 3.29.4 changePropNames ( 可変長配列オブジェクトのプロパティ名の一括変更 )

#### (1) 機能

指定したマップの内容に従って、プロパティ名を一括変更します。マップは、要素のキーに変更前のプロパティ名を、要素の値に変更後のプロパティ名を設定します。

変更後のプロパティ名が可変長配列オブジェクトにすでに存在していても、例外はスローされません。ただし、プロパティ名が変更前、変更後のどちらになるかは保証できません。

また、変更前として指定したプロパティ名が可変長配列オブジェクトに存在しなくても、例外はスローされません。

#### (2) 形式

```
boolean changePropNames(  
    Map      changeMap  
)
```

#### (3) 引数

changeMap ( 入力 )

変更前と変更後のプロパティ名を設定したマップを指定します。

#### (4) 戻り値

true

一つ以上変更されました。

false

変更されませんでした。

#### (5) 例外

NullPointerException

指定したマップの任意のキーまたは値に null を指定した場合



### 3.29.5 getPropCount (メタプロパティ数の取得)

#### (1) 機能

可変長配列オブジェクトのメタプロパティ数を取得します。メタプロパティが設定されていない場合は、0が返却されます。

#### (2) 形式

```
int getPropCount ()
```

#### (3) 引数

なし

#### (4) 戻り値

メタプロパティ数

#### (5) 例外

なし

### 3.29.6 getPropNameSet (メタプロパティの取得)

#### (1) 機能

可変長配列オブジェクトが保持するメタプロパティを取得します。戻り値は新規に作成されて返却されま  
す。

メタプロパティが設定されていない場合は、空オブジェクトが返却されます。

#### (2) 形式

```
Set getPropNameSet()
```

#### (3) 引数

なし

#### (4) 戻り値

メタプロパティ (要素が String 型のプロパティ名集合)

#### (5) 例外

なし

### 3.29.7 getPropVals ( 可変長配列オブジェクトのプロパティ値のリストの取得 )

#### (1) 機能

可変長配列オブジェクトから、指定したプロパティの値のコピーをリストで取得し、そのインターフェースを取得します。

可変長配列オブジェクトの要素が空の場合は、空の List インターフェースを取得します。指定したプロパティ名がない場合は、null が返却されます。

#### (2) 形式

```
List getPropVals(  
    String      propName  
)
```

#### (3) 引数

propName ( 入力 )

プロパティ名を指定します。プロパティ名に null を指定した場合は、例外がスローされます。

#### (4) 戻り値

プロパティ値のリストまたは null

#### (5) 例外

NullPointerException

引数 propName に null を指定した場合

### 3.29.8 propSet (プロパティ値集合の取得)

#### (1) 機能

可変長配列オブジェクトから、指定した行のデータをプロパティ値集合として取得します。

#### (2) 形式

```
DbjPropSet propSet(  
    int     rowIndex  
)
```

#### (3) 引数

rowIndex (入力)

0 から始まる行インデックスを指定します。

指定できる範囲は、0 ~ n-1 (n は行数) です。

#### (4) 戻り値

プロパティ値集合の参照 (DbjPropSet インターフェース)

#### (5) 例外

IndexOutOfBoundsException

行インデックスが範囲を超えた場合

### 3.29.9 removeProp ( プロパティの削除 )

#### (1) 機能

可変長配列オブジェクトから、指定したプロパティを削除します。

指定したプロパティがメタプロパティに含まれている場合は、メタプロパティからも指定したプロパティを削除します。

#### (2) 形式

```
void removeProp(  
    String      propName  
)
```

#### (3) 引数

propName ( 入力 )  
プロパティ名を指定します。

#### (4) 戻り値

なし

#### (5) 例外

NullPointerException  
引数 propName に null を指定した場合

## 3.30 DbjXmlUploadInfo インターフェース

DbjXmlUploadInfo インターフェースは、XML 文書のアップロード情報を扱うインターフェースです。登録する XML 文書の情報および XML プロパティマッピングについての情報を取得、設定するメソッドなどを提供します。

XML 文書のアップロード情報は、次のメソッドの引数に指定できます。

- DbjObj#uploadContents メソッド
- DbjDocSpace#createDocument メソッド
- DbjDocSpace#createVrDocument メソッド

これらのメソッドを実行すると、filePath プロパティに指定したフルパスのコンテンツを基に、XML プロパティマッピング機能および XML インデクスデータ作成機能が実行されます。このとき、XML 文書の構文解析結果が parseMessage プロパティに設定されます。

なお、登録する XML 文書がサブレンディションの場合、このインターフェースで扱う情報は無視され、スーパーインターフェースである DbjUploadInfo インターフェースで扱う情報だけが有効となります。

また、XML 文書のアップロード情報は DbjObj#addRendition メソッドの引数にも指定できますが、この場合も、スーパーインターフェースである DbjUploadInfo インターフェースで扱う情報だけが有効となります。

直接のスーパーインターフェース

- DbjUploadInfo

プロパティ一覧

DbjXmlUploadInfo インターフェースで扱うプロパティの一覧と、各プロパティに対する getter メソッドおよび setter メソッドを表 3-13 に示します。

表 3-13 DbjXmlUploadInfo インターフェースで扱うプロパティ

プロパティ名	データ型	getter メソッド	setter メソッド
filterFilePath	String 型	getFilterFilePath	setFilterFilePath
indexType	int 型	getIndexType	setIndexType
mappingId	String 型	getMappingId	setMappingId
parseLevel	int 型	getParseLevel	setParseLevel
parseMessage	String 型	getParseMessage	clearParseMessage

注 構文解析結果メッセージをクリアするメソッドです。

プロパティ詳細

- filterFilePath プロパティ  
全文検索インデクス作成時に使用するフィルタリング定義ファイルのフルパスです。  
このプロパティ値に null を設定した場合で、かつデフォルトのファイルが存在しないときは、フィルタリング定義ファイルを使用しないで処理を実行します。  
初期値は null です。  
なお、このプロパティ値は、登録する XML 文書がマスタレンディションの場合にだけ有効です。
- indexType プロパティ  
XML 文書から作成する全文検索インデクスの種別です。

初期値は DbjDef.INDEXTYPE\_NOTHING (全文検索インデックスを作成しない) です。

なお、このプロパティ値は、登録する XML 文書がマスタレンディションの場合にだけ有効です。

- mappingId プロパティ  
XML プロパティマッピングのマッピング定義名です。  
XML プロパティマッピング機能を実行しない場合は、null を指定します。  
初期値は null です。
- parseLevel プロパティ  
XML 文書の構文解析レベルです。  
初期値は DbjDef.XMLPARSE\_NO\_EXTERNAL\_ENTITIES (DTD の解析をしないで外部エンティティを無視) です。
- parseMessage プロパティ  
XML 文書の構文解析結果メッセージが設定されます。解析エラーの場合は例外がスローされるので、このプロパティに設定されるメッセージはワーニング (警告) メッセージだけです。ワーニングがなかった場合、このプロパティ値は null となります。  
初期値は null です。

#### DbjUploadInfo インターフェースから継承するプロパティ詳細

- filePath プロパティ  
登録する XML 文書のローカルパス名を表します。ファイル名を含むフルパス名です。  
プロパティマッピング機能を実行しない場合、または全文検索インデックスを作成しない場合は、このプロパティ値に null を指定できます。
- indexPath プロパティ  
XML 文書のアップロード情報オブジェクト (DbjXmlUploadInfo オブジェクト) では、このプロパティは無効です。常に null が設定されます。
- renditionPropSet プロパティ  
登録する XML 文書のレンディションプロパティ値集合です。
- renditionType プロパティ  
登録する XML 文書のレンディションタイプを表します。
- retrievalName プロパティ  
登録する XML 文書のクライアント側でのオリジナルファイル名を表します。

以降、DbjXmlUploadInfo インターフェースのメソッドについて説明します。

### 3.30.1 clearParseMessage ( 構文解析結果メッセージのクリア )

#### (1) 機能

XML 文書の構文解析結果メッセージをクリアします。具体的には、parseMessage プロパティの値に null を設定します。

#### (2) 形式

```
void clearParseMessage()
```

#### (3) 引数

なし

#### (4) 戻り値

なし

#### (5) 例外

なし



### 3.30.2 getFilterFilePath (フィルタリング定義ファイルのフルパスの取得)

#### (1) 機能

全文検索インデクス作成時に使用するフィルタリング定義ファイルのフルパスを取得します。

#### (2) 形式

```
String getFilterFilePath()
```

#### (3) 引数

なし

#### (4) 戻り値

filterFilePath プロパティ

#### (5) 例外

なし

### 3.30.3 getIndexType ( 全文検索インデクスの種別の取得 )

#### (1) 機能

XML 文書から作成する全文検索インデクスの種別を取得します。

#### (2) 形式

```
int getIndexType()
```

#### (3) 引数

なし

#### (4) 戻り値

全文検索インデクスの種別が、次に示す定数で返却されます。

- DbjDef.INDEXTYPE\_NOTHING  
全文検索インデクスを作成しません。インデクスデータは出力されません。
- DbjDef.INDEXTYPE\_PLAIN  
構造指定検索なしの全文検索インデクスを作成します。プレーンテキスト形式のインデクスデータが出力されます。
- DbjDef.INDEXTYPE\_STRUCTURED  
構造指定検索用の全文検索インデクスを作成します。構造指定検索用のインデクスデータ (ESIS-B 形式のデータ) が出力されます。

#### (5) 例外

なし

### 3.30.4 getMappingId ( マッピング定義名の取得 )

#### (1) 機能

XML プロパティマッピングのマッピング定義名を取得します。

#### (2) 形式

```
String getMappingId()
```

#### (3) 引数

なし

#### (4) 戻り値

mappingId プロパティ

#### (5) 例外

なし

### 3.30.5 getParseLevel ( 構文解析レベルの取得 )

#### (1) 機能

XML 文書の構文解析レベルを取得します。

#### (2) 形式

```
int getParseLevel()
```

#### (3) 引数

なし

#### (4) 戻り値

構文解析レベルが、次に示す定数で返却されます。

- DbjDef.XMLPARSE\_NO\_EXTERNAL\_ENTITIES  
DTD の解析をしないで、外部エンティティは無視します。
- DbjDef.XMLPARSE\_WITH\_EXTERNAL\_ENTITIES  
DTD を含めた構文解析をして外部エンティティを読み込みますが、検証は実行しません。
- DbjDef.XMLPARSE\_VALIDATE  
DTD を使用して検証を実行します。

#### (5) 例外

なし

### 3.30.6 getParseMessage ( 構文解析結果メッセージの取得 )

#### (1) 機能

XML 文書の構文解析結果メッセージを取得します。

#### (2) 形式

```
String getParseMessage()
```

#### (3) 引数

なし

#### (4) 戻り値

parseMessage プロパティ

#### (5) 例外

なし

### 3.30.7 setFilterFilePath (フィルタリング定義ファイルのフルパスの設定)

#### (1) 機能

全文検索インデクス作成時に使用するフィルタリング定義ファイルのフルパスを設定します。

なお、指定した値の妥当性は検証されません。

#### (2) 形式

```
void setFilterFilePath(  
    String    filterFilePath  
)
```

#### (3) 引数

filterFilePath (入力)

フィルタリング定義ファイルのフルパスを指定します。

#### (4) 戻り値

なし

#### (5) 例外

なし

### 3.30.8 setIndexType ( 全文検索インデクスの種別の設定 )

#### (1) 機能

作成する全文検索インデクスの種別を設定します。

なお、指定した値の妥当性は検証されません。

#### (2) 形式

```
void setIndexType (  
    int      indexType  
)
```

#### (3) 引数

indexType ( 入力 )

全文検索インデクスの種別を指定します。次に示す定数のどれかを指定できます。

- DbjDef.INDEXTYPE\_NOTHING

全文検索インデクスを作成しません。インデクスデータは出力されません。

- DbjDef.INDEXTYPE\_PLAIN

構造指定検索なしの全文検索インデクスを作成します。プレーンテキスト形式のインデクスデータが出力されます。

- DbjDef.INDEXTYPE\_STRUCTURED

構造指定検索用の全文検索インデクスを作成します。構造指定検索用のインデクスデータ ( ESIS-B 形式のデータ ) が出力されます。

#### (4) 戻り値

なし

#### (5) 例外

なし

### 3.30.9 setMappingId ( マッピング定義名の設定 )

#### (1) 機能

XML プロパティマッピングのマッピング定義名を設定します。

なお、指定した値の妥当性は検証されません。

#### (2) 形式

```
void setMappingId(  
    String mappingId  
)
```

#### (3) 引数

mappingId ( 入力 )

マッピング定義名を指定します。

XML プロパティマッピング機能を実行しない場合は、null を指定します。

#### (4) 戻り値

なし

#### (5) 例外

なし



### 3.30.10 setParseLevel ( 構文解析レベルの設定 )

#### (1) 機能

XML 文書の構文解析レベルを設定します。

なお、指定した値の妥当性は検証されません。

#### (2) 形式

```
void setParseLevel (  
    int      parseLevel  
)
```

#### (3) 引数

parseLevel ( 入力 )

構文解析のレベルを指定します。次に示す定数のどれかを指定できます。

- DbjDef.XMLPARSE\_NO\_EXTERNAL\_ENTITIES  
DTD の解析をしないで、外部エンティティは無視します。
- DbjDef.XMLPARSE\_WITH\_EXTERNAL\_ENTITIES  
DTD を含めた構文解析をして外部エンティティを読み込みますが、検証は実行しません。
- DbjDef.XMLPARSE\_VALIDATE  
DTD を使用して検証を実行します。

#### (4) 戻り値

なし

#### (5) 例外

なし



# 4

## 文書管理クラス詳細

この章では，文書管理クラスのインターフェース，およびメソッドについて説明します。

---

4.1 DbjDocSpace インターフェース

---

4.2 DbjLinkObj インターフェース

---

4.3 DbjLinkObjList インターフェース

---

4.4 DbjObj インターフェース

---

4.5 DbjObjList インターフェース

---

4.6 DbjSession インターフェース

---

4.7 DbjVerObj インターフェース

---

4.8 DbjVerObjList インターフェース

---

4.9 DbjXmlTranslator インターフェース

---

## 4.1 DbjDocSpace インターフェース

---

DbjDocSpace インターフェースは、文書空間へのアクセスを扱うインターフェースです。なお、個別の文書空間オブジェクトへのアクセスには DbjObj インターフェースを使用し、複数の文書空間オブジェクトへのアクセスには DbjObjList インターフェースを使用します。DbjDocSpace インターフェースは、特定の文書空間オブジェクトに依存しないメソッド、および新規に文書空間オブジェクトを生成するメソッドを提供します。

文書空間オブジェクトは DbjDocSpace インターフェースのメソッドで作成できます。個別の文書空間オブジェクトにアクセスする場合は、Proxy オブジェクトのインターフェースを DbjDocSpace#createObjConnection メソッドまたは DbjDocSpace#createObjList メソッドで取得し、そのインターフェースを使用してアクセスします。

DbjDocSpace インターフェースは、接続している文書空間のメタ情報にアクセスできます。この場合、DbjDocSpace#getMeta メソッドを使用して文書空間のメタ情報にアクセスするインターフェースを取得できます。また、DbjDocSpace インターフェースの DbjDocSpace#isAccessControlMode メソッドを利用することで文書空間がアクセス制御モードで動作しているかどうかを調べることができます。

以降、DbjDocSpace インターフェースのメソッドについて説明します。

### 4.1.1 changeSearchACLMode ( 検索実行時のアクセス制御モードの変更 )

#### (1) 機能

検索 ( DbjDocSpace#executeSearch メソッド ) の実行時のアクセス制御モードを変更します。アクセス制御モードの指定の有無によって、検索の実行時にアクセスできるオブジェクトが異なる場合があります。アクセス制御モードで動作している文書空間に対して検索を実行する場合、このメソッドを実行してアクセス制御モードなしの状態を設定して検索を実行できます。デフォルト値には、DbjDef.WITH\_ACL が設定されています。なお、引数に指定する値の妥当性は検証されません。

#### (2) 形式

```
void changeSearchACLMode(
    int          aclMode
)
```

#### (3) 引数

aclMode ( 入力 )

検索の実行時のアクセス制御モードを指定します。アクセス制御機能を使用している場合だけ有効です。次のどちらかの値を指定します。

- DbjDef.WITH\_ACL

アクセス制御機能付き検索モードで実行します。このモードの場合、ユーザはアクセス権がないオブジェクトを検索結果として取得できません。

- DbjDef.WITHOUT\_ACL

アクセス制御機能なし検索 ( アクセス権を無視した検索 ) モードで実行します。このモードの場合、ユーザはアクセス権がないオブジェクトでも検索結果として取得できます。

#### (4) 戻り値

なし

#### (5) 例外

DbjAccessControlNotSupportedException

アクセス制御モードで動作していない場合

DbjCORBAException

CORBA エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

## 4.1.2 createDocument (バージョンなし文書の作成)

### (1) 機能

文書空間にバージョンなし文書を作成して、指定したリンク元フォルダにリンクを設定します。作成したバージョンなし文書の文書空間オブジェクトアクセスインターフェース (DbjObj インターフェース) を取得します。バージョンなし文書を作成する場合にバージョンなし文書を構成するクラス名、およびバージョンなし文書のプロパティの初期値を指定できます。また、リファレンスファイル管理機能を使用したバージョンなし文書を作成する場合は、引数 uploadList に DbjReferenceUploadInfo インターフェースを指定します。

dbrProp\_RetrievalName (レンディションに登録されたコンテンツのファイル名) プロパティを指定する方法と指定の優先順位を次に示します。

表 4-1 dbrProp\_RetrievalName プロパティの指定方法と優先順位

dbrProp_RetrievalName プロパティの指定方法	優先順位
引数 uploadList が持つ要素の retrievalName プロパティに指定	1
引数 uploadList が持つ要素の renditionPropSet プロパティに dbrProp_RetrievalName を指定	2
引数 propSet に dbrProp_RetrievalName を指定 (マスタレンディションだけ)	3

上記のいずれも指定しなかった場合、引数 uploadList の要素が持つ filePath プロパティに含まれるファイル名を設定します。また、リファレンスファイル管理機能を使用している場合は、entry プロパティに含まれるファイル名を設定します。

### (2) 形式

```
DbjObj createDocument (
    String          className,
    DbjPropSet     propSet,
    List           uploadList,
    List           parentLinkList
)
```

### (3) 引数

#### className (入力)

作成するバージョンなし文書のクラス名を指定します。dmaClass\_DocVersion クラス、またはそのサブクラスのクラス名を指定できます。null を指定するとデフォルトクラスである dmaClass\_DocVersion クラスが仮定されます。

#### propSet (入力)

作成する文書のプロパティの初期値として設定するプロパティ値集合を指定します。プロパティの初期値を指定しない場合は null を指定します。

#### uploadList (入力)

要素は、DbjUploadInfo インターフェース、またはそのサブインターフェースです。作成する文書に登録する文書のアップロード情報のリストを指定します。複数の要素を指定するとマルチレンディションになります。アップロードするファイルを指定しない場合は null を指定します。

#### parentLinkList (入力)

要素は、DbjSetLinkInfo インターフェースのサブインターフェースです。作成する文書にリンクを設定するリンク元フォルダオブジェクト群およびリンク種別のリストを指定します。フォルダにリンク

を設定しない場合は null を指定します。

#### (4) 戻り値

作成した文書の文書空間オブジェクトアクセスインターフェース (DbjObj インターフェース)

#### (5) 例外

ClassCastException

引数 parentLinkList の要素が DbjSetLinkInfo インターフェースでなかった場合

DbjAccessControlException

アクセス権エラーの場合

DbjACEOperationException

指定 ACE についてのエラーの場合

DbjCORBAException

CORBA エラーの場合

DbjDBException

DB エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjFileReferenceOperationFailedException

リファレンスファイル管理機能を使用するサーバでのコンテンツ操作が失敗した場合

DbjIllegalObjectTypeException

指定オブジェクト種別が不適切の場合

DbjIllegalPropValException

指定プロパティ値が不正の場合

DbjIOException

Java クラスライブラリ固有の IO エラーの場合

DbjMasterRenditionNotSetException

マスタレンディションのレンディションタイプが設定されていなかった場合

DbjObjectNotFoundException

指定オブジェクトが存在しなかった場合

DbjPublicACLNotFoundException

指定パブリック ACL が存在しなかった場合

DbjPublicACLOutOfRangeException

指定パブリック ACL 数が範囲を超えた場合

DbjRenditionCountOutOfRangeException

指定レンディション数が範囲を超えた場合

DbjRenditionTypeDuplicatedException

指定レンディションタイプが重複していた場合

DbjSessionNotConnectException

#### 4. 文書管理クラス詳細

セッションが接続されていなかった場合

DbjTargetContentPathNotSetException

リファレンスファイル管理機能で、コンテンツ格納先ベースパスが設定されていなかった場合

DbjXmlParseException

XML 文書解析エラーの場合

IllegalArgumentException

引数の指定が不正の場合

NullPointerException

値が必要な引数が null の場合



### 4.1.3 createFolder (バージョンなしフォルダの作成)

#### (1) 機能

文書空間にバージョンなしフォルダを作成し、指定したリンク元フォルダにリンクを設定します。作成したバージョンなしフォルダの文書空間オブジェクトアクセスインターフェース (DbjObj インターフェース) を取得します。バージョンなしフォルダを作成する場合にバージョンなしフォルダを構成するクラス名、およびバージョンなしフォルダのプロパティの初期値を指定できます。

#### (2) 形式

```
DbjObj createFolder(
    String      className,
    DbjPropSet propSet,
    List        parentLinkList
)
```

#### (3) 引数

##### className (入力)

作成するバージョンなしフォルダのクラス名を指定します。edmClass\_ContainerVersion クラス、またはそのサブクラスのクラス名を指定できます。null を指定するとデフォルトクラスである edmClass\_ContainerVersion クラスが仮定されます。

##### propSet (入力)

作成するバージョンなしフォルダのプロパティの初期値として設定するプロパティ値集合を指定します。プロパティの初期値を指定しない場合は null を指定します。

##### parentLinkList (入力)

要素は、DbjSetLinkInfo インターフェースのサブインターフェースです。作成するバージョンなしフォルダにリンクを設定するリンク元フォルダオブジェクト群のリストを指定します。フォルダにリンクを設定しない場合は null を指定します。

#### (4) 戻り値

作成したバージョンなしフォルダの文書空間オブジェクトアクセスインターフェース (DbjObj インターフェース)

#### (5) 例外

##### ClassCastException

引数 parentLinkList の要素が DbjSetLinkInfo インターフェースでなかった場合

##### DbjAccessControlException

アクセス権エラーの場合

##### DbjACEOperationException

指定 ACE についてのエラーの場合

##### DbjCORBAException

CORBA エラーの場合

##### DbjDBException

DB エラーの場合

##### DbjException

#### 4. 文書管理クラス詳細

Java クラスライブラリ固有のエラーの場合

DbjIllegalObjectTypeException

指定オブジェクト種別が不適切の場合

DbjIllegalPropValException

指定プロパティ値が不正の場合

DbjObjectNotFoundException

指定オブジェクトが存在しなかった場合

DbjPublicACLNotFoundException

指定パブリック ACL が存在しなかった場合

DbjPublicACLOutOfRangeException

指定パブリック ACL 数が範囲を超えた場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

## 4.1.4 createIndependentData (独立データの作成)

### (1) 機能

文書空間に独立データを作成します。作成した独立データの文書空間オブジェクトアクセスインターフェース (DbjObj インターフェース) を取得します。独立データを作成する場合に独立データを構成するクラス名、および独立データのプロパティの初期値を指定できます。

### (2) 形式

```
DbjObj createIndependentData(
    String      className,
    DbjPropSet propSet
)
```

### (3) 引数

className (入力)

作成する独立データのクラス名を指定します。edmClass\_IndependentPersistence クラス、またはそのサブクラスのクラス名を指定できます。null を指定するとデフォルトクラスである edmClass\_IndependentPersistence クラスが仮定されます。

propSet (入力)

作成する独立データのプロパティの初期値として設定するプロパティ値集合を指定します。プロパティの初期値を指定しない場合は null を指定します。

### (4) 戻り値

作成した独立データの文書空間オブジェクトアクセスインターフェース (DbjObj インターフェース)

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjACEOperationException  
指定 ACE についてのエラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjIllegalObjectTypeException  
指定オブジェクト種別が不適切の場合

DbjIllegalPropValException  
指定プロパティ値が不正の場合

DbjObjectNotFoundException  
指定オブジェクトが存在しなかった場合

DbjPublicACLNotFoundException

#### 4. 文書管理クラス詳細

指定パブリック ACL が存在しなかった場合

DbjPublicACLOutOfRangeException

指定パブリック ACL 数が範囲を超えた場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

## 4.1.5 createLinkObjList ( 複数のリンクオブジェクトアクセスインターフェースの取得 )

### (1) 機能

複数のリンクオブジェクトにアクセスするインターフェース ( DbjLinkObjList インターフェース ) を取得します。このインターフェースを実装するオブジェクトは、要素を DbjLinkObj インターフェースとするリストオブジェクトです。ユーザアプリケーションプログラムは、このインターフェースを使用して、複数のリンクオブジェクトに対して一括してアクセスできます。

形式 1 では、要素数が 0 の空のリストオブジェクトである複数のリンクオブジェクトアクセスインターフェースを取得します。形式 2 では、要素が DbjLinkObj インターフェースであるコレクションインターフェースを引数 ( srcObjs ) に指定します。各要素 ( DbjLinkObj インターフェース ) は、リストオブジェクトの要素として設定されます。

### (2) 形式

#### (a) 形式 1

```
DbjLinkObjList createLinkObjList()
```

#### (b) 形式 2

```
DbjLinkObjList createLinkObjList(  
    Collection    srcObjs  
)
```

### (3) 引数

srcObjs ( 入力 )

要素が DbjLinkObj インターフェースのコレクションインターフェースを指定します。null を指定すると例外がスローされます。

### (4) 戻り値

複数のリンクオブジェクトアクセスインターフェース ( DbjLinkObjList インターフェース )

### (5) 例外

ClassCastException

引数 srcObjs の要素が DbjObj インターフェース以外の場合

NullPointerException

引数 srcObjs が null の場合

## 4.1.6 createObjConnection ( 文書空間オブジェクトアクセスインターフェースの取得 )

### (1) 機能

指定した文書空間オブジェクトにアクセスするインターフェース ( DbjObj インターフェース ) を取得します。アクセス対象の文書空間オブジェクトを OIID で指定します。なお、OIID の妥当性は検証されません。

このメソッドによって文書空間オブジェクトにアクセスするための Proxy オブジェクトを作成して、そのインターフェース ( DbjObj インターフェース ) を取得します。ユーザアプリケーションプログラムは、取得した DbjObj インターフェースを使用して対象の文書空間オブジェクトにアクセスします。ただし、このメソッドでは実際に文書空間オブジェクトに接続しません。

### (2) 形式

```
DbjObj createObjConnection(  
    String      oiid  
)
```

### (3) 引数

oiid ( 入力 )

アクセス対象の文書空間オブジェクトの OIID を指定します。null を指定すると例外がスローされません。

### (4) 戻り値

文書空間オブジェクトアクセスインターフェース ( DbjObj インターフェース )

### (5) 例外

NullPointerException

引数 oiid が null の場合

## 4.1.7 createObjList ( 複数の文書空間オブジェクトアクセスインターフェースの取得 )

### (1) 機能

複数の文書空間オブジェクトにアクセスするインターフェース ( DbjObjList インターフェース ) を取得します。このインターフェースを実装するオブジェクトは、要素を DbjObj インターフェースとするリストオブジェクトです。ユーザアプリケーションプログラムは、このインターフェースを使用して複数の文書空間オブジェクトに対して一括してアクセスできます。

形式 1 では、要素数が 0 の空のリストオブジェクトである複数文書空間オブジェクトにアクセスするインターフェースを取得します。取得後、ユーザが List#add メソッドによって要素を追加して使用します。

形式 2 では、要素が DbjObj インターフェースであるコレクションインターフェースを指定します。各要素 ( DbjObj インターフェース ) は、リストオブジェクトの要素として設定されます。

形式 3 および形式 4 では、リストオブジェクトの各要素を検索結果集合の各行からリストオブジェクトの要素にマッピングして複数文書空間オブジェクトアクセスインターフェースを取得します。形式 3 では、OID の入っている列を列インデクスで指定します。形式 4 では、OID の入っている列を列名で指定します。さらに、検索結果集合で名前の付いている列 ( 列名が null 以外のもの ) は、各要素の Proxy オブジェクトのプロパティとして設定されます。名前なし検索結果集合を指定すると、列名はすべて null になります。よって、要素の Proxy オブジェクトは、すべてのプロパティは保持しません。また、OID をプロパティとして指定した列は、対象外になります。形式 3 および形式 4 の tableName に null を指定すると、列名から表名 ( または関連名 ) 部分が除かれた値がプロパティ名として設定されます。tableName に null 以外を指定すると、指定した表名 ( または関連名 ) を含む列名のプロパティだけが対象になります。

### (2) 形式

#### (a) 形式 1

```
DbjObjList createObjList()
```

#### (b) 形式 2

```
DbjObjList createObjList(
    Collection      srcObjs
)
```

#### (c) 形式 3

```
DbjObjList createObjList(
    DbjResultSet   result,
    int            columnIndexOfOiid,
    String         tableName
)
```

#### (d) 形式 4

```
DbjObjList createObjList(
    DbjResultSet   result,
    String         columnNameOfOiid,
    String         tableName
)
```

### (3) 引数

srcObjs ( 入力 )

要素が DbjObj インターフェースのコレクションインターフェースを指定します。null を指定すると例外がスローされます。

#### 4. 文書管理クラス詳細

result (入力)

検索結果集合の DbjResultSet インターフェースを指定します。null を指定すると例外がスローされます。

columnIndexOfOiid (入力)

検索結果集合のうち、OIID を表す列インデックスを指定します。

columnNameOfOiid (入力)

検索結果集合のうち、OIID を表す列名を指定します。null を指定すると例外がスローされます。

tableName (入力)

表名 (または関連名) を指定します。

#### (4) 戻り値

複数の文書空間オブジェクトアクセスインターフェース (DbjObjList インターフェース)

#### (5) 例外

ClassCastException

引数 srcObjs の要素が DbjObj インターフェース以外の場合

IndexOutOfBoundsException

引数 columnIndexOfOiid が範囲外の場合

NullPointerException

引数 srcObjs , result , または columnNameOfOiid が null の場合



## 4.1.8 createPublicACL (パブリック ACL の作成)

### (1) 機能

文書空間にパブリック ACL を作成します。作成したパブリック ACL の文書空間オブジェクトアクセスインターフェース (DbjObj インターフェース) を取得します。パブリック ACL の作成時にパブリック ACL を構成するクラス名, およびパブリック ACL のプロパティの初期値を指定できます。また, パブリック ACL をバインドする文書空間オブジェクトも指定できます。

### (2) 形式

```
DbjObj createPublicACL(
    String          className,
    DbjPropSet     propSet,
    List           bindObjectList
)
```

### (3) 引数

#### className (入力)

作成するパブリック ACL のクラス名を指定します。edmClass\_PublicACL クラスを指定できます。null を指定するとデフォルトクラスである edmClass\_PublicACL クラスが仮定されます。

#### propSet (入力)

作成するパブリック ACL のプロパティの初期値として設定するプロパティ値集合を指定します。プロパティの初期値を指定しない場合は null を指定します。

#### bindObjectList (入力)

要素は DbjObj インターフェースです。作成するパブリック ACL にバインドする文書空間オブジェクトのリストを指定します。バインドする文書空間オブジェクトがない場合は null を指定します。

### (4) 戻り値

作成したパブリック ACL の文書空間オブジェクトアクセスインターフェース (DbjObj インターフェース)

### (5) 例外

#### ClassCastException

引数 bindObjectList の要素が DbjObj インターフェースでなかった場合

#### DbjAccessControlException

アクセス権エラーの場合

#### DbjACEOperationException

指定 ACE についてのエラーの場合

#### DbjCORBAException

CORBA エラーの場合

#### DbjDBException

DB エラーの場合

#### DbjException

Java クラスライブラリ固有のエラーの場合

#### 4. 文書管理クラス詳細

DbjIllegalObjectTypeException

指定オブジェクト種別が不適切の場合

DbjIllegalPropValException

指定プロパティ値が不正の場合

DbjObjectNotFoundException

指定オブジェクトが存在しなかった場合

DbjPublicACLAlreadyBoundException

指定パブリック ACL はすでにバインドされていた場合

DbjPublicACLNotFoundException

指定パブリック ACL は存在しなかった場合

DbjPublicACLOutOfRangeException

指定パブリック ACL 数が範囲を超えた場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

## 4.1.9 createVrDocument (バージョン付き文書の作成)

### (1) 機能

文書空間にバージョン付き文書を作成し、指定したリンク元フォルダにリンクを設定します。作成したバージョン付き文書の文書空間オブジェクトアクセスインターフェース (DbjObj インターフェース) を取得します。バージョン付き文書を作成する場合にバージョン付き文書を構成するクラス名、およびバージョン付き文書のプロパティの初期値を指定できます。また、リファレンスファイル管理機能を使用したバージョン付き文書オブジェクトを作成する場合は、引数 uploadList に DbjReferenceUploadInfo インターフェースを指定します。

dbrProp\_RetrievalName (レンディションに登録されたコンテンツのファイル名) プロパティを指定する方法と指定の優先順位を次に示します。

表 4-2 dbrProp\_RetrievalName プロパティの指定方法と優先順位

dbrProp_RetrievalName プロパティの指定方法	優先順位
引数 uploadList が持つ要素の retrievalName プロパティに指定	1
引数 uploadList が持つ要素の renditionPropSet プロパティの dbrProp_RetrievalName に指定	2
引数 propSet にバージョンングオブジェクトのプロパティとして dbrProp_RetrievalName を指定 (マスタレンディションだけ)	3
引数 propSet にバージョンオブジェクトのプロパティとして dbrProp_RetrievalName を指定 (マスタレンディションだけ)	4

上記のいずれも指定しなかった場合、引数 uploadList の要素が持つ filePath プロパティに含まれるファイル名を設定します。また、リファレンスファイル管理機能を使用している場合は、entry プロパティに含まれるファイル名を設定します。

### (2) 形式

```
DbjObj createVrDocument (
    String      classNameVersioning,
    String      classNameVersion,
    DbjPropSet  propSet,
    List        uploadList,
    List        parentLinkList
)
```

### (3) 引数

#### classNameVersioning (入力)

作成するバージョン付き文書のバージョンングオブジェクトのクラス名を指定します。dmaClass\_ConfigurationHistory クラス、またはそのサブクラスのクラス名を指定できます。null を指定するとデフォルトクラスである dmaClass\_ConfigurationHistory クラスが仮定されます。

#### classNameVersion (入力)

作成するバージョン付き文書のバージョンオブジェクトのクラス名を指定します。dmaClass\_DocVersion クラス、またはそのサブクラスのクラス名を指定できます。null を指定するとデフォルトクラスである edmClass\_VersionTracedDocVersion クラスが仮定されます。

#### propSet (入力)

作成するバージョン付き文書のプロパティの初期値として設定するプロパティ値集合を指定します。バージョンングオブジェクトとバージョンオブジェクト (カレントバージョン) のプロパティを同時に設定できます。プロパティの初期値を指定しない場合は null を指定します。

#### 4. 文書管理クラス詳細

##### uploadList (入力)

要素は、DbjUploadInfo インターフェイスまたはそのサブインターフェイスです。作成するバージョン付き文書に登録するファイル情報（アップロード情報）のリストを指定します。複数の要素を指定するとマルチレンディションになります。アップロードするファイルを指定しない場合は null を指定します。

##### parentLinkList (入力)

要素は、DbjSetLinkInfo インターフェイスのサブインターフェイスです。作成するバージョン付き文書にリンクを設定するリンク元フォルダオブジェクト群とリンク種別のリストを指定します。フォルダにリンクを設定しない場合は null を指定します。

#### (4) 戻り値

作成したバージョン付き文書の文書空間オブジェクトアクセスインターフェイス (DbjObj インターフェイス)

#### (5) 例外

##### ClassCastException

引数 parentLinkList の要素が DbjSetLinkInfo インターフェイスでなかった場合

##### DbjAccessControlException

アクセス権エラーの場合

##### DbjACEOperationException

指定 ACE についてのエラーの場合

##### DbjCORBAException

CORBA エラーの場合

##### DbjDBException

DB エラーの場合

##### DbjException

Java クラスライブラリ固有のエラーの場合

##### DbjFileReferenceOperationFailedException

リファレンスファイル管理機能を使用するサーバでのコンテンツ操作が失敗した場合

##### DbjIllegalObjectTypeException

指定オブジェクト種別が不適切の場合

##### DbjIllegalPropValException

指定プロパティ値が不正の場合

##### DbjIOException

Java クラスライブラリ固有の IO エラーの場合

##### DbjMasterRenditionNotSetException

マスタレンディションのレンディションタイプが設定されていなかった場合

##### DbjObjectNotFoundExpection

指定オブジェクトが存在しなかった場合

##### DbjPublicACLNotFoundExpection

指定パブリック ACL が存在しなかった場合

DbjPublicACLOutOfRangeException

指定パブリック ACL 数が範囲を超えた場合

DbjRenditionCountOutOfRangeException

指定レンディション数が範囲を超えた場合

DbjRenditionTypeDuplicatedException

指定レンディションタイプが重複していた場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

DbjTargetContentPathNotSetException

リファレンスファイル管理機能で、コンテンツ格納先ベースパスが設定されていなかった場合

DbjXmlParseException

XML 文書解析エラーの場合

IllegalArgumentException

引数の指定が不正の場合

NullPointerException

値が必要な引数が null の場合

## 4.1.10 createVrFolder (バージョン付きフォルダの作成)

### (1) 機能

文書空間にバージョン付きフォルダを作成し、指定したリンク元フォルダにリンクを設定します。作成したバージョン付きフォルダの文書空間オブジェクトアクセスインターフェース (DbjObj インターフェース) を取得します。バージョン付きフォルダを作成する場合にバージョン付きフォルダを構成するクラス名、およびバージョン付きフォルダのプロパティの初期値を指定できます。

### (2) 形式

```
DbjObj createVrFolder(
    String      classNameVersioning,
    String      classNameVersion,
    DbjPropSet  propSet,
    List        parentLinkList
)
```

### (3) 引数

#### classNameVersioning (入力)

作成するバージョン付きフォルダのバージョンングオブジェクトのクラス名を指定します。dmaClass\_ConfigurationHistory クラス、またはそのサブクラスのクラス名を指定できます。null を指定するとデフォルトクラスである dmaClass\_ConfigurationHistory クラスが仮定されます。

#### classNameVersion (入力)

作成するバージョン付きフォルダのバージョンオブジェクトのクラス名を指定します。edmClass\_ContainerVersion クラス、またはそのサブクラスのクラス名を指定できます。null を指定するとデフォルトクラスである edmClass\_ContainerVersion クラスが仮定されます。

#### propSet (入力)

作成するバージョン付きフォルダのプロパティの初期値として設定するプロパティ値集合を指定します。バージョンングオブジェクトとバージョンオブジェクト (カレントバージョン) のプロパティを同時に設定できます。プロパティの初期値を指定しない場合は null を指定します。

#### parentLinkList (入力)

要素は DbjSetLinkInfo インターフェースのサブインターフェースです。作成するバージョン付きフォルダにリンクを設定するリンク元フォルダオブジェクト群とリンク種別のリストを指定します。フォルダにリンクを設定しない場合は null を指定します。

### (4) 戻り値

作成したバージョン付きフォルダの文書空間オブジェクトアクセスインターフェース (DbjObj インターフェース)

### (5) 例外

#### ClassCastException

引数 parentLinkList の要素が DbjSetLinkInfo インターフェースでなかった場合

#### DbjAccessControlException

アクセス権エラーの場合

#### DbjACEOperationException

指定 ACE についてのエラーの場合

- DbjCORBAException  
CORBA エラーの場合
- DbjDBException  
DB エラーの場合
- DbjException  
Java クラスライブラリ固有のエラーの場合
- DbjIllegalObjectTypeException  
指定オブジェクト種別が不適切の場合
- DbjIllegalPropValException  
指定プロパティ値が不正の場合
- DbjObjectNotFoundException  
指定オブジェクトが存在しなかった場合
- DbjPublicACLNotFoundException  
指定パブリック ACL が存在しなかった場合
- DbjPublicACLOutOfRangeException  
指定パブリック ACL 数が範囲を超えた場合
- DbjSessionNotConnectException  
セッションが接続されていなかった場合

## 4.1.11 executeSearch ( 検索の実行 )

### (1) 機能

文書空間に対して edmSQL 文による検索を実行し、検索結果集合オブジェクトインターフェース ( DbjResultSet インターフェース ) を取得します。

形式 1 および形式 2 では、名前なし検索結果集合を取得します。

形式 3 および形式 4 では、名前付き検索結果集合を取得します。

形式 2 および形式 4 では、ロック種別を指定できます。形式 1 および形式 3 では、ロックなし指定 ( DbjDef.LOCK\_NONE 指定 ) に相当します。

形式 3 および形式 4 の引数 ( eqlStatement ) には、SELECT 句の項目部分に「\$\_」を使用して記述します。例えば、select \$\_ from mdmClass\_Folder\_c のように指定します。また、引数 ( selectItems ) に指定した要素は、「,」(コンマ) で区切られて「\$\_」に置き換えられます。検索結果集合の列名は、selectItems で指定した要素になります。SELECT 句の項目指定部分は、「\$\_」だけ使用できます。例えば、select A,\$\_ from ... と指定すると例外がスローされます。なお、形式 3 および形式 4 では、取得される列数と selectItems の要素数が不一致の場合、例外がスローされます。

### (2) 形式

#### (a) 形式 1

```
DbjResultSet executeSearch(
    String          eqlStatement,
    List           qParamList,
    DbjFetchInfo   fetchInfo
)
```

#### (b) 形式 2

```
DbjResultSet executeSearch(
    String          eqlStatement,
    List           qParamList,
    DbjFetchInfo   fetchInfo,
    int            lockType
)
```

#### (c) 形式 3

```
DbjResultSet executeSearch(
    List           selectItems,
    String         eqlStatement,
    List           qParamList,
    DbjFetchInfo   fetchInfo
)
```

#### (d) 形式 4

```
DbjResultSet executeSearch(
    List           selectItems,
    String         eqlStatement,
    List           qParamList,
    DbjFetchInfo   fetchInfo,
    int            lockType
)
```

### (3) 引数

selectItems ( 入力 )

要素のデータ型は String 型です。SELECT 句に指定する項目のリストを指定します。名前付き検索



結果集合の列名になります。null を指定すると例外がスローされます。

eqlStatement (入力)

edmSQL 文を指定します。null を指定すると例外がスローされます。

qParamList (入力)

? パラメタの値を表します。要素は、DbjQParam インターフェースのサブクラスのリストを指定します。NULL 値の場合は、要素の値に null を指定します。? パラメタの指定がない場合は、null を指定します。

fetchInfo (入出力)

検索結果取得情報の条件を指定します。全件を取得する場合、またはキャッシュ検索をしない場合は null を指定します。キャッシュ検索をすると、cacheKey プロパティと cacheTotal プロパティが設定されます。全件を取得しない場合、maxFetchCount プロパティに指定されている件数を取得した後に、comparator プロパティに指定したコンパレータによりソートされます。そのため、全件を取得する場合と、maxFetchCount プロパティを指定して検索した場合を比較すると、comparator プロパティに同じオブジェクトを指定しても結果が異なる場合があります。

lockType (入力)

検索結果集合に対して取得するロック種別を指定します。次に示す値のどれかを指定できます。

- DbjDef.LOCK\_NONE  
ロックなし
- DbjDef.LOCK\_READ  
read ロック
- DbjDef.LOCK\_WRITE  
write ロック
- DbjDef.LOCK\_READFORUPDATE  
READFORUPDATE ロック

#### (4) 戻り値

検索結果集合オブジェクトインターフェース (DbjResultSet インターフェース)

#### (5) 例外

DbjAccessControlNotSupportedException

アクセス制御モードに対応していない場合にアクセス制御モードで検索しようとした場合

DbjCORBAException

CORBA エラーの場合

DbjDBException

DB エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

DbjIllegalCacheStartIndexException

検索結果数より大きい値を引数 fetchInfo (DbjFetchInfo インターフェース) の startIndex プロパティに指定した場合

#### 4. 文書管理クラス詳細

IllegalArgumentException

引数が不正の場合

NullPointerException

引数 eqlStatement が null の場合

### 4.1.12 getMeta ( 文書空間のメタ情報の取得 )

#### (1) 機能

接続している文書空間のメタ情報を取得します。

#### (2) 形式

```
DbjMeta getMeta()
```

#### (3) 引数

なし

#### (4) 戻り値

メタ情報インターフェース ( DbjMeta インターフェース )

#### (5) 例外

なし

### 4.1.13 getSearchACLMode ( 検索実行時のアクセス制御モードの取得 )

#### (1) 機能

DbjDocSpace#executeSearch メソッドの実行時のアクセス制御モードを取得します。アクセス制御モードが動作している場合だけ有効です。

#### (2) 形式

```
int getSearchACLMode()
```

#### (3) 引数

なし

#### (4) 戻り値

DbjDocSpace#executeSearch メソッドの実行時のアクセス制御モードとして次に示す値が返却されます。

DbjDef.WITH\_ACL

アクセス制御機能付き検索を実行します。

DbjDef.WITHOUT\_ACL

アクセス制御機能なし検索 ( アクセス権を無視した検索 ) を実行します。

#### (5) 例外

DbjAccessControlNotSupportedException

アクセス制御モードで動作していない場合

DbjCORBAException

CORBA エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

#### 4.1.14 isAccessControlMode ( アクセス制御モードの取得 )

##### (1) 機能

文書空間がアクセス制御モードで実行されているかどうかについての情報を取得します。

##### (2) 形式

```
boolean isAccessControlMode()
```

##### (3) 引数

なし

##### (4) 戻り値

true

アクセス制御モードで実行しています。

false

アクセス制御モードで実行していません。

##### (5) 例外

DbjCORBAException

CORBA エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

## 4.1.15 removeObjects ( 検索条件を指定したオブジェクトの削除 )

### (1) 機能

文書空間に対して edmSQL 文による検索を実行し、検索結果として取得したオブジェクトを削除します。このメソッドは、検索結果集合の 1 列目を、削除対象のオブジェクトの OIID とみなします。したがって、指定する edmSQL 文の主問い合わせの SELECT 句の第一項目には、削除対象とする表の dmaProp\_OIID プロパティ ( または削除対象の OIID 文字列が設定されているプロパティ ) を必ず指定します。

### (2) 形式

```
void removeObjects(  
    String      eqlStatement,  
    List        qParamList  
)
```

### (3) 引数

eqlStatement ( 入力 )

edmSQL 文を指定します。null を指定すると例外がスローされます。

qParamList ( 入力 )

? パラメタの値を表します。要素には、DbjQParam インターフェースのサブクラスのリストを指定します。NULL 値の場合は、要素の値に null を指定します。また、? パラメタの指定がない場合は、null を指定します。

### (4) 戻り値

なし

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjAccessControlNotSupportedException  
アクセス制御モードに対応していない場合にアクセス制御モードで検索しようとした場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjLastVersionException  
最後のバージョンを消そうとした場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合

NullPointerException  
引数 eqlStatement が null の場合

## 4.2 DbjLinkObj インターフェース

---

DbjLinkObj インターフェースは、一つのリンクオブジェクトにアクセスするインターフェースです。  
以降、DbjLinkObj インターフェースのメソッドについて説明します。

## 4.2.1 changeToVTFix ( 構成管理型リンクの構成管理モードを FIX モードに変更 )

### (1) 機能

リンク Proxy オブジェクトのアクセス対象のターゲットオブジェクトであるリンクオブジェクトが指定した構成管理型リンクの構成管理モードを FIX モードに変更します。

形式 1 では、アクセス対象のリンクオブジェクトが指定した構成管理型リンクを指定したバージョンのバージョン識別子で固定します。バージョン識別子に null を指定するとカレントバージョンになります。

形式 2 では、すべての構成管理型リンクを、その時点の状態固定します。

### (2) 形式

#### (a) 形式 1

```
void changeToVTFix(  
    String      fixVersionId  
)
```

#### (b) 形式 2

```
void changeToVTFix()
```

### (3) 引数

fixVersionId ( 入力 )

構成管理型リンクによって固定するバージョンのバージョン識別子を指定します。null を指定するとカレントバージョンになります。

### (4) 戻り値

なし

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjObjectNotFoundException  
指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合



## 4.2.2 changeToVTFloat ( 構成管理型リンクの構成管理モードを FLOATING モードに変更 )

### (1) 機能

リンク Proxy オブジェクトのアクセス対象のターゲットオブジェクトであるリンクオブジェクトが指定した構成管理型リンクの構成管理モードを FLOATING モードに変更します。

### (2) 形式

```
void changeToVTFloat ()
```

### (3) 引数

なし

### (4) 戻り値

なし

### (5) 例外

DbjAccessControlException

アクセス権エラーの場合

DbjCORBAException

CORBA エラーの場合

DbjDBException

DB エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjObjectNotFoundException

指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

### 4.2.3 getLinkId (リンク識別子の取得)

#### (1) 機能

リンク Proxy オブジェクトのアクセス対象のターゲットオブジェクトであるリンクオブジェクトのリンク識別子を取得します。

#### (2) 形式

```
String getLinkId()
```

#### (3) 引数

なし

#### (4) 戻り値

リンク識別子

#### (5) 例外

なし

## 4.2.4 getLinkType (リンク種別の取得)

### (1) 機能

リンク Proxy オブジェクトのアクセス対象のターゲットオブジェクトであるリンクオブジェクトのリンク種別を取得します。

### (2) 形式

```
int getLinkType()
```

### (3) 引数

なし

### (4) 戻り値

リンク種別

### (5) 例外

なし

## 4.2.5 getOwnerObj (リンク元オブジェクトの取得)

### (1) 機能

リンク Proxy オブジェクトのアクセス対象のターゲットオブジェクトであるリンクオブジェクトのリンク元オブジェクトを取得します。取得するのは、リンク元オブジェクトの Proxy オブジェクトのインターフェース (DbjObj インターフェース) です。

### (2) 形式

```
DbjObj getOwnerObj()
```

### (3) 引数

なし

### (4) 戻り値

リンク元オブジェクトのインターフェース (DbjObj インターフェース)

### (5) 例外

なし

## 4.2.6 getTargetObj (リンク先オブジェクトの取得)

### (1) 機能

リンク Proxy オブジェクトのアクセス対象のターゲットオブジェクトであるリンクオブジェクトのリンク先オブジェクトを取得します。取得するのは、リンク先オブジェクトの Proxy オブジェクトのインターフェース (DbjObj インターフェース) です。

### (2) 形式

```
DbjObj getTargetObj ()
```

### (3) 引数

なし

### (4) 戻り値

リンク先オブジェクトのインターフェース (DbjObj インターフェース)

### (5) 例外

なし

## 4.2.7 propSet (リンク Proxy オブジェクトのプロパティ値集合インターフェースの取得)

### (1) 機能

リンク Proxy オブジェクトのプロパティ値集合インターフェースを取得します。プロパティ値集合インターフェースは、参照権限で取得します。取得したインターフェースを使用して値を設定すると、Proxy オブジェクトのリンクプロパティ値も更新されます。なお、プロパティ値集合が設定されていなくても null は返却されません。

### (2) 形式

```
DbjPropSet propSet()
```

### (3) 引数

なし

### (4) 戻り値

リンク Proxy オブジェクトのプロパティ値集合インターフェース (DbjPropSet インターフェース)

### (5) 例外

なし

## 4.2.8 readProperties (リンクプロパティの取得)

### (1) 機能

リンク Proxy オブジェクトのアクセス対象のターゲットオブジェクトであるリンクオブジェクトの指定プロパティを、各要素のリンク Proxy オブジェクトに読み込みます。

形式 1 では、リンク Proxy オブジェクトの保持しているプロパティ値集合でのすべてのプロパティをリンクオブジェクトから読み込みます。形式 2 では、読み込み対象のプロパティのコレクションを指定します。

### (2) 形式

#### (a) 形式 1

```
void readProperties()
```

#### (b) 形式 2

```
void readProperties(
    Collection propDefs
)
```

### (3) 引数

propDefs (入力)

要素は、リンクプロパティ名 (String 型) です。取得するプロパティのプロパティ名のコレクションを指定します。null を指定すると例外がスローされます。

### (4) 戻り値

なし

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjObjectNotFoundException  
指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合

NullPointerException  
引数 propDefs が null の場合

## 4.2.9 removeObject (リンクオブジェクトの削除)

### (1) 機能

リンク Proxy オブジェクトのアクセス対象のターゲットオブジェクトであるリンクオブジェクトを削除して、リンクを解除します。

### (2) 形式

```
boolean removeObject()
```

### (3) 引数

なし

### (4) 戻り値

true

リンクオブジェクトが削除されました。

false

リンクオブジェクトが削除されませんでした。この値は、すでにリンクオブジェクトが削除されていた場合に返却されます。

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjObjectNotFoundException  
指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合



## 4.2.10 setPropSet (リンク Proxy オブジェクトのプロパティ値集合の設定)

### (1) 機能

リンク Proxy オブジェクトのプロパティ値集合に、指定したプロパティ値集合の内容をコピーして設定します。

### (2) 形式

```
void setPropSet (  
    DbjPropSet    propSet  
)
```

### (3) 引数

propSet (入力)

設定するプロパティ値集合インターフェースを指定します。null を指定するとオブジェクトのプロパティ値集合には、空値が仮定されます。

### (4) 戻り値

なし

### (5) 例外

なし

## 4.2.11 writeProperties (リンクプロパティ値の設定)

### (1) 機能

リンク Proxy オブジェクトのアクセス対象のターゲットオブジェクトであるリンクオブジェクトのプロパティを一括して設定します。

形式1では、アクセス対象のターゲットオブジェクトであるリンクオブジェクトのプロパティを、指定したプロパティ値集合で設定します。形式2および形式3では、アクセス対象のターゲットオブジェクトであるリンクオブジェクトのプロパティを、リンク Proxy オブジェクトに設定されたプロパティ値集合で設定します。形式3では、リンク Proxy オブジェクトに設定されているプロパティのうち、アクセス対象のターゲットオブジェクトであるリンクオブジェクトに反映する対象のプロパティを限定して指定できます。

### (2) 形式

#### (a) 形式1

```
void writeProperties(  
    DbjPropSet    propSet  
)
```

#### (b) 形式2

```
void writeProperties()
```

#### (c) 形式3

```
void writeProperties(  
    Collection    propdef  
)
```

### (3) 引数

propSet (入力)

リンクオブジェクトに設定するリンクプロパティ値集合を指定します。null を指定すると例外がスローされます。

propDefs (入力)

要素はリンクプロパティ名 (String 型) です。リンクオブジェクトに設定する対象のリンクプロパティ値のコレクションを指定します。null を指定すると例外がスローされます。

### (4) 戻り値

なし

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjIllegalPropValException

指定プロパティ値が不正の場合

DbjObjectNotFoundException

指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

NullPointerException

引数 propSet , または propDefs が null の場合

## 4.3 DbjLinkObjList インターフェース

---

DbjLinkObjList インターフェースは、DbjLinkObj インターフェースを要素とするリストインターフェースを扱います。

スーパーインターフェース

- java.util.List

以降、DbjLinkObjList インターフェースのメソッドについて説明します。

### 4.3.1 changeToVTFix ( 複数の構成管理型リンクの構成管理モードを一括して FIX モードに変更 )

#### (1) 機能

リスト要素のリンク Proxy オブジェクトについて、これらのアクセス対象のターゲットオブジェクトであるリンクオブジェクトの構成管理モードを、一括して FIX モードに変更します。なお、リンク Proxy オブジェクトのターゲットオブジェクトであるリンクオブジェクトには、構成管理型リンクを指定する必要があります。

#### (2) 形式

```
void changeToVTFix()
```

#### (3) 引数

なし

#### (4) 戻り値

なし

#### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjObjectNotFoundException  
指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合

### 4.3.2 changeToVTFloat ( 複数の構成管理型リンクの構成管理モードを一括して FLOATING モードに変更 )

#### (1) 機能

リスト要素のリンク Proxy オブジェクトについて、これらのアクセス対象のターゲットオブジェクトであるリンクオブジェクトの構成管理モードを、一括して FLOATING モードに変更します。なお、リンク Proxy オブジェクトのターゲットオブジェクトであるリンクオブジェクトには、構成管理型リンクを指定する必要があります。

#### (2) 形式

```
void changeToVTFloat()
```

#### (3) 引数

なし

#### (4) 戻り値

なし

#### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjObjectNotFoundException  
指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合

### 4.3.3 getLinkIdList (ターゲットリンク識別子プロパティのリストの取得)

#### (1) 機能

リンク Proxy オブジェクトが保持するリスト要素すべてのターゲットリンク識別子プロパティ値をリストで取得します。

#### (2) 形式

```
List getLinkIdList()
```

#### (3) 引数

なし

#### (4) 戻り値

ターゲットリンク識別子プロパティ値 (String 型) のリスト

#### (5) 例外

ClassCastException

リスト要素が DbjLinkObj インターフェースでなかった場合

### 4.3.4 getLinkObj ( リスト要素の DbjLinkObj インターフェースの取得 )

#### (1) 機能

インデクスで指定されたリスト要素の DbjLinkObj インターフェースを取得します。

#### (2) 形式

```
DbjLinkObj getLinkObj(  
    int    index  
)
```

#### (3) 引数

index ( 入力 )

リスト要素のインデクスを指定します。

#### (4) 戻り値

リスト要素の DbjLinkObj インターフェース

#### (5) 例外

ClassCastException

インデクスで指定されたリスト要素が DbjLinkObj インターフェースでなかった場合

IndexOutOfBoundsException

引数に指定したリスト要素のインデクスが不正の場合



### 4.3.5 getOwnerObjList (リンク元オブジェクトのリストの取得)

#### (1) 機能

リンク元オブジェクトが保持するリスト要素すべてのオーナーオブジェクトのプロパティ値をリストで取得します。

#### (2) 形式

```
DbjObjList getOwnerObjList()
```

#### (3) 引数

なし

#### (4) 戻り値

オーナーオブジェクトのプロパティ値のリスト (DbjObjList インターフェース)

#### (5) 例外

ClassCastException

リスト要素が DbjLinkObj インターフェースでなかった場合

### 4.3.6 getTargetObjList (リンク先オブジェクトのリストの取得)

#### (1) 機能

リンク先オブジェクトが保持するリスト要素すべてのターゲットオブジェクトのプロパティ値をリストで取得します。

#### (2) 形式

```
DbjObjList getTargetObjList()
```

#### (3) 引数

なし

#### (4) 戻り値

ターゲットオブジェクトのプロパティ値のリスト (DbjObjList インターフェース)

#### (5) 例外

ClassCastException

リスト要素が DbjLinkObj インターフェースでなかった場合

### 4.3.7 readProperties (リンクプロパティ値の一括取得)

#### (1) 機能

リスト要素のターゲットオブジェクトであるリンクオブジェクトの指定プロパティを各要素のリンク Proxy オブジェクトに一括して読み込みます。リストの要素である各 DbjLinkObj インターフェースに対して、DbjLinkObj#readProperties メソッドを実行します。

形式 1 では、Proxy オブジェクトの保持しているプロパティ値集合でのすべてのプロパティをターゲットオブジェクトであるリンクオブジェクトの指定プロパティから読み込みます。形式 2 では、読み込み対象のプロパティのコレクションを指定します。

#### (2) 形式

##### (a) 形式 1

```
void readProperties()
```

##### (b) 形式 2

```
void readProperties(
    Collection propDefs
)
```

#### (3) 引数

propDefs (入力)

要素はリンクプロパティ名 (String 型) です。取得するプロパティのプロパティ名のコレクションを指定します。null を指定すると例外がスローされます。

#### (4) 戻り値

なし

#### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjObjectNotFoundException  
指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合

NullPointerException  
引数 propDefs が null の場合

### 4.3.8 removeObjects ( リンクオブジェクトの一括削除 )

#### (1) 機能

リストオブジェクト要素のアクセス対象のターゲットオブジェクトであるリンクオブジェクトを一括して削除します。リストの要素である各 DbjLinkObj インターフェースに対して、DbjLinkObj#removeObject メソッドを実行します。

#### (2) 形式

```
List removeObjects()
```

#### (3) 引数

なし

#### (4) 戻り値

削除されたリンクオブジェクトの DbjLinkObj インターフェースのリスト

#### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjObjectNotFoundException  
指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合

### 4.3.9 writeProperties (リンクプロパティ値の一括設定)

#### (1) 機能

リスト要素のアクセス対象のターゲットオブジェクトであるリンクオブジェクトのプロパティを一括して設定します。

形式1では、要素のアクセス対象のターゲットオブジェクトであるリンクオブジェクトのプロパティを、指定したプロパティ値集合で設定します。形式2および形式3では、要素のアクセス対象のターゲットオブジェクトであるリンクオブジェクトのプロパティを、各要素のリンク Proxy オブジェクトに設定されたプロパティ値集合で設定します。形式3では、リンク Proxy オブジェクトに設定されているプロパティのうち、ターゲットオブジェクトであるリンクオブジェクトに反映するプロパティを限定して指定できます。

#### (2) 形式

##### (a) 形式1

```
void writeProperties(
    DbjPropSet      propSet
)
```

##### (b) 形式2

```
void writeProperties()
```

##### (c) 形式3

```
void writeProperties(
    Collection      propdef
)
```

#### (3) 引数

propSet (入力)

リンクオブジェクトに設定するリンクプロパティ値集合を指定します。null を指定すると例外がスローされます。

propDefs (入力)

要素はリンクプロパティ名 (String 型) です。リンクオブジェクトに設定する対象のプロパティ値集合を指定します。null を指定すると例外がスローされます。

#### (4) 戻り値

なし

#### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjIllegalPropValException

#### 4. 文書管理クラス詳細

指定プロパティ値が不正の場合

DbjObjectNotFoundException

指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

NullPointerException

引数 propSet , または propDefs が null の場合

## 4.4 DbjObj インターフェース

---

DbjObj インターフェースは、一つの文書空間オブジェクトへのアクセスを扱います。このインターフェースは、Proxy オブジェクトを通じて一つの文書空間オブジェクトにアクセスするメソッドを提供します。

以降、DbjObj インターフェースのメソッドについて説明します。

## 4.4.1 addRendition ( レンディションの追加 )

### (1) 機能

Proxy オブジェクトのアクセス対象の文書にレンディションを追加します。すでに登録されているレンディションが指定された場合は、例外がスローされます。レンディションを追加できるアクセス対象の文書空間オブジェクトは、文書だけです。

### (2) 形式

```
void addRendition(  
    List          uploadList  
)
```

### (3) 引数

uploadList ( 入力 )

要素は、DbjUploadInfo インターフェースです。文書に登録するレンディションのファイル情報 ( アップロード情報 ) のリストを指定します。null を指定すると例外がスローされます。

### (4) 戻り値

なし

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjIllegalObjectTypeException  
指定オブジェクト種別が不適切の場合

DbjIllegalPropValException  
指定レンディションプロパティ値が不正の場合

DbjIOException  
Java クラスライブラリ固有の IO エラーの場合

DbjMasterRenditionNotSetException  
マスタレンディションのレンディションタイプが設定されていなかった場合

DbjNotCheckOutException  
チェックアウトされていないオブジェクトに対して仮のバージョン識別子が指定された場合

DbjObjectNotFoundExcpetion  
指定オブジェクトが存在しなかった場合

DbjRenditionCountOutOfRangeExcpetion



レンディション数が範囲を超えた場合

DbjRenditionTypeDuplicatedException

指定レンディションタイプが重複していた場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

DbjXmlParseException

XML 文書解析エラーの場合

NullPointerException

引数 uploadList が null の場合

## 4.4.2 bindPublicACL (パブリック ACL のバインド)

### (1) 機能

Proxy オブジェクトのアクセス対象の文書空間オブジェクトに、指定したパブリック ACL をバインドします。ただし、Proxy オブジェクトのアクセス対象の文書空間オブジェクトには、パブリック ACL を指定しないでください。

### (2) 形式

```
void bindPublicACL(  
    List      publicACLList  
)
```

### (3) 引数

publicACLIdList (入力)

要素は DbjObj インターフェースです。各要素には、バインドするパブリック ACL のリストを指定します。null を指定すると例外がスローされます。

### (4) 戻り値

なし

### (5) 例外

ClassCastException

引数 publicACLList の要素が DbjObj インターフェースでなかった場合

DbjAccessControlException

アクセス権エラーの場合

DbjAccessControlNotSupportedException

アクセス制御機能に対応していない文書空間だった場合

DbjCORBAException

CORBA エラーの場合

DbjDBException

DB エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjIllegalObjectTypeException

指定オブジェクト種別が不適切の場合

DbjObjectNotFoundException

指定オブジェクトが存在しなかった場合

DbjPublicACLAlreadyBoundException

指定パブリック ACL はすでにバインドされていた場合

DbjPublicACLNotFoundException

指定パブリック ACL が存在しなかった場合

DbjPublicACLOutOfRangeException

パブリック ACL 数が制限を超えた場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

NullPointerException

引数 publicACLIdList が null の場合

### 4.4.3 cancelCheckOut (バージョン付きオブジェクトのチェックアウトの取り消し)

#### (1) 機能

Proxy オブジェクトのアクセス対象であるバージョン付きオブジェクトのチェックアウトを取り消します。チェックアウトを取り消すことができるターゲットオブジェクトは、バージョン付きオブジェクトだけです。

形式 1 および形式 2 では、取り消しと同時に Proxy オブジェクトに設定されているプロパティ値が、カレントバージョンのプロパティに設定されます。形式 2 では、設定するプロパティを限定できます (指定したプロパティだけが設定対象になります)。形式 3 では、Proxy オブジェクトのものに関係なく、指定したプロパティ値集合がカレントバージョンのプロパティを対象として設定されます。Proxy オブジェクトのターゲットバージョン識別子のプロパティは、カレントバージョン (null) になります。

#### (2) 形式

##### (a) 形式 1

```
void cancelCheckOut ()
```

##### (b) 形式 2

```
void cancelCheckOut (
    Collection    propDefs
)
```

##### (c) 形式 3

```
void cancelCheckOut (
    DbjPropSet    propSet
)
```

#### (3) 引数

propDefs (入力)

要素はプロパティ名 (String 型) です。チェックアウトの取り消し後のカレントバージョンを対象としたバージョン付きオブジェクトに設定する Proxy オブジェクトのプロパティとして、どのプロパティを設定するかを指定します。null を指定するとプロパティを設定しません。

propSet (入力)

チェックアウトの取り消し後のカレントバージョンを対象としたバージョン付きオブジェクトに設定するプロパティ値集合を指定します。null を指定するとプロパティを設定しません。

#### (4) 戻り値

なし

#### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjACEOperationException  
指定 ACE についてのエラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException

DB エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjFileReferenceCurrentContentNotFoundException

リファレンスファイル管理機能を使用するバージョン付きの文書オブジェクトで、カレントバージョンにコンテンツが存在しなかった場合

DbjFileReferenceMismatchStatusException

リファレンスファイル管理機能を使用するサーバでのコンテンツ操作でエラーが発生し、オブジェクトとコンテンツが不整合な状態になった場合

DbjFileReferenceOperationFailedException

リファレンスファイル管理機能を使用するサーバでのコンテンツ操作が失敗した場合

DbjIllegalObjectTypeException

指定オブジェクト種別が不適切の場合

DbjIllegalPropValException

指定プロパティ値が不正の場合

DbjNotCheckOutException

チェックアウトしていなかった場合

DbjObjectNotFoundException

指定オブジェクトが存在しなかった場合

DbjPublicACLNotFoundException

指定パブリック ACL が存在しなかった場合

DbjPublicACLOutOfRangeException

指定パブリック ACL 数が範囲を超えた場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

DbjTargetContentPathNotSetException

リファレンスファイル管理機能で、コンテンツ格納先ベースパスが設定されていなかった場合

NullPointerException

引数 propSet が null の場合

#### 4.4.4 changeMasterRendition ( マスタレンディションの変更 )

##### (1) 機能

Proxy オブジェクトのアクセス対象の文書のマスタレンディションを、指定したレンディションタイプのレンディションに変更します。マスタレンディションを変更できるターゲットオブジェクトは、文書だけです。

##### (2) 形式

```
void changeMasterRendition(  
    String      renditionType  
)
```

##### (3) 引数

renditionType ( 入力 )

マスタレンディションに変更するレンディションタイプを指定します。null を指定すると例外がスローされます。

##### (4) 戻り値

なし

##### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjIllegalObjectTypeException  
指定オブジェクト種別が不適切の場合

DbjIsMasterRenditionException  
すでにマスタレンディションだった場合

DbjNotCheckOutException  
チェックアウトされていないオブジェクトに対して仮のバージョン識別子が指定された場合

DbjObjectNotFoundExpection  
指定オブジェクトが存在しなかった場合

DbjRenditionIsEmptyException  
レンディションが空の場合

DbjRenditionNotConvertedException  
指定レンディションが変換されていなかった場合

DbjRenditionNotFoundExpection  
指定レンディションが存在しなかった場合

DbjSessionNotConnectException

セッションが接続されていない

NullPointerException

引数 renditionType が null の場合

## 4.4.5 changeToVTFix ( 構成管理型リンクの構成管理モードを FIX モードに変更 )

### (1) 機能

Proxy オブジェクトのアクセス対象の文書空間オブジェクトが指定した構成管理型リンクの構成管理モードを FIX モードに変更します。

形式 1 では、構成管理型リンクのリンク識別子を指定したバージョンのバージョン識別子で固定します。この場合、バージョン識別子に null を指定するとカレントバージョンになります。形式 2 では、変更するリンクのリンク識別子および固定するバージョンのバージョン識別子をマップで指定して固定します。要素のキーがリンク識別子となり、その値がバージョン識別子になります。この場合、バージョン識別子に null を指定するとカレントバージョンになります。形式 3 では、Proxy オブジェクトのターゲットオブジェクトのすべての構成管理型リンクを、その時点の状態で固定します。

### (2) 形式

#### (a) 形式 1

```
void changeToVTFix(  
    String      linkId,  
    String      fixVersionId  
)
```

#### (b) 形式 2

```
void changeToVTFix(  
    Map      changeMap  
)
```

#### (c) 形式 3

```
void changeToVTFix()
```

### (3) 引数

linkId ( 入力 )

構成管理型リンクのリンク識別子を指定します。null を指定すると例外がスローされます。

fixVersionId ( 入力 )

構成管理型リンクによって固定するバージョンのバージョン識別子を指定します。null を指定するとカレントバージョンになります。

changeMap ( 入力 )

要素のキーには、変更する構成管理型リンクのリンク識別子を指定します。値には、固定するバージョンのバージョン識別子のマップを指定します。null を指定すると例外がスローされます。

### (4) 戻り値

なし

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合



DbjDBException

DB エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjIllegalObjectTypeException

指定オブジェクト種別が不適切の場合

DbjNotCheckOutException

チェックアウトされていないオブジェクトに対して仮のバージョン識別子が指定された場合

DbjObjectNotFoundException

指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException

セッションが接続されていない

NullPointerException

引数 linkId , または changeMap が null の場合

## 4.4.6 changeToVTFLOAT ( 構成管理型リンクの構成管理モードを FLOATING モードに変更 )

### (1) 機能

Proxy オブジェクトのアクセス対象の文書空間オブジェクトが指定した構成管理型リンクの構成管理モードを FLOATING モードに変更します。

形式 1 では、変更する構成管理型リンクのリンク識別子を指定して常に最新のバージョンになるようにします。形式 2 では、変更する構成管理型リンクのリンク識別子を要素としたリストを指定して常に最新のバージョンになるようにします。形式 3 では、自オブジェクトのすべての構成管理型リンクが常に最新のバージョンになるようにします。

### (2) 形式

#### (a) 形式 1

```
void changeToVTFLOAT (
    String    linkId
)
```

#### (b) 形式 2

```
void changeToVTFLOAT (
    List     linkIdList
)
```

#### (c) 形式 3

```
void changeToVTFLOAT ()
```

### (3) 引数

linkId ( 入力 )

構成管理型リンクのリンク識別子を指定します。null を指定すると例外がスローされます。

linkIdList ( 入力 )

要素に構成管理型リンクのリンク識別子のリストを指定します。null を指定すると例外がスローされます。

### (4) 戻り値

なし

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjIllegalObjectTypeException

指定オブジェクト種別が不適切の場合

DbjNotCheckOutException

チェックアウトされていないオブジェクトに対して仮のバージョン識別子が指定された場合

DbjObjectNotFoundException

指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException

セッションが接続されていない

NullPointerException

引数 linkId , または linkIdList が null

## 4.4.7 checkIn (バージョン付きオブジェクトのチェックイン)

### (1) 機能

Proxy オブジェクトのアクセス対象のバージョン付きオブジェクトをチェックインします。チェックインできるターゲットオブジェクトは、バージョン付きオブジェクトだけです。

形式1および形式2では、チェックインと同時に Proxy オブジェクトに設定されているプロパティ値集合が、チェックイン後のカレントバージョンを対象としてプロパティに設定されます。形式2では、設定するプロパティを限定できます(指定したプロパティだけが設定の対象になります)。形式3では、Proxy オブジェクトのものに関係なく、指定したプロパティ値集合がカレントバージョンを対象として設定されます。Proxy オブジェクトのターゲットバージョン識別子のプロパティは、新しいカレントバージョン (null) になります。

### (2) 形式

#### (a) 形式1

```
void checkIn()
```

#### (b) 形式2

```
void checkIn(
    Collection      propDefs
)
```

#### (c) 形式3

```
void checkIn(
    DbjPropSet      propSet
)
```

### (3) 引数

#### propDefs (入力)

要素はプロパティ名 (String 型) です。チェックイン後のカレントバージョンを対象としたバージョン付きオブジェクトに設定する Proxy オブジェクトのプロパティとして、どのプロパティを設定するかを指定します。null を指定するとプロパティは設定されません。

#### propSet (入力)

チェックイン後のカレントバージョンを対象とした、バージョン付きオブジェクトに設定するプロパティ値集合を指定します。null を指定するとプロパティは設定されません。

### (4) 戻り値

なし

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjACEOperationException  
指定 ACE についてのエラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException

DB エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjFileReferenceCurrentContentNotFoundExpection

リファレンスファイル管理機能を使用するバージョン付きの文書オブジェクトで、カレントバージョンにコンテンツが存在しなかった場合

DbjFileReferenceOperationFailedException

リファレンスファイル管理機能を使用するサーバでのコンテンツ操作が失敗した場合

DbjIllegalObjectTypeException

指定オブジェクト種別が不適切の場合

DbjIllegalPropValException

指定プロパティ値が不正の場合

DbjNotCheckOutException

チェックアウトしていなかった場合

DbjObjectNotFoundExpection

指定オブジェクトが存在しなかった場合

DbjPublicACLNotFoundExpection

指定パブリック ACL が存在しなかった場合

DbjPublicACLOutOfRangeException

指定パブリック ACL 数が範囲を超えた場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

DbjTargetContentPathNotSetException

リファレンスファイル管理機能で、コンテンツ格納先ベースパスが設定されていなかった場合

NullPointerExpection

引数 propSet が null の場合

## 4.4.8 checkOut (バージョン付きオブジェクトのチェックアウト)

### (1) 機能

Proxy オブジェクトのアクセス対象のバージョン付きオブジェクトをチェックアウトします。チェックアウトによって、チェックアウト中の仮のバージョン識別子を取得します。チェックアウトできるターゲットオブジェクトは、バージョン付きオブジェクトだけです。

形式 1 および形式 2 では、チェックアウトと同時に Proxy オブジェクトに設定されているプロパティ値集合がチェックアウト中の仮のバージョンを対象としてプロパティに設定されます。形式 2 では、設定するプロパティを限定できます (指定したプロパティだけが、プロパティ設定の対象になります)。形式 3 では、Proxy オブジェクトのものに関係なく、指定したプロパティ値集合が仮のバージョンを対象として設定されます。

なお、マルチレンディションの文書をチェックアウトする場合は、チェックアウト時にレンディションがコピーされます。この場合、コンテンツはコピーされません。また、Proxy オブジェクトのターゲットバージョン識別子のプロパティは、チェックアウト中のバージョンを指します。

### (2) 形式

#### (a) 形式 1

```
void checkOut ()
```

#### (b) 形式 2

```
void checkOut (
    Collection      propDefs
)
```

#### (c) 形式 3

```
void checkOut (
    DbjPropSet      propSet
)
```

### (3) 引数

propDefs (入力)

要素はプロパティ名 (String 型) です。チェックアウト後の仮のバージョンオブジェクトを対象としたバージョン付きオブジェクトに設定する Proxy オブジェクトのプロパティとして、どのプロパティを設定するかを指定します。null を指定するとプロパティは設定されません。

propSet (入力)

チェックアウト後の仮のバージョンオブジェクトを対象としたバージョン付きオブジェクトに設定するプロパティ値集合を指定します。null を指定するとプロパティは設定されません。

### (4) 戻り値

なし

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjACEOperationException  
指定 ACE についてのエラーの場合

DbjAlreadyCheckOutException

すでにチェックアウトしている場合

DbjCORBAException

CORBA エラーの場合

DbjDBException

DB エラーの場合

DbjException

Java クラスライブラリ固有エラーの場合

DbjFileReferenceCurrentContentNotFoundException

リファレンスファイル管理機能を使用するバージョン付きの文書オブジェクトで、カレントバージョンにコンテンツが存在しなかった場合

DbjFileReferenceOperationFailedException

リファレンスファイル管理機能を使用するサーバでのコンテンツ操作が失敗した場合

DbjIllegalObjectTypeException

指定オブジェクト種別が不適切の場合

DbjIllegalPropValException

指定プロパティ値が不正の場合

DbjObjectNotFoundException

指定オブジェクトが存在しなかった場合

DbjPublicACLNotFoundException

指定パブリック ACL が存在しなかった場合

DbjPublicACLOutOfRangeException

指定パブリック ACL 数が範囲を超えた場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

DbjTargetContentPathNotSetException

リファレンスファイル管理機能で、コンテンツ格納先ベースパスが設定されていなかった場合

NullPointerException

引数 propSet が null の場合

## 4.4.9 deleteRendition ( レンディションの削除 )

### (1) 機能

Proxy オブジェクトのアクセス対象の文書から指定したレンディションタイプのレンディションを削除します。レンディションの削除によって、存在しなかったレンディションタイプのリストが返却されます。レンディションを削除できるターゲットオブジェクトは、文書だけです。

### (2) 形式

```
List deleteRendition(  
    List renditionTypeList  
)
```

### (3) 引数

renditionTypeList ( 入力 )

要素には、削除するレンディションタイプ (String 型) のリストを指定します。null を指定すると例外がスローされます。

### (4) 戻り値

すでに削除されていたレンディションタイプ (String 型) のリスト

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjIllegalObjectTypeException  
指定オブジェクト種別が不適切の場合

DbjIsMasterRenditionException  
マスタレンディションを削除しようとした場合

DbjNotCheckOutException  
チェックアウトされていないオブジェクトに対して仮のバージョン識別子が指定された場合

DbjObjectNotFoundException  
指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合

NullPointerException  
引数 renditionTypeList が null の場合



## 4.4.10 deleteVersion (バージョンの削除)

### (1) 機能

Proxy オブジェクトのアクセス対象のバージョン付きオブジェクトから指定した複数のバージョンを削除します。バージョンを削除できるターゲットオブジェクトは、文書だけです。ただし、複数のバージョンを削除する場合、1バージョンは残す必要があります。もし、すべてのバージョンを指定した場合は、例外がスローされます。バージョンの削除によって、すでに削除されていたバージョンなど、削除されなかったバージョンのバージョン識別子のリストが返却されます。

### (2) 形式

```
List deleteVersion(
    List versionIds
)
```

### (3) 引数

versionIds (入力)

要素には、削除するバージョンのバージョン識別子 (String 型) のリストを指定します。バージョン識別子に null を指定するとカレントバージョンになります。バージョン識別子が重複していてもエラーにはなりません。null を指定すると例外がスローされます。

### (4) 戻り値

すでに削除されていたオブジェクトのバージョンのバージョン識別子 (String 型) のリスト

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjFileReferenceCurrentContentNotFoundException  
リファレンスファイル管理機能を使用するバージョン付きの文書オブジェクトで、カレントバージョンにコンテンツが存在しなかった場合

DbjFileReferenceMismatchStatusException  
リファレンスファイル管理機能を使用するサーバでのコンテンツ操作でエラーが発生し、オブジェクトとコンテンツが不整合な状態になった場合

DbjFileReferenceOperationFailedException  
リファレンスファイル管理機能を使用するサーバでのコンテンツ操作が失敗した場合

DbjIllegalObjectTypeException  
指定オブジェクト種別が不適切の場合

DbjLastVersionException  
最後のバージョンを削除しようとした場合

#### 4. 文書管理クラス詳細

DbjNotCheckOutException

チェックアウトされていないオブジェクトに対して仮のバージョン識別子が指定された場合

DbjObjectNotFoundException

指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

DbjTargetContentPathNotSetException

リファレンスファイル管理機能で、コンテンツ格納先ベースパスが設定されていなかった場合

NullPointerException

引数 versionIds が null の場合

## 4.4.11 downloadContents ( 文書のコンテンツのダウンロード )

### (1) 機能

文書のコンテンツをダウンロードします。

形式 1 では、レンディションタイプを指定した文書のコンテンツを、指定したローカルファイルにダウンロードします。リファレンスファイル文書に対しては実行できません。形式 2 では、リファレンスファイル文書のコンテンツを、指定したパスにダウンロードします。リファレンスファイル文書に対してだけ実行できます。なお、ダウンロードの対象になるターゲットオブジェクトは、文書だけです。

### (2) 形式

#### (a) 形式 1

```
DbjContentInfo downloadContents(
    String      renditionType,
    String      filePath
)
```

#### (b) 形式 2

```
DbjReferenceContentInfo downloadContents(
    String      renditionType,
    DbjReferencePathInfo pathInfo
)
```

### (3) 引数

renditionType ( 入力 )

レンディションタイプを指定します。null を指定するとマスタレンディションが仮定されます。

filePath ( 入力 )

ダウンロード先のパス名 ( ローカルファイル, およびファイル名を含みます ) を指定します。null を指定すると例外がスローされます。

pathInfo ( 入力 )

ダウンロード先の情報を指定します。null を指定すると例外がスローされます。

### (4) 戻り値

形式 1 の場合

コンテンツ情報 ( DbjContentInfo インターフェース )

形式 2 の場合

リファレンスファイル文書のコンテンツ情報 ( DbjReferenceContentInfo インターフェース )

### (5) 例外

DbjAccessControlException

アクセス権エラーの場合

DbjContentNotRegisteredException

リファレンスファイル管理機能を使用したオブジェクトをローカルファイルにダウンロードする場合に、オブジェクトにコンテンツが登録されていないとき

DbjContentTypeMismatchException

リファレンスファイル管理機能で、オブジェクトのコンテンツ種別とメソッドが要求するコンテンツ

#### 4. 文書管理クラス詳細

種別が一致しない場合

DbjCORBAException

CORBA エラーの場合

DbjDBException

DB エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjFileReferenceOperationFailedException

リファレンスファイル管理機能を使用するサーバでのコンテンツ操作が失敗した場合

DbjIllegalObjectTypeException

指定オブジェクト種別が不適切の場合

DbjIOException

Java クラスライブラリ固有の IO エラーの場合

DbjNotCheckOutException

チェックアウトされていないオブジェクトに対して仮のバージョン識別子が指定された場合

DbjObjectNotFoundExpection

指定オブジェクトが存在しなかった場合

DbjRenditionNotFoundExpection

指定レンディションが存在しなかった場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

DbjTargetContentPathNotSetException

リファレンスファイル管理機能で、コンテンツ格納先ベースパスが設定されていなかった場合

NullPointerException

引数 filePath または引数 pathInfo が null の場合

## 4.4.12 getBindObjectList (パブリック ACL にバインドしている文書空間オブジェクト一覧の取得)

### (1) 機能

Proxy オブジェクトのアクセス対象のパブリック ACL にバインドされている文書空間オブジェクトの一覧情報を取得します。一覧情報を取得できるターゲットオブジェクトは、パブリック ACL だけです。取得対象とするオブジェクトのオブジェクト種別を指定できます。

バージョン付きオブジェクトが取得対象のオブジェクトに含まれる場合は、バージョンングオブジェクトのプロパティだけ取得でき、バージョンオブジェクトのプロパティは取得できません。

### (2) 形式

```
DbjObjList getBindObjectList(
    Collection      propDefs,
    int             objType,
    DbjFetchInfo   fetchInfo
)
```

### (3) 引数

#### propDefs (入力)

要素はプロパティ名 (String 型) です。一覧で取得するオブジェクトのプロパティ名のコレクションを指定します。取得するプロパティがない場合は null を指定します。

#### objType (入力)

一覧で取得するオブジェクトのオブジェクト種別を指定します。次に示す定数のどれかを指定できます。なお、論理和を使用した指定もできます。

- DbjDef.OBJTYPE\_FOLDER  
バージョンなしフォルダ
- DbjDef.OBJTYPE\_DOC  
バージョンなし文書
- DbjDef.OBJTYPE\_VRDOC  
バージョン付き文書
- DbjDef.OBJTYPE\_IP  
独立データ
- DbjDef.OBJTYPE\_VRFOLDER  
バージョン付きフォルダ
- DbjDef.OBJTYPE\_ANY  
任意のオブジェクト種別

#### fetchInfo (入出力)

一覧で取得する場合の検索結果取得情報を指定します。全件を取得する場合、またはキャッシュ検索しない場合は null を指定します。全件を取得しない場合、maxFetchCount プロパティに指定されている件数を取得した後に、comparator プロパティに指定したコンパレータによりソートされます。そのため、全件を取得する場合と、maxFetchCount プロパティを指定して検索した場合を比較すると、comparator プロパティに同じオブジェクトを指定しても結果が異なる場合があります。

### (4) 戻り値

バインドしているオブジェクト一覧情報 (DbjObjList インターフェース)

(5) 例外

DbjAccessControlException

アクセス権エラーの場合

DbjAccessControlNotSupportedException

アクセス制御機能に対応していない文書だった場合

DbjCORBAException

CORBA エラーの場合

DbjDBException

DB エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjIllegalCacheStartIndexException

検索結果数より大きい値を引数 fetchInfo ( DbjFetchInfo インターフェース ) の startIndex プロパティに指定した場合

DbjIllegalObjectTypeException

指定オブジェクト種別が不適切の場合

DbjObjectNotFoundException

指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

IllegalArgumentException

引数 fetchInfo の指定が不正の場合

### 4.4.13 getCheckOutStatus (バージョン付きオブジェクトのチェックアウト状態の取得)

#### (1) 機能

Proxy オブジェクトのアクセス対象のバージョン付きオブジェクトのチェックアウト状態を取得します。DbjCheckOutInfo インターフェースを使用して、チェックアウト状態、チェックアウトしているユーザ識別子、チェックアウトしている仮のバージョンのバージョン識別子などを取得できます。チェックアウト状態を取得できるターゲットオブジェクトは、バージョン付きオブジェクトだけです。

#### (2) 形式

```
DbjCheckOutInfo getCheckOutStatus()
```

#### (3) 引数

なし

#### (4) 戻り値

チェックアウト情報 (DbjCheckOutInfo インターフェース)

#### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjIllegalObjectTypeException  
指定オブジェクト種別が不適切の場合

DbjObjectNotFoundException  
指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合

#### 4.4.14 getChildList (フォルダのリンク先オブジェクト (下位オブジェクト) の取得)

##### (1) 機能

Proxy オブジェクトのアクセス対象のフォルダにリンクが設定されているリンク先オブジェクト (下位オブジェクト) の一覧情報を取得します。取得対象とするリンク種別とリンク先オブジェクトのオブジェクト種別を指定できます。

ユーザアプリケーションプログラムの取得対象のオブジェクトに、リンク種別として直接型リンクまたは参照型リンクを指定し、バージョン付きオブジェクトが含まれる場合について説明します。この場合、取得対象のオブジェクトがバージョンングオブジェクトのときは、バージョンングオブジェクトのプロパティだけ取得でき、バージョンオブジェクトのプロパティは取得できません。また、取得対象のオブジェクトがバージョンングオブジェクトを構成するバージョンオブジェクトのときは、バージョンオブジェクトのプロパティだけ取得できます。ただし、プロパティの指定には@プレフィックスを付けずに指定します。

ユーザアプリケーションプログラムの取得対象のオブジェクトに、リンク種別として構成管理型リンクを指定し、バージョン付きオブジェクトが含まれる場合について説明します。この場合、バージョンオブジェクトのプロパティを取得できます。プロパティの指定には、@プレフィックスを付けずに指定します。バージョンングオブジェクトのプロパティは取得できません。戻り値のリンクオブジェクト一覧情報から取得できるリンク先オブジェクトは、構成管理の対象となるバージョン付きオブジェクトです。このバージョンオブジェクトに対するバージョンングオブジェクトを取得する場合は、getVersioningInfo メソッドを使用します。

##### (2) 形式

```
DbjLinkObjList getChildList (
    Collection      propDefs,
    Collection      linkPropDefs,
    int             linkType,
    int             objType,
    DbjFetchInfo    fetchInfo
)
```

##### (3) 引数

###### propDefs (入力)

要素はプロパティ名 (String 型) です。一覧で取得するリンク先オブジェクトのプロパティ名のコレクションを指定します。取得するプロパティがない場合は null を指定します。

###### linkPropDefs (入力)

要素はプロパティ名 (String 型) です。一覧で取得するリンクプロパティ名のコレクションを指定します。取得するリンクプロパティがない場合は null を指定します。

###### linkType (入力)

リンク種別を指定します。次に示す定数のどれかを指定できます。なお、論理和を使用した指定もできます。

- DbjDef.LINK\_DCR  
直接型リンク
- DbjDef.LINK\_RCR  
参照型リンク
- DbjDef.LINK\_DCR | DbjDef.LINK\_RCR  
直接型リンクまたは参照型リンク



- DbjDef.LINK\_VTFIX  
構成管理型リンク（構成管理モードは FIX モード）
- DbjDef.LINK\_VTFLOAT  
構成管理型リンク（構成管理モードは FLOATING モード）
- DbjDef.LINK\_VTFIX | DbjDef.LINK\_VTFLOAT  
構成管理モードが FIX モードの構成管理型リンク，または構成管理モードが FLOATING モードの構成管理型リンク

#### objType（入力）

一覧で取得するリンク先オブジェクトのオブジェクト種別を指定します。次に示す定数のどれかを指定できます。なお，論理和を使用した指定もできます。

リンク種別に構成管理型リンク（DbjDef.LINK\_VTFIX または DbjDef.LINK\_VTFLOAT）を指定した場合は，バージョン付き文書（DbjDef.OBJTYPE\_VRDOC），バージョン付きフォルダ（DbjDef.OBJTYPE\_VRFOLDER）または任意のオブジェクト種別（DbjDef.OBJTYPE\_ANY）だけが有効になります。

- DbjDef.OBJTYPE\_FOLDER  
バージョンなしフォルダ
- DbjDef.OBJTYPE\_DOC  
バージョンなし文書
- DbjDef.OBJTYPE\_VRDOC  
バージョン付き文書
- DbjDef.OBJTYPE\_VRFOLDER  
バージョン付きフォルダ
- DbjDef.OBJTYPE\_ANY  
任意のオブジェクト種別

#### fetchInfo（入出力）

一覧で取得する場合の検索結果取得情報を指定します。全件を取得する場合，またはキャッシュ検索をしない場合は null を指定します。

全件を取得しない場合，maxFetchCount プロパティに指定されている件数を取得した後に，comparator プロパティに指定したコンパレータによりソートされます。そのため，全件を取得する場合と，maxFetchCount プロパティを指定して検索した場合を比較すると，comparator プロパティに同じオブジェクトを指定しても結果が異なる場合があります。

### （4）戻り値

リンクオブジェクト一覧情報（DbjLinkObjList インターフェース）

### （5）例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

#### 4. 文書管理クラス詳細

DbjIllegalObjectTypeException

指定オブジェクト種別が不適切の場合

DbjIllegalCacheStartIndexException

検索結果数より大きい値を引数 fetchInfo ( DbjFetchInfo インターフェース ) の startIndex プロパティに指定した場合

DbjNotCheckOutException

チェックアウトされていないオブジェクトに対して仮のバージョン識別子が指定された場合

DbjObjectNotFoundException

指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

IllegalArgumentException

引数 linkType , fetchInfo の指定が不正の場合

## 4.4.15 getClassNames ( アクセス対象文書空間オブジェクトを構成する DMA クラス名の取得 )

### (1) 機能

Proxy オブジェクトのアクセス対象の文書空間オブジェクトの DMA クラス名を取得します。

バージョンオブジェクトについては、バージョンングオブジェクトおよびバージョンオブジェクトを構成する二つの DMA クラスのクラス名をこの順に返し、このほかのオブジェクトについては一つの DMA クラスのクラス名を返します。ただし、バージョン付きオブジェクトに対しては、アクセス権エラーが発生するおそれがあります。

### (2) 形式

```
List getClassNames ()
```

### (3) 引数

なし

### (4) 戻り値

要素を DMA クラスのクラス名 (String 型) とするリストインターフェース

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjObjectNotFoundException  
指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合

## 4.4.16 getDCRParent ( 直接型リンクによるリンク元オブジェクトの取得 )

### (1) 機能

Proxy オブジェクトのアクセス対象のフォルダに直接型リンクが設定されているリンク元 ( 上位 ) オブジェクトの情報を取得します。リンク元オブジェクトがない場合は、null が返却されます。リンクオブジェクトについての情報 ( リンク識別子 ) は取得できません。

### (2) 形式

```
DbjObj getDCRParent (
    Collection      propDefs
)
```

### (3) 引数

propDefs ( 入力 )

要素はプロパティ名 ( String 型 ) です。取得するリンク元オブジェクトのプロパティ名のコレクションを指定します。取得するプロパティがない場合は null を指定します。

### (4) 戻り値

リンク元のリンクオブジェクト ( DbjObj インターフェース )

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjIllegalObjectTypeException  
指定オブジェクト種別が不適切の場合

DbjObjectNotFoundException  
指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合

#### 4.4.17 getLockType ( Proxy オブジェクトのアクセスロック種別の取得 )

##### (1) 機能

Proxy オブジェクトのアクセス対象の文書空間オブジェクトにアクセスする場合のロック種別 ( Proxy オブジェクトのアクセスロック種別のプロパティ ) を取得します。取得するロック種別は、参照系メソッドによって取得するロック種別です。

##### (2) 形式

```
int getLockType()
```

##### (3) 引数

なし

##### (4) 戻り値

次のどちらかのロック種別を取得します。

DbjDef.LOCK\_READ

read ロックを取得します。

DbjDef.LOCK\_WRITE

write ロックを取得します。

##### (5) 例外

なし

#### 4.4.18 getObjType ( アクセス対象文書空間オブジェクトのオブジェクト種別の取得 )

##### (1) 機能

Proxy オブジェクトのアクセス対象である文書空間オブジェクトのオブジェクト種別を取得します。

##### (2) 形式

```
int getObjType ()
```

##### (3) 引数

なし

##### (4) 戻り値

次のどれかのオブジェクト種別が返却されます。

DbjDef.OBJTYPE\_DOC

バージョンなし文書です。

DbjDef.OBJTYPE\_FOLDER

バージョンなしフォルダです。

DbjDef.OBJTYPE\_VRDOC

バージョン付き文書です。

DbjDef.OBJTYPE\_VRFOLDER

バージョン付きフォルダです。

DbjDef.OBJTYPE\_IP

独立データです。

DbjDef.OBJTYPE\_PUBLICACL

パブリック ACL です。

##### (5) 例外

DbjCORBAException

CORBA エラーの場合

DbjDBException

DB エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjObjectNotFoundException

指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

#### 4.4.19 getOiid ( アクセス対象文書空間オブジェクトの OIID の取得 )

##### (1) 機能

Proxy オブジェクトのアクセス対象である文書空間オブジェクトの OIID を取得します。

##### (2) 形式

```
String getOiid()
```

##### (3) 引数

なし

##### (4) 戻り値

Proxy オブジェクトのアクセス対象である文書空間オブジェクトの OIID 文字列 (String 型)

##### (5) 例外

なし

## 4.4.20 getParentList (フォルダのリンク元オブジェクトの取得)

### (1) 機能

Proxy オブジェクトのアクセス対象のフォルダにリンクが設定されているリンク元オブジェクトの一覧情報を取得します。取得対象のリンク種別およびリンク元オブジェクトのオブジェクト種別を指定できます。

このメソッドのリンク種別に DbjDef.LINK\_VT を指定した場合、取得したリンクオブジェクトには構成管理モード (FIX モードまたは FLOATING モード) の区別がありません。したがって、このメソッドで取得したリンクオブジェクトの構成管理モードが FIX モードまたは FLOATING モードのどちらの場合であっても、リンク種別には DbjDef.LINK\_VT が設定されます。

### (2) 形式

```
DbjLinkObjList getParentList (
    Collection      propDefs,
    Collection      linkPropDefs,
    int             linkType,
    int             objType,
    DbjFetchInfo    fetchInfo
)
```

### (3) 引数

#### propDefs (入力)

要素はプロパティ名 (String 型) です。一覧で取得するリンク元オブジェクトのプロパティ名のコレクションを指定します。取得するプロパティがない場合は null を指定します。

#### linkPropDefs (入力)

要素はプロパティ名 (String 型) です。一覧で取得するリンクプロパティ名のコレクションを指定します。取得するリンクプロパティがない場合は null を指定します。

#### linkType (入力)

リンク種別を指定します。次に示す定数のどれかを指定できます。

- DbjDef.LINK\_DCR  
直接型リンク
- DbjDef.LINK\_RCR  
参照型リンク
- DbjDef.LINK\_DCR | DbjDef.LINK\_RCR  
直接型リンクまたは参照型リンク
- DbjDef.LINK\_VT (=DbjDef.LINK\_VTFIX | DbjDef.LINK\_VTFLOAT)  
構成管理モードが FIX モードの構成管理型リンク、または構成管理モードが FLOATING モードの構成管理型リンク

#### objType (入力)

一覧で取得するリンク先オブジェクトのオブジェクト種別を指定します。次に示す定数のどれかを指定できます。なお、論理和を使用した指定もできます。

- DbjDef.OBJTYPE\_FOLDER  
バージョンなしフォルダ
- DbjDef.OBJTYPE\_VRFOLDER  
バージョン付きフォルダ
- DbjDef.OBJTYPE\_ANY  
任意のオブジェクト種別



**fetchInfo ( 入出力 )**

一覧で取得する場合の検索結果取得情報を指定します。全件を取得する場合、またはキャッシュ検索をしない場合は null を指定します。全件を取得しない場合、maxFetchCount プロパティに指定されている件数を取得した後に、comparator プロパティに指定したコンパレータによりソートされます。そのため、全件を取得する場合と、maxFetchCount プロパティを指定して検索した場合を比較すると、comparator プロパティに同じオブジェクトを指定しても結果が異なる場合があります。

**( 4 ) 戻り値**

リンクオブジェクト一覧情報 ( DbjLinkObjList インターフェース )

**( 5 ) 例外**

DbjAccessControlException

アクセス権エラーの場合

DbjCORBAException

CORBA エラーの場合

DbjDBException

DB エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjIllegalCacheStartIndexException

検索結果数より大きい値を引数 fetchInfo ( DbjFetchInfo インターフェース ) の startIndex プロパティに指定した場合

DbjIllegalObjectTypeException

指定オブジェクト種別が不適切の場合

DbjObjectNotFoundException

指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

IllegalArgumentException

引数 linkType , fetchInfo の指定が不正の場合

## 4.4.21 getPublicACLList ( バインドしているパブリック ACL 一覧の取得 )

### (1) 機能

Proxy オブジェクトのアクセス対象の文書空間オブジェクトがバインドしているパブリック ACL の一覧情報を取得します。ただし、Proxy オブジェクトのアクセス対象の文書空間オブジェクトには、パブリック ACL を指定しないでください。

### (2) 形式

```
DbjObjList getPublicACLList(  
    Collection propDefs  
)
```

### (3) 引数

propDefs (入力)

要素はプロパティ名 (String 型) です。一覧で取得するパブリック ACL のプロパティ名のコレクションを指定します。取得するプロパティがない場合は null を指定します。

### (4) 戻り値

バインドしているパブリック ACL 一覧情報 ( DbjObjList インターフェース )

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjAccessControlNotSupportedException  
アクセス制御機能に対応していない文書空間だった場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjIllegalObjectTypeException  
指定オブジェクト種別が不適切の場合

DbjObjectNotFoundException  
指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合

## 4.4.22 getRelList ( 文書間リンク一覧の取得 )

### (1) 機能

Proxy オブジェクトのアクセス対象の文書に文書間リンクが設定されている文書の一覧情報を取得します。一覧情報を取得できるターゲットオブジェクトは、文書だけです。この場合、取得対象とするオブジェクトのオブジェクト種別を指定できます。また、引数 ( relationendType ) の指定によって、取得するオブジェクトの情報をフィルタリングできます。

### (2) 形式

```
DbjLinkObjList getRelList(
    Collection propDefs,
    Collection linkPropDefs,
    int relationendType,
    int objType,
    DbjFetchInfo fetchInfo
)
```

### (3) 引数

#### propDefs ( 入力 )

要素はプロパティ名 ( String 型 ) です。一覧で取得するリンク先オブジェクトのプロパティ名集合を指定します。取得するプロパティがない場合は null を指定します。

#### linkPropDefs ( 入力 )

要素はプロパティ名 ( String 型 ) です。一覧で取得するリンクプロパティ名集合を指定します。取得するリンクプロパティがない場合は null を指定します。

#### relationendType ( 入力 )

取得するオブジェクトの条件を指定します。次に示す定数のどれかを指定できます。RELATIONEND\_STATUS 定数を指定しない場合は、リンク先のオブジェクトはすべて取得します。なお、DbjDef.RELATIONEND\_HEAD または DbjDef.RELATIONEND\_TAIL のどちらか一つは、必ず指定してください。

- DbjDef.RELATIONEND\_HEAD  
リンク先オブジェクトを取得します。RELATIONEND\_TAIL と同時に指定できません。
- DbjDef.RELATIONEND\_TAIL  
リンク元オブジェクトを取得します。RELATIONEND\_HEAD と同時に指定できません。
- DbjDef.RELATIONEND\_STATUS\_EXIST  
リンク先オブジェクトがあるオブジェクトを取得します。
- DbjDef.RELATIONEND\_STATUS\_NOT\_EXIST  
リンク先オブジェクトが削除されて存在しないオブジェクトを取得します。

#### objType ( 入力 )

一覧で取得するリンク元またはリンク先オブジェクトのオブジェクト種別を指定します。次に示す定数のどれかを指定できます。なお、論理和を使用した指定もできます。

- DbjDef.OBJTYPE\_DOC  
バージョンなし文書
- DbjDef.OBJTYPE\_VRDOC  
バージョン付き文書
- DbjDef.OBJTYPE\_ANY  
任意のオブジェクト種別

#### 4. 文書管理クラス詳細

##### fetchInfo ( 入出力 )

一覧で取得する場合の検索結果取得情報を指定します。全件を取得する場合、またはキャッシュ検索をしない場合は null を指定します。全件を取得しない場合、maxFetchCount プロパティに指定されている件数を取得した後に、comparator プロパティに指定したコンパレータによりソートされます。そのため、全件を取得する場合と、maxFetchCount プロパティを指定して検索した場合を比較すると、comparator プロパティに同じオブジェクトを指定しても結果が異なる場合があります。

##### ( 4 ) 戻り値

リンクオブジェクト一覧情報 ( DbjLinkObjList インターフェース )

##### ( 5 ) 例外

DbjAccessControlException

アクセス権エラーの場合

DbjCORBAException

CORBA エラーの場合

DbjDBException

DB エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjIllegalCacheStartIndexException

検索結果数より大きい値を引数 fetchInfo ( DbjFetchInfo インターフェース ) の startIndex プロパティに指定した場合

DbjIllegalObjectTypeException

指定オブジェクト種別が不適切の場合

DbjObjectNotFoundException

指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

IllegalArgumentException

引数 fetchInfo の指定が不正の場合

## 4.4.23 getRenditionList ( レンディション情報一覧の取得 )

### (1) 機能

Proxy オブジェクトのアクセス対象の文書が保持するレンディション情報の一覧をすべて取得します。レンディション情報を取得できるアクセス対象の文書空間オブジェクトは、文書だけです。

### (2) 形式

```
DbjRenditionList getRenditionList(
    Collection propDefs
)
```

### (3) 引数

propDefs ( 入力 )

要素はレンディションプロパティ名 ( String 型 ) です。一覧で取得するレンディションのプロパティ名のコレクションを指定します。取得するレンディションプロパティがない場合は null を指定します。次に示すプロパティを指定できます。

- dbrProp\_RetrievalName プロパティ
- dbrProp\_RenditionStatus プロパティ
- dbrProp\_ContentType プロパティ

### (4) 戻り値

レンディション一覧情報 ( DbjRenditionList インターフェース )

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjIllegalObjectTypeException  
指定オブジェクト種別が不適切の場合

DbjNotCheckOutException  
チェックアウトされていないオブジェクトに対して仮のバージョン識別子が指定された場合

DbjObjectNotFoundException  
指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合

#### 4.4.24 getTargetVersion ( アクセス対象ターゲットバージョンのバージョン識別子の取得 )

##### (1) 機能

Proxy オブジェクトのアクセス対象である文書空間オブジェクトのバージョンのバージョン識別子を取得します。

##### (2) 形式

```
String getTargetVersion()
```

##### (3) 引数

なし

##### (4) 戻り値

Proxy オブジェクトのアクセス対象である文書空間オブジェクトのバージョンのバージョン識別子 (String 型) が返却されます。なお、カレントバージョンの場合は、null が返却されることがあります。

##### (5) 例外

なし

## 4.4.25 getVersionId (バージョンオブジェクトのバージョン識別子の取得)

### (1) 機能

Proxy オブジェクトのアクセス対象オブジェクトがバージョン付きオブジェクトのバージョンオブジェクトの場合、そのバージョン識別子を取得します。アクセス対象の文書空間オブジェクトがバージョンオブジェクトでない場合は、null が返却されます。

### (2) 形式

```
String getVersionId()
```

### (3) 引数

なし

### (4) 戻り値

バージョンオブジェクトのバージョン識別子 (String 型)

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjObjectNotFoundException  
指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合

## 4.4.26 getVersioningInfo (バージョン付きオブジェクトのバージョンニングオブジェクトの取得)

### (1) 機能

Proxy オブジェクトのアクセス対象の文書空間オブジェクトがバージョン付きオブジェクトの場合、このバージョンオブジェクトに対応するバージョンニングオブジェクトの Proxy オブジェクトインターフェース (DbjObj インターフェース) を取得します。該当するバージョンのバージョン識別子は、その Proxy オブジェクトインターフェースの DbjObj#getTargetVersion メソッドで取得できます。したがって、取得した DbjObj インターフェースが指す Proxy オブジェクトの対象バージョンは、Proxy オブジェクトのバージョン識別子を示します。

アクセス対象の文書空間オブジェクトがバージョンオブジェクトでない場合、null が返却されます。

### (2) 形式

```
DbjObj getVersioningInfo()
```

### (3) 引数

なし

### (4) 戻り値

バージョン付きオブジェクトの Proxy オブジェクトインターフェース (DbjObj インターフェース)

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjObjectNotFoundException  
指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合



## 4.4.27 getVersionObjList (バージョン付きオブジェクトのバージョン一覧の取得)

### (1) 機能

Proxy オブジェクトのアクセス対象であるバージョン付きオブジェクトのバージョンオブジェクトの一覧情報を取得します。一覧情報を取得するアクセス対象の文書空間オブジェクトは、バージョン付きオブジェクトだけです。取得対象のバージョンオブジェクトから取得するプロパティを指定できます。また、一覧で取得するバージョンオブジェクトの並べ方を指定できます。

### (2) 形式

```
DbjVerObjList getVersionObjList (
    Collection          propDefs,
    int                 order,
    DbjFetchInfo        fetchInfo
)
```

### (3) 引数

#### propDefs (入力)

要素はプロパティ名 (String 型) です。一覧で取得するバージョンオブジェクトのプロパティ名のコレクションを指定します。取得するプロパティがない場合は null を指定します。

dmaProp\_OIID プロパティを指定しない場合、dmaProp\_OIID プロパティが自動的に追加されるものとみなします。

#### order (入力)

一覧で取得する場合の並べ方を指定します。次の定数のどれかを指定できます。

- DbjDef.ORDER\_ASC  
古い順でソートする
- DbjDef.ORDER\_DESC  
新しい順でソートする
- DbjDef.ORDER\_NONE  
ソート方法を指定しない

#### fetchInfo (入出力)

一覧で取得する場合の検索結果取得情報を指定します。全件を取得する場合、またはキャッシュ検索をしない場合は null を指定します。

キャッシュ検索すると cacheKey プロパティと cacheTotal プロパティが API によって設定されます。全件を取得しない場合、maxFetchCount プロパティに指定されている件数を取得した後に、comparator プロパティに指定したコンパレータによりソートされます。そのため、全件を取得する場合と、maxFetchCount プロパティを指定して検索した場合を比較すると、comparator プロパティに同じオブジェクトを指定しても結果が異なる場合があります。

### (4) 戻り値

バージョンオブジェクトの一覧情報 (DbjVerObjList インターフェース)

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException

#### 4. 文書管理クラス詳細

CORBA エラーの場合

DbjDBException

DB エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjIllegalCacheStartIndexException

検索結果数より大きい値を引数 fetchInfo ( DbjFetchInfo インターフェース ) の startIndex プロパティに指定した場合

DbjIllegalObjectTypeException

指定オブジェクト種別が不適切の場合

DbjObjectNotFoundException

指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

IllegalArgumentException

引数 fetchInfo の指定が不正の場合

## 4.4.28 link (リンク先オブジェクトとのリンク)

### (1) 機能

Proxy オブジェクトのアクセス対象オブジェクトのリンク元オブジェクトに、指定した複数のリンク先オブジェクトとのリンクを設定します。

形式 1 では、DbjSetLinkInfo インターフェースを要素とするリストオブジェクトで、リンク先オブジェクトとリンク種別を指定します。この場合、リンク先オブジェクトごとにリンク種別を選択できます。形式 2 では、リンク種別は一つだけ選択でき、DbjObj インターフェースのリストにリンク先オブジェクトを指定できます。

### (2) 形式

#### (a) 形式 1

```
void link(
    List      childLinkList
)
```

#### (b) 形式 2

```
void link(
    int      linkType,
    List     childObjs
)
```

### (3) 引数

#### childLinkList (入力)

要素は DbjSetLinkInfo インターフェースのサブインターフェースです。リンクを設定するリンク先オブジェクトとそのリンクの種類のリストを指定します。null を指定すると例外がスローされます。

#### linkType (入力)

リンク種別を指定します。次のどれかを指定できます。

- DbjDef.LINK\_DCR  
直接型リンク
- DbjDef.LINK\_RCR  
参照型リンク
- DbjDef.LINK\_VTFIX  
構成管理型リンク (構成管理モードは FIX モード)
- DbjDef.LINK\_VTFLOAT  
構成管理型リンク (構成管理モードは FLOATING モード)
- DbjDef.LINK\_REL  
文書間リンク

#### childObjs (入力)

要素は DbjObj インターフェースです。リンク対象のリンク先オブジェクトのリストを指定します。null を指定すると例外がスローされます。

### (4) 戻り値

なし

### (5) 例外

ClassCastException

#### 4. 文書管理クラス詳細

リスト引数の要素が不適切の場合

DbjAccessControlException

アクセス権エラーの場合

DbjCORBAException

CORBA エラーの場合

DbjDBException

DB エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjIllegalObjectTypeException

指定オブジェクト種別が不適切の場合

DbjIllegalPropValException

指定プロパティ値が不正の場合

DbjNotCheckOutException

チェックアウトされていないオブジェクトに対して仮のバージョン識別子が指定された場合

DbjObjectNotFoundExcpetion

指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

IllegalArgumentException

引数 linkType が不正の場合

NullPointerException

引数 childLinkList , または childObjs が null の場合

## 4.4.29 lock (アクセスロック種別の異なる文書空間オブジェクトインターフェースの取得)

### (1) 機能

Proxy オブジェクトのアクセス対象の文書空間オブジェクトに、指定したロック種別でロックを取得する Proxy オブジェクトのインターフェース (DbjObj インターフェース) を取得します。

元の DbjObj インターフェース (this) でアクセスする際のロック種別は、変更されません。なお、引数に指定する値の妥当性は検証されません。

### (2) 形式

```
DbjObj lock(  
    int      lockType  
)
```

### (3) 引数

lockType (入力)

ロック種別を指定します。次に示す定数のどちらかを指定します。

- DbjDef.LOCK\_READ  
read ロック
- DbjDef.LOCK\_WRITE  
write ロック

### (4) 戻り値

DbjObj インターフェース (DbjObj インターフェース)

### (5) 例外

なし

### 4.4.30 move (直接型リンクが設定されている文書空間オブジェクトの移動)

#### (1) 機能

Proxy オブジェクトのアクセス対象の文書空間オブジェクトに対してリンク元オブジェクトから直接型リンクが設定されている場合、そのリンクを解除して代わりに指定したリンク元フォルダオブジェクトに直接型リンクを設定します。文書空間オブジェクトに直接型リンクが設定されていない場合は、指定したリンク元フォルダに直接型リンクを設定します。

#### (2) 形式

```
void move(  
    DbjObj      parentFolder  
)
```

#### (3) 引数

parentFolder (入力)

自オブジェクトが移動するリンク元フォルダを指定します。null を指定すると例外がスローされま  
す。

#### (4) 戻り値

なし

#### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjIllegalObjectTypeException  
指定オブジェクト種別が不適切の場合

DbjNotCheckOutException  
チェックアウトされていないオブジェクトに対して仮のバージョン識別子が指定された場合

DbjObjectNotFoundException  
指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合

NullPointerException  
引数 parentFolder が null の場合

### 4.4.31 propSet ( Proxy オブジェクトのプロパティ値集合インターフェースの取得 )

#### (1) 機能

Proxy オブジェクトのプロパティ値集合インターフェースを取得します。プロパティ値集合インターフェースは、参照権限で取得します。取得したプロパティ値集合インターフェースを使用して値を設定すると Proxy オブジェクトのプロパティ値も更新されます。ただし、メソッド実行時に、プロパティ値集合インターフェースが設定されていなくても null は返却されません。

#### (2) 形式

```
DbjPropSet propSet ()
```

#### (3) 引数

なし

#### (4) 戻り値

Proxy オブジェクトのプロパティ値集合インターフェース ( DbjPropSet インターフェース )

#### (5) 例外

なし

### 4.4.32 readProperties ( 文書空間オブジェクトのプロパティ値の読み込み )

#### (1) 機能

Proxy オブジェクトのアクセス対象である文書空間オブジェクトの指定プロパティを、Proxy オブジェクトのプロパティに読み込みます。Proxy オブジェクトに読み込まれたプロパティ値集合は、DbjObj#propSet メソッドで DbjPropSet インターフェースを取得して参照できます。読み込まれたプロパティ値集合は、Proxy オブジェクトで保持しているプロパティ値集合に上書きして追加されます。

形式 1 では、Proxy オブジェクトの保持しているプロパティ値集合でのすべてのプロパティを文書空間オブジェクトから読み込みます。形式 2 では、読み込み対象のプロパティのコレクションを指定します。

#### (2) 形式

##### (a) 形式 1

```
void readProperties()
```

##### (b) 形式 2

```
void readProperties(  
    Collection    propDefs  
)
```

#### (3) 引数

propDefs ( 入力 )

要素はプロパティ名 ( String 型 ) です。取得するプロパティのプロパティ名のコレクションを指定します。null を指定すると例外がスローされます。

#### (4) 戻り値

なし

#### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjNotCheckOutException  
チェックアウトされていないオブジェクトに対して仮のバージョン識別子が指定された場合

DbjObjectNotFoundException  
指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合



NullPointerException

引数 propDefs が null の場合

### 4.4.33 removeObject ( 文書空間オブジェクトの削除 )

#### (1) 機能

Proxy オブジェクトのアクセス対象の文書空間オブジェクトを削除します。

形式 2 では、フォルダの削除の場合は直接型リンクが設定されているすべてのリンク先オブジェクトを削除します。

形式 3 では、リファレンスファイル管理機能を使用したオブジェクトを削除すると同時に、コンテンツを格納していたフォルダを削除できます。

なお、デッドロックを防止するため、リンク先オブジェクトを削除する場合は、リンク元オブジェクトを削除するまたはリンク元オブジェクトに write ロックします。このメソッドの実行時に、文書空間オブジェクトが存在しない場合、例外はスローされません。

#### (2) 形式

##### (a) 形式 1

```
boolean removeObject()
```

##### (b) 形式 2

```
boolean removeObject(  
    boolean isAlsoChildren  
)
```

##### (c) 形式 3

```
boolean removeObject(  
    DbjReferencePathInfo pathInfo  
)
```

#### (3) 引数

isAlsoChildren ( 入力 )

リンク先オブジェクトの削除を指定します。true の場合、リンク先オブジェクトを削除します。false の場合、リンク先オブジェクトを削除しません。

pathInfo ( 入力 )

リファレンスファイル管理機能を使用したオブジェクトの削除時に使用する、パス情報を設定します。null を指定すると例外がスローされます。

#### (4) 戻り値

次に示す値が返却されます。

true

リンク先オブジェクトが削除されました。

false

該当する文書空間オブジェクトが存在しませんでした。

#### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException

DB エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjFileReferenceCurrentContentNotFoundException

リファレンスファイル管理機能を使用するバージョン付きの文書オブジェクトで、カレントバージョンにコンテンツが存在しなかった場合

DbjFileReferenceMismatchStatusException

リファレンスファイル管理機能を使用するサーバでのコンテンツ操作でエラーが発生し、オブジェクトとコンテンツが不整合な状態になった場合

DbjFileReferenceOperationFailedException

リファレンスファイル管理機能を使用するサーバでのコンテンツ操作が失敗した場合

DbjLastVersionException

最後のバージョンを削除しようとした場合

DbjNotCheckOutException

チェックアウトされていないオブジェクトに対して仮のバージョン識別子が指定された場合

DbjObjectNotFoundException

指定オブジェクトが存在しなかった場合

DbjReferenceTypeMismatchException

リファレンスファイル管理機能で、オブジェクトのリファレンス種別とメソッドが要求するリファレンス種別が一致しない場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

DbjTargetContentPathNotSetException

リファレンスファイル管理機能で、コンテンツ格納先ベースパスが設定されていなかった場合

NullPointerException

引数 pathInfo が null の場合

#### 4.4.34 setPropSet ( Proxy オブジェクトのプロパティ値集合の設定 )

##### (1) 機能

Proxy オブジェクトのプロパティ値集合に、指定したプロパティ値集合の内容をコピーして設定します。

##### (2) 形式

```
void setPropSet (
    DbjPropSet    propSet
)
```

##### (3) 引数

propSet ( 入力 )

プロパティ値集合インターフェースを指定します。null を指定するとオブジェクトのプロパティ値集合には空値が仮定されます。

##### (4) 戻り値

なし

##### (5) 例外

なし

### 4.4.35 setTargetVersion ( アクセス対象ターゲットバージョンのバージョン識別子の変更 )

#### (1) 機能

Proxy オブジェクトのアクセス対象である文書空間オブジェクトのバージョンのバージョン識別子を変更します。バージョン識別子の変更によって、元の DbjObj インターフェースを取得します。デフォルトでは、Proxy オブジェクトのアクセス対象のバージョンは、カレントバージョンになります。なお、バージョン識別子の妥当性は検証されません。

#### (2) 形式

```
DbjObj setTargetVersion(  
    String      versionId  
)
```

#### (3) 引数

versionId ( 入力 )

変更する対象バージョンのバージョン識別子を指定します。null を指定するとカレントバージョンに変更されません。

#### (4) 戻り値

元のオブジェクトインターフェース ( this )

#### (5) 例外

なし

## 4.4.36 unbindPublicACL (パブリック ACL のアンバインド)

### (1) 機能

Proxy オブジェクトのアクセス対象の文書空間オブジェクトから指定したパブリック ACL をアンバインドします。ただし、Proxy オブジェクトがアクセスする文書空間オブジェクトには、パブリック ACL を指定しないでください。

### (2) 形式

```
void unbindPublicACL(  
    List      publicACLList  
)
```

### (3) 引数

publicACLIdList (入力)

要素は DbjObj インターフェースです。各要素に、アンバインドするパブリック ACL を指定します。null を指定すると例外がスローされます。

### (4) 戻り値

なし

### (5) 例外

ClassCastException

引数 publicACLList の要素が DbjObj インターフェースでなかった場合

DbjAccessControlException

アクセス権エラーの場合

DbjAccessControlNotSupportedException

接続先の文書空間がアクセス制御動作モードに対応していなかった場合

DbjCORBAException

CORBA エラーの場合

DbjDBException

DB エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjIllegalObjectTypeException

指定オブジェクト種別が不適切の場合

DbjObjectNotFoundException

指定オブジェクトが存在しなかった場合

DbjPublicACLNotBoundException

指定パブリック ACL がバインドされていない場合

DbjSessionNotConnectException

セッションが接続されていない場合

NullPointerException

引数 `publicACLIdList` が `null` の場合

## 4.4.37 unlink (リンク先オブジェクトとのリンクの解除)

### (1) 機能

Proxy オブジェクトのアクセス対象である文書空間オブジェクトのリンク元オブジェクトから、指定した複数のリンク先オブジェクトとのリンクを解除します。

リンクを解除するリンク先オブジェクトは、DbjLinkObj インターフェースまたは DbjObj インターフェースを要素とするリストで指定します。DbjLinkObj インターフェースと DbjObj インターフェースを同時に指定することもできます。DbjObj インターフェースを指定した場合、その DbjObj インターフェースが指す文書空間オブジェクトとの直接型リンクおよび参照型リンクがすべて解除されます。

なお、指定したリンクがすでに解除されていた場合は、そのリンクまたはリンク先オブジェクトのインターフェースを要素とするリストが戻り値として返却されます。

### (2) 形式

```
List unlink(  
    List childList  
)
```

### (3) 引数

childList (入力)

要素は DbjLinkObj インターフェースまたは DbjObj インターフェースです。

リストの要素として、リンクを解除するリンクオブジェクトのインターフェースまたはリンク先オブジェクトのインターフェースを指定します。null を指定すると例外がスローされます。

### (4) 戻り値

すでにリンクが解除されていたインターフェースのリスト

### (5) 例外

ClassCastException

リスト引数の要素が不適切の場合

DbjAccessControlException

アクセス権エラーの場合

DbjCORBAException

CORBA エラーの場合

DbjDBException

DB エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjIllegalObjectTypeException

指定オブジェクト種別が不適切の場合

DbjNotCheckOutException

チェックアウトされていないオブジェクトに対して仮のバージョン識別子が指定された場合

DbjObjectNotFoundExpection

指定オブジェクトが存在しなかった場合



DbjSessionNotConnectException

セッションが接続されていなかった場合

NullPointerException

引数 childLinkList , または childObjs が null の場合

## 4.4.38 unlinkByLinkId (リンク識別子の指定によるリンク先オブジェクトとのリンクの解除)

### (1) 機能

Proxy オブジェクトのアクセス対象である文書空間オブジェクトのリンク元オブジェクトから複数のリンク先オブジェクトのリンク識別子を指定してリンクを解除します。なお、すでにリンクが解除されていたため、メソッドの実行によってリンクが解除されなかったリンクのリンク識別子の要素が返却されます。

### (2) 形式

```
List unlinkByLinkId(  
    List linkIds  
)
```

### (3) 引数

linkIds (入力)

要素には、解除するリンク識別子 (String 型) のリストを指定します。null を指定すると例外がスローされます。

### (4) 戻り値

すでにリンクが解除されていたため、メソッドの実行によって、リンクが解除されなかったリンクのリンク識別子 (String 型) のリスト

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjObjectNotFoundException  
指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合

NullPointerException  
引数 linkIds が null の場合

### 4.4.39 uploadContents ( 文書のコンテンツのアップロード )

#### (1) 機能

指定したレンディションタイプのコンテンツをアップロードして更新します。アップロードの対象になるターゲットオブジェクトは、文書だけです。また、マルチレンディション文書の場合、更新の対象になるレンディションは、指定したレンディションタイプのレンディションだけです。

アップロード情報に指定するレンディションタイプとアップロード対象に指定するレンディションタイプに異なるレンディションを指定した場合、アップロードするコンテンツのレンディションタイプを変更できます。この場合、アップロード情報に指定したレンディションタイプがすでに文書に存在していると、例外がスローされます。アップロード情報の指定でレンディションタイプに null を設定した場合は、レンディションタイプは変更されません。

リファレンスファイル文書を更新する場合は、引数 upLoadInfo に DbjReferenceUploadInfo インターフェースを指定します。

#### (2) 形式

```
void uploadContents(
    String          renditionType,
    DbjUploadInfo  uploadInfo
)
```

#### (3) 引数

renditionType ( 入力 )

更新対象のレンディションタイプを指定します。null を指定するとマスタレンディションが仮定されます。

なお、全文検索インデックスを作成できるのは、null を指定した場合だけです。null 以外を指定した場合、全文検索インデックスは作成されません。

uploadInfo ( 入力 )

文書に登録するファイルのアップロード情報を指定します。null を指定すると例外がスローされます。

#### (4) 戻り値

なし

#### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjContentTypeMismatchException  
リファレンスファイル管理機能で、オブジェクトのコンテンツ種別とメソッドが要求するコンテンツ種別が一致しない場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException

#### 4. 文書管理クラス詳細

Java クラスライブラリ固有のエラーの場合

DbjFileReferenceMismatchStatusException

リファレンスファイル管理機能を使用するサーバでのコンテンツ操作でエラーが発生し、オブジェクトとコンテンツが不整合な状態になった場合

DbjFileReferenceOperationFailedException

リファレンスファイル管理機能を使用するサーバでのコンテンツ操作が失敗した場合

DbjIllegalObjectTypeException

指定オブジェクト種別が不適切の場合

DbjIllegalPropValException

指定レンディションプロパティ値が不正の場合

DbjIOException

Java クラスライブラリ固有の IO エラーの場合

DbjMasterRenditionNotSetException

マスタレンディションのレンディションタイプが設定されていなかった場合

DbjNotCheckOutException

チェックアウトされていないオブジェクトに対して仮のバージョン識別子が指定された場合

DbjObjectNotFoundExpection

指定オブジェクトが存在しなかった場合

DbjReferenceTypeMismatchException

リファレンスファイル管理機能で、オブジェクトのリファレンス種別とメソッドが要求するリファレンス種別が一致しない場合

DbjRenditionCountOutOfRangeExpection

指定レンディション数が範囲を超えた場合

DbjRenditionNotFoundExpection

指定レンディションが存在しなかった場合

DbjRenditionTypeDuplicatedException

指定レンディションタイプが重複していた場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

DbjTargetContentPathNotSetException

リファレンスファイル管理機能で、コンテンツ格納先ベースパスが設定されていなかった場合

DbjXmlParseException

XML 文書解析エラーの場合

IllegalArgumentExpection

引数に不正な値が指定されていた場合

NullPointerExpection

引数 uploadInfo が null の場合

## 4.4.40 writeProperties ( 文書空間オブジェクトのプロパティ値の設定 )

### (1) 機能

Proxy オブジェクトのアクセス対象である文書空間オブジェクトのプロパティ値を設定します。

形式 1 では、Proxy オブジェクトがアクセス対象とする文書空間オブジェクトのプロパティを、指定したプロパティ値集合で設定します。この場合、Proxy オブジェクトのプロパティ値集合は無視されます。形式 2 および形式 3 では、Proxy オブジェクトに設定されているプロパティ値集合で文書空間オブジェクトのプロパティを設定します。形式 2 では、DbjObj#readProperties メソッドで読み込んだプロパティ値集合のプロパティ値を書き換えて文書空間オブジェクトのプロパティ値を更新します。形式 3 では、Proxy オブジェクトに設定されているプロパティのうち、文書空間オブジェクトに反映するプロパティを限定して指定できます。なお、形式 3 の場合、指定した引数の値によって Proxy オブジェクトの保持するプロパティ値集合は変更されません。

### (2) 形式

#### (a) 形式 1

```
void writeProperties(
    DbjPropSet      propSet
)
```

#### (b) 形式 2

```
void writeProperties()
```

#### (c) 形式 3

```
void writeProperties(
    Collection      propDefs
)
```

### (3) 引数

propSet ( 入力 )

文書空間オブジェクトに設定するプロパティ値集合を指定します。null を指定すると例外がスローされます。

propDefs ( 入力 )

要素はプロパティ名 (String 型) です。文書空間オブジェクトに設定する対象になるプロパティ値のコレクションを指定します。Proxy オブジェクトのプロパティ値のコレクションに存在しないプロパティを指定した場合は無視されます。null を指定すると例外がスローされます。

### (4) 戻り値

なし

### (5) 例外

DbjAccessControlException

アクセス権エラーの場合

DbjACEOperationException

指定 ACE についてのエラーの場合

DbjCORBAException

CORBA エラーの場合

#### 4. 文書管理クラス詳細

DbjDBException

DB エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjIllegalPropValException

指定プロパティ値が不正の場合

DbjNotCheckOutException

チェックアウトされていないオブジェクトに対して仮のバージョン識別子が指定された場合

DbjObjectNotFoundException

指定オブジェクトが存在しなかった場合

DbjPublicACLNotFoundException

指定パブリック ACL が存在しなかった場合

DbjPublicACLOutOfRangeException

指定パブリック ACL 数が範囲を超えた場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

NullPointerException

引数 propSet , または propDefs が null の場合

#### 4.4.41 writeRenditionProperties ( レンディションプロパティの設定 )

##### (1) 機能

Proxy オブジェクトのアクセス対象である文書の指定レンディションについて、レンディションプロパティを設定します。レンディションプロパティを設定できるターゲットオブジェクトは、文書だけです。

##### (2) 形式

```
void writeRenditionProperties (
    String          renditionType,
    DbjPropSet     propSet
)
```

##### (3) 引数

renditionType ( 入力 )

設定対象のレンディションタイプを指定します。null を指定するとマスタレンディションが仮定されます。

propSet ( 入力 )

設定するレンディションのプロパティ値集合を指定します。null を指定すると例外がスローされます。

##### (4) 戻り値

なし

##### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjIllegalObjectTypeException  
指定オブジェクト種別が不適切の場合

DbjIllegalPropValException  
指定プロパティ値が不正の場合

DbjNotCheckOutException  
チェックアウトされていないオブジェクトに対して仮のバージョン識別子が指定された場合

DbjObjectNotFoundException  
指定オブジェクトが存在しなかった場合

DbjRenditionNotFoundException  
指定レンディションが存在しなかった場合

DbjSessionNotConnectException

#### 4. 文書管理クラス詳細

セッションが接続されていなかった場合

NullPointerException

引数 propSet が null の場合



## 4.5 DbjObjList インターフェース

---

DbjObjList インターフェースは、複数の文書空間オブジェクトに一括してアクセスするインターフェースです。要素は、DbjObj インターフェースです。

スーパーインターフェース

- java.util.List

以降、DbjObjList インターフェースのメソッドについて説明します。

## 4.5.1 getObj ( リスト要素の取得 )

### (1) 機能

インデックスで指定したリスト要素 ( DbjObj インターフェース ) を取得します。

### (2) 形式

```
DbjObj getObj (
    int    index
)
```

### (3) 引数

index ( 入力 )

要素のインデックスを指定します。

### (4) 戻り値

DbjObj インターフェース

### (5) 例外

ClassCastException

リスト要素が DbjObj インターフェースでなかった場合

IndexOutOfBoundsException

指定したインデックスが不正の場合

## 4.5.2 lock (アクセスロック種別の異なる複数文書空間オブジェクトインターフェースの取得)

### (1) 機能

アクセス対象の文書空間に対して、指定したアクセスロック種別のロックを保持する DbjObjList インターフェースを取得します。取得されたインターフェースには、要素の文書空間オブジェクトインターフェースのロックの指定よりも優先してロックが指定されます。なお、デフォルトでは、各要素の文書空間オブジェクトインターフェースのロック種別が有効になります。

### (2) 形式

```
DbjObjList lock(  
    int      lockType  
)
```

### (3) 引数

lockType (入力)

ロック種別として、次に示す定数のどちらかを指定します。

- DbjDef.LOCK\_READ  
read ロック
- DbjDef.LOCK\_WRITE  
write ロック

### (4) 戻り値

複数文書空間オブジェクトインターフェース (DbjObjList インターフェース)

### (5) 例外

なし

### 4.5.3 move ( 直接型リンクが設定されている文書空間オブジェクトの 一括移動 )

#### ( 1 ) 機能

リスト要素の文書空間オブジェクトに対してリンク元フォルダから直接型リンクが設定されている場合、そのリンクを解除して引数で指定したリンク元フォルダに直接型リンクを設定します。リスト要素の文書空間オブジェクトに直接型リンクが設定されていない場合は、指定したリンク元フォルダに直接型リンクを設定します。

#### ( 2 ) 形式

```
void move (
    DbjObj      parentFolder
)
```

#### ( 3 ) 引数

parentFolder ( 入力 )

要素オブジェクトが移動するリンク元フォルダを指定します。null を指定すると例外がスローされま  
す。

#### ( 4 ) 戻り値

なし

#### ( 5 ) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjIllegalObjectTypeException  
指定オブジェクト種別が不適切の場合

DbjNotCheckOutException  
チェックアウトされていないオブジェクトに対して仮のバージョン識別子が指定された場合

DbjObjectNotFoundException  
指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合

NullPointerException  
引数 parentFolder が null の場合

## 4.5.4 readProperties ( 複数の文書空間オブジェクトプロパティ値の一括取得 )

### (1) 機能

Proxy オブジェクトのアクセス対象である文書空間オブジェクトの指定プロパティを、リストの各要素の Proxy オブジェクトに読み込みます。リストの要素の各 DbjObj インターフェースに対して、同じ形式の DbjObj#readProperties メソッドを実行します。

形式 1 では、Proxy オブジェクトの保持しているプロパティ値集合でのすべてのプロパティを文書空間オブジェクトから読み込みます。形式 2 では、読み込み対象のプロパティのコレクションを指定します。

### (2) 形式

#### (a) 形式 1

```
void readProperties()
```

#### (b) 形式 2

```
void readProperties(
    Collection propDefs
)
```

### (3) 引数

propDefs ( 入力 )

要素はプロパティ名 (String 型) です。取得するプロパティのプロパティ名のコレクションを指定します。null を指定すると例外がスローされます。

### (4) 戻り値

なし

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjNotCheckOutException  
チェックアウトされていないオブジェクトに対して仮のバージョン識別子が指定された場合

DbjObjectNotFoundExpection  
指定オブジェクトが存在しなかった場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合

NullPointerException

#### 4. 文書管理クラス詳細

引数 propDefs が null の場合

## 4.5.5 removeObjects ( 文書空間オブジェクトの一括削除 )

### (1) 機能

リストオブジェクトの各要素の Proxy オブジェクトのアクセス対象の文書空間オブジェクトを一括して削除します。リストの要素の各 DbjObj インターフェースに対して、DbjObj#removeObject メソッドを実行します。

### (2) 形式

#### (a) 形式 1

```
List removeObjects()
```

#### (b) 形式 2

```
List removeObjects(
    boolean isAlsoChildren
)
```

### (3) 引数

isAlsoChildren( 入力 )

リンク先の文書空間オブジェクトの削除を指定します。true を指定すると、リンク先の文書空間オブジェクトが削除されます。この場合、直接型リンクが設定されている下位のすべての文書空間オブジェクトが削除されます。false を指定すると、リンク先の文書空間オブジェクトは削除されません。

### (4) 戻り値

削除されたリスト要素の各 DbjObj インターフェースのリスト

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjLastVersionException  
最後のバージョンを削除しようとした場合

DbjNotCheckOutException  
チェックアウトされていないオブジェクトに対して仮のバージョン識別子が指定された場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合

## 4.5.6 setPropSet (要素のプロパティ値集合の設定)

### (1) 機能

Proxy オブジェクトのプロパティ値集合の全要素に、指定したプロパティ値集合の内容をコピーして設定します。

### (2) 形式

```
void setPropSet (  
    DbjPropSet    propSet  
)
```

### (3) 引数

propSet (入力)

プロパティ値集合インターフェースを指定します。null を指定すると、オブジェクトのプロパティ値集合には空値が仮定されます。

### (4) 戻り値

なし

### (5) 例外

なし



## 4.5.7 writeProperties ( 複数の文書空間オブジェクトプロパティ値の一括設定 )

### (1) 機能

リスト要素の Proxy オブジェクトのアクセス対象である文書空間オブジェクトのプロパティを一括して設定します。

形式 1 では、要素のアクセス対象である文書空間オブジェクトのプロパティを、指定したプロパティ値集合で設定します。この場合、要素の Proxy オブジェクトのプロパティ値集合は無視されます。形式 2 および形式 3 では、要素のアクセス対象である文書空間オブジェクトのプロパティを各要素の Proxy オブジェクトに設定されたプロパティ値集合で設定します。形式 2 では、DbjObj#readProperties メソッドで読み込んだプロパティ値集合のプロパティ値を書き換えて文書空間オブジェクトのプロパティ値を更新します。形式 3 では、Proxy オブジェクトに設定されているプロパティのうち、要素のアクセス対象である文書空間オブジェクトに反映するプロパティを限定して指定できます。なお、形式 3 の場合、指定した引数の値によって Proxy オブジェクトの保持するプロパティ値集合は変更されません。

### (2) 形式

#### (a) 形式 1

```
void writeProperties(
    DbjPropSet    propSet
)
```

#### (b) 形式 2

```
void writeProperties()
```

#### (c) 形式 3

```
void writeProperties(
    Collection    propdef
)
```

### (3) 引数

propSet ( 入力 )

文書空間オブジェクトに設定するプロパティ値集合を指定します。null を指定すると例外がスローされます。

propDefs ( 入力 )

要素はプロパティ名 ( String 型 ) です。文書空間オブジェクトに設定する対象のプロパティ値のコレクションを指定します。null を指定すると例外がスローされます。

### (4) 戻り値

なし

### (5) 例外

DbjAccessControlException  
アクセス権エラーの場合

DbjACEOperationException  
指定 ACE についてのエラーの場合

DbjCORBAException

#### 4. 文書管理クラス詳細

CORBA エラーの場合

DbjDBException

DB エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjIllegalPropValException

指定プロパティ値が不正の場合

DbjNotCheckOutException

チェックアウトされていないオブジェクトに対して仮のバージョン識別子が指定された場合

DbjObjectNotFoundExpection

指定オブジェクトが存在しなかった場合

DbjPublicACLNotFoundExpection

指定パブリック ACL は存在しなかった場合

DbjPublicACLOutOfRangeException

指定パブリック ACL 数が範囲を超えた場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

NullPointerException

引数 propSet , または propDefs が null の場合

## 4.6 DbjSession インターフェース

---

DbjSession インターフェースは、Java クラスライブラリのセッションを扱います。Java クラスライブラリの一つのセッションオブジェクトは文書空間への一つの接続に対応し、その接続している期間をセッションといいます。DbjSession インターフェースは、同時に一つのセッションのセッション管理をサポートします。ユーザアプリケーションプログラムが複数のセッションを実現するためには、複数の DbjSession インターフェースを扱う必要があります。

以降、DbjSession インターフェースのメソッドについて説明します。

## 4.6.1 begin (トランザクションの開始)

### (1) 機能

トランザクションを開始します。このメソッドを二重に実行した場合は例外がスローされます。

### (2) 形式

```
void begin()
```

### (3) 引数

なし

### (4) 戻り値

なし

### (5) 例外

DbjCORBAException

CORBA エラーの場合

DbjDBException

DB エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

IllegalStateException

DbjSession#begin メソッドが二重に実行された場合

## 4.6.2 checkSession (セッションが有効かどうかのチェック)

### (1) 機能

接続中の文書空間に対して有効なセッションがあるかどうかを調べます。無効なセッションを次に示します。

- 切断 (logout) しているセッション
- 確立 (login) していないセッション
- 文書空間との接続がタイムアウトしているセッション

### (2) 形式

```
boolean checkSession()
```

### (3) 引数

なし

### (4) 戻り値

true

有効なセッションです。

false

無効なセッションです。

注

本メソッドはセッションのアイドル時間をリセットしません。そのため、セッションタイムアウトの設定をしている場合、次のメソッド発行までの間にセッションがタイムアウトする可能性があります。

### (5) 例外

なし

### 4.6.3 commit (トランザクションの確定)

#### (1) 機能

トランザクションを確定します。トランザクションが開始されていない場合、例外がスローされます。

#### (2) 形式

```
void commit()
```

#### (3) 引数

なし

#### (4) 戻り値

なし

#### (5) 例外

DbjCORBAException

CORBA エラーの場合

DbjDBException

DB エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

IllegalStateException

DbjSession#begin メソッドが実行されていないのに DbjSession#commit メソッドが実行された場合

## 4.6.4 getLoginUserInfo ( ユーザ情報の取得 )

### (1) 機能

ログインしているユーザのユーザ情報を取得します。

### (2) 形式

```
DbjPropSet getLoginUserInfo(  
    Collection propDefs  
)
```

### (3) 引数

propDefs ( 入力 )

要素はプロパティ名 ( String 型 ) です。取得したいユーザ情報プロパティのプロパティ名を指定します。

### (4) 戻り値

ユーザ情報プロパティ値集合

### (5) 例外

DbjAccessControlNotSupportedException

接続先の文書空間がアクセス制御動作モードに対応していなかった場合

DbjCORBAException

CORBA エラーの場合

DbjException

Java クラスライブラリ固有のエラーの場合

DbjSessionNotConnectException

セッションが接続されていない場合

NullPointerException

引数 propDefs が null の場合

## 4.6.5 getReferencePath (コンテンツ格納先ベースパスの取得)

### (1) 機能

リファレンスファイル管理機能を使用して、コンテンツを任意の領域に格納する場合、  
DbjSession#setReferencePath メソッドで設定したコンテンツ格納先ベースパスを取得します。

### (2) 形式

```
String getReferencePath()
```

### (3) 引数

なし

### (4) 戻り値

リファレンスファイル管理機能のコンテンツ格納先ベースパス

### (5) 例外

DbjException

Java クラスライブラリ固有のエラーの場合

DbjSessionNotConnectException

セッションが接続されていなかった場合



## 4.6.6 login ( ログイン )

### (1) 機能

文書空間にログインしてセッションを確立し、同時にユーザ識別子およびパスワードの認証を実行して、文書空間インターフェース ( DbjDocSpace インターフェース ) を取得します。このメソッドは、一つの文書空間にアクセスするほかのメソッドの実行前に実行する必要があります。確立しているセッションの有効期限は、DbjSession#logout メソッドを実行するまで、または文書空間への接続をタイムアウト ( DocumentBroker サーバで設定される ) するまでです。なお、ユーザ認証の失敗、および二重のログインがあった場合は、例外がスローされます。

### (2) 形式

```
DbjDocSpace login(
    String      userName,
    String      passWord
)
```

### (3) 引数

userName ( 入力 )

ログインするユーザのユーザ識別子を指定します。

passWord ( 入力 )

ログインするユーザのパスワードを指定します。

### (4) 戻り値

文書空間オブジェクトアクセスインターフェース ( DbjDocSpace インターフェース )

### (5) 例外

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjIllegalDocSpaceIdException  
接続先の文書空間識別子が不正の場合

DbjNotAuthenticatedException  
認証エラーの場合

DbjSessionOverflowException  
セッション数が同時に接続できる数を越えた場合

IllegalStateException  
すでにログインしていた場合

## 4.6.7 logout ( ログアウト )

### (1) 機能

文書空間からログアウトしてセッションを切断します。セッションの切断によって、ログインセッション内で継続されて使用される内部データ（検索結果キャッシュなど）は消去されます。また、切断するセッションに未確定のトランザクションがある場合は、自動的にトランザクションが取り消されて最後にトランザクションが確定された状態まで戻ります。なお、このメソッドのエラーは、エラーログファイルに出力されます。

### (2) 形式

```
void logout()
```

### (3) 引数

なし

### (4) 戻り値

なし

### (5) 例外

なし

## 4.6.8 rollback (トランザクションの取り消し)

### (1) 機能

トランザクションを取り消して、最後に DbjSession#commit メソッドが実行された状態まで戻します。  
トランザクションが開始されていなくても正常終了します。

### (2) 形式

```
void rollback()
```

### (3) 引数

なし

### (4) 戻り値

なし

### (5) 例外

DbjCORBAException  
CORBA エラーの場合

DbjDBException  
DB エラーの場合

DbjException  
Java クラスライブラリ固有のエラーの場合

DbjSessionNotConnectException  
セッションが接続されていなかった場合

## 4.6.9 setReferencePath (コンテンツ格納先ベースパスの設定)

### (1) 機能

リファレンスファイル管理機能を使用して、コンテンツを任意の領域に格納する場合に、コンテンツ格納先ベースパスを設定します。

### (2) 形式

```
void setReferencePath( String referencePath )
```

### (3) 引数

referencePath ( 入力 )

リファレンスファイル管理機能で、コンテンツの格納先となるベースパスを指定します。

### (4) 戻り値

なし

### (5) 例外

DbjException

Java クラスライブラリ固有のエラーの場合

DbjSessionNotConnectException

セッションが接続されていなかった場合

## 4.7 DbjVerObj インターフェース

---

DbjVerObj インターフェースは、一つの文書空間オブジェクトへのアクセスを扱うことに加えて、Proxy オブジェクトのターゲットオブジェクトであるバージョンオブジェクトのバージョン識別子を扱うインターフェースです。

スーパーインターフェース

- DbjObj

プロパティ一覧

DbjVerObj インターフェースが扱うプロパティと getter メソッドを次の表に示します。

表 4-3 DbjVerObj インターフェースで扱うプロパティ

プロパティ名	データ型	getter メソッド
バージョン識別子	String	getVersionId

以降、DbjVerObj インターフェースのメソッドについて説明します。

### 4.7.1 getVersionId ( 文書空間オブジェクトのバージョン識別子の取得 )

#### (1) 機能

文書空間オブジェクトに設定されている Proxy オブジェクトのターゲットオブジェクトであるバージョンオブジェクトのバージョン識別子のプロパティ値 (String 型) を取得します。

#### (2) 形式

```
String getVersionId()
```

#### (3) 引数

なし

#### (4) 戻り値

バージョンオブジェクトのバージョン識別子のプロパティ値 (versionId フィールドの値)

#### (5) 例外

なし

## 4.8 DbjVerObjList インターフェース

---

DbjVerObjList インターフェースは、DbjVerObj インターフェースを要素とするリストインターフェースを扱います。

スーパーインターフェース

- DbjObjList

以降、DbjVerObjList インターフェースのメソッドについて説明します。

### 4.8.1 getVerObj ( 要素の DbjVerObj インターフェースの取得 )

#### (1) 機能

インデクスで指定された要素の DbjVerObj インターフェースを取得します。

#### (2) 形式

```
DbjVerObj getVerObj(  
    int    index  
)
```

#### (3) 引数

index ( 入力 )

要素インデクスを指定します。

#### (4) 戻り値

要素の DbjVerObj インターフェース

#### (5) 例外

ClassCastException

指定要素が DbjObj インターフェースでなかった場合

IndexOutOfBoundsException

指定したインデクスが不正の場合



## 4.8.2 getVersionIdList ( 文書空間オブジェクトのバージョン識別子リストの取得 )

### (1) 機能

文書空間オブジェクトが保持する要素すべてについて、Proxy オブジェクトのターゲットオブジェクトであるバージョンオブジェクトのバージョン識別子のプロパティ値をリストで取得します。

### (2) 形式

```
List getVersionIdList()
```

### (3) 引数

なし

### (4) 戻り値

バージョンオブジェクトのバージョン識別子のプロパティ値 ( versionId フィールドの値 ) のリスト ( String 型のリスト )

### (5) 例外

ClassCastException

指定要素が DbjObj インターフェースでなかった場合

## 4.9 DbjXmlTranslator インターフェース

---

DbjXmlTranslator インターフェースは、XML 文書の XML プロパティマッピング機能を扱います。

以降、DbjXmlTranslator インターフェースのメソッドについて説明します。

## 4.9.1 getMappedProperties ( XML 文書の XML プロパティマッピング情報の作成 )

### (1) 機能

XML 文書を解析して、マッピング定義情報に従って XML 文書からプロパティを抽出して文書のプロパティなどの情報を作成します。

### (2) 形式

```
DbjMappedProp getMappedProperties(
    String      filePath,
    int         parseLevel,
    String      mappingId
)
```

### (3) 引数

filePath ( 入力 )

XML 文書ファイルのローカルパスを絶対パスで指定します。null を指定すると例外がスローされます。

parseLevel ( 入力 )

構文解析のレベルを指定します。次のどれかを指定できます。

- DbjDef.XMLPARSE\_NO\_EXTERNAL\_ENTITIES  
DTD の解析をしないで、外部エンティティは無視します。
- DbjDef.XMLPARSE\_WITH\_EXTERNAL\_ENTITIES  
DTD を含めた構文解析をして外部エンティティを読み込みますが、検証を実行しません。
- DbjDef.XMLPARSE\_VALIDATE  
DTD を使用して検証を実行します

mappingId ( 入力 )

マッピング定義名を指定します。null を指定すると例外がスローされます。

### (4) 戻り値

DbjMappedProp インターフェース

### (5) 例外

DbjException

Java クラスライブラリ固有のエラーの場合

DbjIOException

Java クラスライブラリ固有の IO エラーの場合

DbjXmlParseException

XML ファイルの構文解析エラーの場合

NullPointerException

引数 filePath , または mappingId が null の場合



# 5

## メタクラス詳細

この章では、メタクラスのインターフェース、およびメソッドについて説明します。

---

5.1 DbjClassDesc インターフェース

---

5.2 DbjMeta インターフェース

---

5.3 DbjMetaManager インターフェース

---

5.4 DbjPropDesc インターフェース

---

## 5.1 DbjClassDesc インターフェース

---

DbjClassDesc インターフェースは、一つの文書空間オブジェクトクラスのメタ情報（クラスディスクリプション）を扱うインターフェースです。

以降、DbjClassDesc インターフェースのメソッドについて説明します。

### 5.1.1 getName (クラス名の取得)

#### (1) 機能

クラス名を取得します。

#### (2) 形式

```
String getName()
```

#### (3) 引数

なし

#### (4) 戻り値

クラス名

#### (5) 例外

なし

## 5.1.2 `getProperties` (プロパティディスクリプションの取得)

### (1) 機能

クラスが保持するプロパティディスクリプションをすべて取得します。

### (2) 形式

```
List getProperties()
```

### (3) 引数

なし

### (4) 戻り値

プロパティディスクリプション (`DbjPropDesc` インターフェース) を要素とするリスト

### (5) 例外

なし



### 5.1.3 getSubClasses (サブクラスのクラスディスクリプションの取得)

#### (1) 機能

直接のサブクラスのクラスディスクリプションをすべて取得します。

サブクラスがない場合は、要素が空のリストが返却されます。

#### (2) 形式

```
List getSubClasses()
```

#### (3) 引数

なし

#### (4) 戻り値

クラスディスクリプション (DbjClassDesc インターフェース) を要素とするリスト

#### (5) 例外

なし

### 5.1.4 getSuperClass ( スーパークラスのクラスディスクリプションの取得 )

#### (1) 機能

直接のスーパークラスのクラスディスクリプションを取得します。

スーパークラスがない場合は、null が返却されます。

#### (2) 形式

```
DbjClassDesc getSuperClass()
```

#### (3) 引数

なし

#### (4) 戻り値

スーパークラスのクラスディスクリプション ( DbjClassDesc インターフェース )

#### (5) 例外

なし

## 5.2 DbjMeta インターフェース

---

DbjMeta インターフェースは、一つの文書空間のメタ情報を扱うインターフェースです。

以降、DbjMeta インターフェースのメソッドについて説明します。

## 5.2.1 getClassDesc ( 指定したクラスのクラスディスクリプションの取得 )

### (1) 機能

指定したクラスのクラスディスクリプションを取得します。

### (2) 形式

```
DbjClassDesc getClassDesc (
    String          className
)
```

### (3) 引数

className ( 入力 )

クラス名を指定します。

null を指定した場合は、例外がスローされます。

### (4) 戻り値

指定したクラスのクラスディスクリプション ( DbjClassDesc インターフェース )

### (5) 例外

IllegalArgumentException

指定したクラスがなかった場合

NullPointerException

引数 className に null を指定した場合

## 5.2.2 getDocSpaceId ( 文書空間識別子の取得 )

### (1) 機能

文書空間識別子を取得します。

### (2) 形式

```
String getDocSpaceId()
```

### (3) 引数

なし

### (4) 戻り値

文書空間識別子

### (5) 例外

なし

### 5.2.3 getExtFromRenditionType ( レンディションタイプに対応する拡張子の取得 )

#### (1) 機能

レンディション定義ファイルの定義内容に従って、指定したレンディションタイプに対応する拡張子を取得します。

指定したレンディションタイプがレンディション定義ファイルに定義されていない場合は、null が返却されます。また、指定したレンディションタイプに対応する拡張子が複数定義されている場合は、その中のどれかの拡張子が返却されます。

#### (2) 形式

```
String getExtFromRenditionType(  
    String renditionType  
)
```

#### (3) 引数

renditionType ( 入力 )

レンディションタイプを指定します。

null を指定した場合は、例外がスローされます。

#### (4) 戻り値

指定したレンディションタイプに対応する拡張子

#### (5) 例外

NullPointerException

引数 renditionType に null を指定した場合

## 5.2.4 getPropDataType ( 指定したプロパティのデータ型の取得 )

### (1) 機能

指定したプロパティのデータ型を取得します。

### (2) 形式

```
int getPropDataType(  
    String      propName  
)
```

### (3) 引数

propName ( 入力 )

プロパティ名を指定します。

null を指定した場合は、例外がスローされます。

### (4) 戻り値

指定したプロパティのデータ型が、次に示す定数で返却されます。

- DbjDef.DATATYPE\_BOOL  
 BOOL 型
- DbjDef.DATATYPE\_INT  
 INT 型
- DbjDef.DATATYPE\_STR  
 STR 型
- DbjDef.DATATYPE\_STRLIST  
 STRLIST 型
- DbjDef.DATATYPE\_VARRAY  
 VARRAY 型

### (5) 例外

IllegalArgumentException

指定したプロパティがなかった場合

NullPointerException

引数 propName に null を指定した場合

## 5.2.5 getPropDesc ( 指定したプロパティのプロパティディスクリプションの取得 )

### (1) 機能

指定したプロパティのプロパティディスクリプションを取得します。

### (2) 形式

```
DbjPropDesc getPropDesc (  
    String      propName  
)
```

### (3) 引数

propName ( 入力 )

プロパティ名を指定します。

null を指定した場合は、例外がスローされます。

### (4) 戻り値

指定したプロパティのプロパティディスクリプション ( DbjPropDesc インターフェース )

### (5) 例外

IllegalArgumentException

指定したプロパティがなかった場合

NullPointerException

引数 propName に null を指定した場合



## 5.2.6 getRenditionType ( 拡張子に対応するレンディションタイプの取得 )

### (1) 機能

レンディション定義ファイルの定義内容に従って、指定した拡張子に対応するレンディションタイプを取得します。

指定した拡張子に対応するレンディションタイプがレンディション定義ファイルに定義されていない場合は、null が返却されます。

### (2) 形式

```
String getRenditionType(  
    String extension  
)
```

### (3) 引数

extension ( 入力 )

ファイルの拡張子を指定します。指定する拡張子の先頭に「.」(ピリオド)は不要です。  
null を指定した場合は、例外がスローされます。

### (4) 戻り値

指定した拡張子に対応するレンディションタイプ (「MIME::」は含みません)

### (5) 例外

NullPointerException

引数 extension に null を指定した場合

## 5.3 DbjMetaManager インターフェース

---

DbjMetaManager インターフェースは、文書空間のメタ情報を管理するためのインターフェースです。

以降、DbjMetaManager インターフェースのメソッドについて説明します。

### 5.3.1 getMeta (メタ情報の取得)

#### (1) 機能

指定した文書空間のメタ情報を取得します。メタ情報は DbjMeta インターフェースで返却されます。

なお、メタ情報を取得するには、クラス定義情報ファイルをあらかじめ作成して、決められたディレクトリ下に格納しておく必要があります。クラス定義情報ファイルの作成方法および格納先については、マニュアル「DocumentBroker Version 3 クラスライブラリ Java 解説」の「7.6.1 クラス定義情報ファイル」を参照してください。

#### (2) 形式

```
DbjMeta getMeta(  
    String docspaceId  
)
```

#### (3) 引数

docspaceId (入力)

文書空間識別子を GUID 文字列で指定します。GUID 文字列は、16 進数「X」によって、「XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX」(8 けた -4 けた -4 けた -4 けた -12 けた)の形式で表されます。「X」は、0 ~ 9, a ~ f (小文字), および A ~ F (大文字) のどれかです。null を指定した場合は、例外がスローされます。

#### (4) 戻り値

指定した文書空間のメタ情報 (DbjMeta インターフェース)

#### (5) 例外

IllegalArgumentException

指定した文書空間がなかった場合

NullPointerException

引数 docspaceId に null を指定した場合

## 5.4 DbjPropDesc インターフェース

---

DbjPropDesc インターフェースは、文書空間オブジェクトの一つのプロパティのメタ情報（プロパティディスクリプション）を扱うインターフェースです。

以降、DbjPropDesc インターフェースのメソッドについて説明します。

## 5.4.1 getDataTyPe ( プロパティのデータ型の取得 )

### (1) 機能

プロパティのデータ型を取得します。

### (2) 形式

```
int getDataTyPe ()
```

### (3) 引数

なし

### (4) 戻り値

プロパティのデータ型が、次に示す定数で返却されます。

- DbjDef.DATATYPE\_BOOL  
BOOL 型
- DbjDef.DATATYPE\_INT  
INT 型
- DbjDef.DATATYPE\_STR  
STR 型
- DbjDef.DATATYPE\_STRLIST  
STRLIST 型
- DbjDef.DATATYPE\_VARRAY  
VARRAY 型

### (5) 例外

なし

## 5.4.2 getName (プロパティ名の取得)

### (1) 機能

プロパティ名を取得します。

### (2) 形式

```
String getName()
```

### (3) 引数

なし

### (4) 戻り値

プロパティ名

### (5) 例外

なし

### 5.4.3 getVArrayClass ( VARRAY 型プロパティを扱うクラスのクラス ディスクリプションの取得 )

#### (1) 機能

VARRAY 型プロパティを扱うクラスのクラスディスクリプションを取得します。

VARRAY 型プロパティを扱うクラスでない場合は、null が返却されます。

#### (2) 形式

```
DbjClassDesc getVArrayClass ()
```

#### (3) 引数

なし

#### (4) 戻り値

VARRAY 型プロパティを扱うクラスのクラスディスクリプション ( DbjClassDesc インターフェース )

#### (5) 例外

なし





# 6

## 例外クラス詳細

この章では、例外クラスのクラスについて説明します。

- 
- 6.1 例外クラスの詳細

---

  - 6.2 DbjAccessControlException クラス

---

  - 6.3 DbjAccessControlNotSupportedException クラス

---

  - 6.4 DbjACEOperationException クラス

---

  - 6.5 DbjACLOutOfRangeException クラス

---

  - 6.6 DbjAlreadyCheckOutException クラス

---

  - 6.7 DbjCheckOutException クラス

---

  - 6.8 DbjContentNotRegisteredException クラス

---

  - 6.9 DbjContentTypeMismatchException クラス

---

  - 6.10 DbjCORBAException クラス

---

  - 6.11 DbjDBDeadLockException クラス

---

  - 6.12 DbjDBException クラス

---

  - 6.13 DbjDBLockTimeoutException クラス

---

  - 6.14 DbjDisconnectedSessionException クラス

---

  - 6.15 DbjError クラス

---

  - 6.16 DbjException クラス

---

  - 6.17 DbjFileAccessException クラス

---

  - 6.18 DbjFileNotFoundException クラス

---

  - 6.19 DbjFileReferenceCurrentContentNotfoundException クラス

---

  - 6.20 DbjFileReferenceMismatchStatusException クラス

---

  - 6.21 DbjFileReferenceOperationFailedException クラス

---

  - 6.22 DbjIllegalCacheStartIndexException クラス

---

6.23	DbjIllegalDocSpaceIdException	クラス
6.24	DbjIllegalObjectTypeException	クラス
6.25	DbjIllegalPropValException	クラス
6.26	DbjInitializeError	クラス
6.27	DbjInternalError	クラス
6.28	DbjIOException	クラス
6.29	DbjIsMasterRenditionException	クラス
6.30	DbjLastVersionException	クラス
6.31	DbjMasterRenditionNotSetException	クラス
6.32	DbjNotAuthenticatedException	クラス
6.33	DbjNotCheckOutException	クラス
6.34	DbjNotLoginException	クラス
6.35	DbjObjectNotFoundException	クラス
6.36	DbjOutOfMemoryError	クラス
6.37	DbjPublicACLAlreadyBoundException	クラス
6.38	DbjPublicACLNotBoundException	クラス
6.39	DbjPublicACLNotFoundException	クラス
6.40	DbjPublicACLOperationException	クラス
6.41	DbjPublicACLOutOfRangeException	クラス
6.42	DbjReferenceTypeMismatchException	クラス
6.43	DbjRenditionCountOutOfRangeException	クラス
6.44	DbjRenditionIsEmptyException	クラス
6.45	DbjRenditionNotConvertedException	クラス
6.46	DbjRenditionNotFoundException	クラス
6.47	DbjRenditionTypeDuplicatedException	クラス
6.48	DbjSessionException	クラス
6.49	DbjSessionNotConnectException	クラス
6.50	DbjSessionOverflowException	クラス
6.51	DbjSubjectLengthOutOfRangeException	クラス
6.52	DbjTargetContentPathNotSetException	クラス
6.53	DbjUnknownError	クラス

---

6.54 DbjXmlParseException クラス

---

## 6.1 例外クラスの詳細

---

Java クラスライブラリの API でエラーが発生すると、例外がスローされます。例外クラスは、Java クラスライブラリ固有の例外を扱うクラス群です。ここでは、それぞれの例外クラスの機能、例外クラスのスーパークラス、メソッド（コンストラクタ）一覧を説明します。なお、例外クラスのコンストラクタについては、「1.5.3 例外クラスのコンストラクタ」を参照してください。

## 6.2 DbjAccessControlException クラス

---

DbjAccessControlException クラスは、ユーザのメソッドの実行時に、その操作に対してユーザにアクセス権がない場合のエラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjAccessControlException()

DbjAccessControlException(String s)

## 6.3 DbjAccessControlNotSupportedException クラス

---

DbjAccessControlNotSupportedException クラスは、文書空間がアクセス制御機能に対応していない場合のエラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjAccessControlNotSupportedException()

DbjAccessControlNotSupportedException(String s)

## 6.4 DbjACEOperationException クラス

---

DbjACEOperationException クラスは、ACE の操作エラーを表すスーパークラスです。特定の ACE 操作のエラーは、サブクラスがスローされます。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjACEOperationException()

DbjACEOperationException(String s)

## 6.5 DbjACLOutOfRangeException クラス

---

DbjACLOutOfRangeException クラスは、文書空間オブジェクトの ACE の個数が制限値を超えて追加された場合のエラーを表します。

### (1) スーパークラス

DbjACEOperationException

### (2) コンストラクター一覧

DbjACLOutOfRangeException()

DbjACLOutOfRangeException(String s)



## 6.6 DbjAlreadyCheckOutException クラス

---

DbjAlreadyCheckOutException クラスは、すでにチェックアウトされている場合のエラーを表します。

### (1) スーパークラス

DbjCheckOutException

### (2) コンストラクター一覧

DbjAlreadyCheckOutException()

DbjAlreadyCheckOutException(String s)

## 6.7 DbjCheckOutException クラス

---

DbjCheckOutException クラスは、チェックアウトについてのエラーを表すスーパークラスです。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjCheckOutException()

DbjCheckOutException(String s)

## 6.8 DbjContentNotRegisteredException クラス

---

DbjContentNotRegisteredException クラスは、リファレンスファイル管理機能を使用したオブジェクトをローカルファイルにダウンロードする場合に、オブジェクトにコンテンツが登録されていないときのエラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjContentNotRegisteredException()

DbjContentNotRegisteredException(String s)

## 6.9 DbjContentTypeMismatchException クラス

---

DbjContentTypeMismatchException クラスは、リファレンスファイル管理機能で、オブジェクトのコンテンツ種別とメソッドが要求するコンテンツ種別が一致しない場合のエラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjContentTypeMismatchException()

DbjContentTypeMismatchException(String s)

## 6.10 DbjCORBAException クラス

---

DbjCORBAException クラスは、CORBA エラー（TPBroker でのエラー）を表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjCORBAException()

DbjCORBAException(String s)

## 6.11 DbjDBDeadLockException クラス

---

DbjDBDeadLockException クラスは、DB のデッドロックエラーを表します。

### (1) スーパークラス

DbjDBException

### (2) コンストラクター一覧

DbjDBDeadLockException()

DbjDBDeadLockException(String s)

## 6.12 DbjDBException クラス

---

DbjDBException クラスは、DB エラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjDBException()

DbjDBException(String s)

## 6.13 DbjDBLockTimeoutException クラス

---

DbjDBLockTimeoutException クラスは、DB のロックタイムアウトエラーを表します。

### (1) スーパークラス

DbjDBException

### (2) コンストラクター一覧

DbjDBLockTimeoutException()

DbjDBLockTimeoutException(String s)



## 6.14 DbjDisconnectedSessionException クラス

---

DbjDisconnectedSessionException クラスは、文書空間との接続が切断された場合のエラーを表します。

### (1) スーパークラス

DbjSessionNotConnectException

### (2) コンストラクター一覧

DbjDisconnectedSessionException()

DbjDisconnectedSessionException(String s)

## 6.15 DbjError クラス

---

DbjError クラスは `java.lang.Error` クラスを継承し、Java クラスライブラリ固有の致命的なエラーを表すスーパークラスです。メモリ不足、Java クラスライブラリの初期化エラー、Java クラスライブラリ内で予期しない内部エラーが発生した場合のエラーなど、ユーザアプリケーションプログラムで対処できないエラーを表します。

DbjError クラスに関連付けられたエラーメッセージは次の内容を含みます。

- エラー内容、または原因コード
- エラー発生メソッド
- エラー発生箇所（ソース行番号、ソースファイル名）

### (1) スーパークラス

`java.lang.Error`

### (2) コンストラクター一覧

`DbjError()`

`DbjError(String s)`

## 6.16 DbjException クラス

---

DbjException クラスは `java.lang.Exception` クラスを継承し、Java クラスライブラリ固有のエラーを表すスーパークラスです。

### (1) スーパークラス

`java.lang.Exception`

### (2) コンストラクター一覧

`DbjException()`

`DbjException(String s)`

## 6.17 DbjFileAccessException クラス

---

DbjFileAccessException クラスは、ファイルのアクセス権がない場合のエラーを表します。

### (1) スーパークラス

DbjIOException

### (2) コンストラクター一覧

DbjFileAccessException()

DbjFileAccessException(String s)

## 6.18 DbjFileNotFoundException クラス

---

DbjFileNotFoundException クラスは、指定ファイルが存在しない場合のエラーを表します。

### (1) スーパークラス

DbjIOException

### (2) コンストラクター一覧

DbjFileNotFoundException()

DbjFileNotFoundException(String s)

## 6.19 DbjFileReferenceCurrentContentNotFoundExcepti on クラス

---

DbjFileReferenceCurrentContentNotFoundException クラスは、リファレンスファイル管理機能を使用したバージョン付きの文書オブジェクトで、カレントバージョンにコンテンツが存在しない場合のエラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjFileReferenceCurrentContentNotFoundException()

DbjFileReferenceCurrentContentNotFoundException(String s)

## 6.20 DbjFileReferenceMismatchStatusException クラス

---

DbjFileReferenceMismatchStatusException クラスは、リファレンスファイル管理機能を使用するサーバでのコンテンツ操作でエラーが発生し、オブジェクトとコンテンツが不整合な状態になった場合のエラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjFileReferenceMismatchStatusException()

DbjFileReferenceMismatchStatusException(String s)

## 6.21 DbjFileReferenceOperationFailedException クラス

---

DbjFileReferenceOperationFailedException クラスは、リファレンスファイル管理機能を使用するサーバでのコンテンツ操作が失敗した場合のエラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjFileReferenceOperationFailedException()

DbjFileReferenceOperationFailedException(String s)



## 6.22 DbjIllegalCacheStartIndexException クラス

---

DbjIllegalCacheStartIndexException クラスは、キャッシュ検索時に検索結果より大きい値を DbjFetchInfo インターフェースの startIndex プロパティへ指定した場合のエラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjIllegalCacheStartIndexException()

DbjIllegalCacheStartIndexException(String s)

## 6.23 DbjIllegalDocSpaceIdException クラス

---

DbjIllegalDocSpaceIdException クラスは、不正な文書空間識別子が指定された場合のエラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjIllegalDocSpaceIdException()

DbjIllegalDocSpaceIdException(String s)

## 6.24 DbjIllegalObjectTypeException クラス

---

DbjIllegalObjectTypeException クラスは、メソッドによる操作に不適切な文書空間オブジェクト種別が指定された場合のエラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjIllegalObjectTypeException()

DbjIllegalObjectTypeException(String s)

## 6.25 DbjIllegalPropValException クラス

---

DbjIllegalPropValException クラスは、不正なプロパティ値を指定した場合のエラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjIllegalPropValException()

DbjIllegalPropValException(String s)

## 6.26 DbjInitializeError クラス

---

DbjInitializeError クラスは、Java クラスライブラリの初期化を実行した時のエラーを表します。

### (1) スーパークラス

DbjError

### (2) コンストラクター一覧

DbjInitializeError()

DbjInitializeError(String s)

## 6.27 DbjInternalError クラス

---

DbjInternalError クラスは、Java クラスライブラリの内部エラーを表します。

### (1) スーパークラス

DbjError

### (2) コンストラクター一覧

DbjInternalError()

DbjInternalError(String s)

## 6.28 DbjIOException クラス

---

DbjIOException クラスは、Java クラスライブラリ固有の IO エラーを表すスーパークラスです。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjIOException()

DbjIOException(String s)

## 6.29 DbjsMasterRenditionException クラス

---

DbjsMasterRenditionException クラスは、マスタレンディションが指定された場合のエラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjsMasterRenditionException()

DbjsMasterRenditionException(String s)



## 6.30 DbjLastVersionException クラス

---

DbjLastVersionException クラスは、操作対象となるバージョン付きオブジェクトの最後の 1 バージョンを削除しようとした場合のエラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjLastVersionException()

DbjLastVersionException(String s)

## 6.31 DbjMasterRenditionNotSetException クラス

---

DbjMasterRenditionNotSetException クラスは、マスタレンディションのレンディションタイプが設定されていない場合のエラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjMasterRenditionNotSetException()

DbjMasterRenditionNotSetException(String s)

## 6.32 DbjNotAuthenticatedException クラス

---

DbjNotAuthenticatedException クラスは、ユーザ認証エラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjNotAuthenticatedException()

DbjNotAuthenticatedException(String s)

## 6.33 DbjNotCheckOutException クラス

---

DbjNotCheckOutException クラスは、チェックアウトされていない場合のエラーを表します。

### (1) スーパークラス

DbjCheckOutException

### (2) コンストラクター一覧

DbjNotCheckOutException()

DbjNotCheckOutException(String s)

## 6.34 DbjNotLoginException クラス

---

DbjNotLoginException クラスは、文書空間にログインされていない場合のエラーを表します。

### (1) スーパークラス

DbjSessionNotConnectException

### (2) コンストラクター一覧

DbjNotLoginException()

DbjNotLoginException(String s)

## 6.35 DbjObjectNotFoundException クラス

---

DbjObjectNotFoundException クラスは、操作対象となる文書空間オブジェクトがメソッド実行にすでに削除されていた場合、または存在しない場合のエラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjObjectNotFoundException()

DbjObjectNotFoundException(String s)

## 6.36 DbjOutOfMemoryError クラス

---

DbjOutOfMemoryError クラスは、メモリ不足が発生したことを表します。

### (1) スーパークラス

DbjError

### (2) コンストラクター一覧

DbjOutOfMemoryError()

DbjOutOfMemoryError(String s)

## 6.37 DbjPublicACLAlreadyBoundException クラス

---

DbjPublicACLAlreadyBoundException クラスは、指定したパブリック ACL がすでにバインドされている場合のエラーを表します。

### (1) スーパークラス

DbjPublicACLOperationException

### (2) コンストラクター一覧

DbjPublicACLAlreadyBoundException()

DbjPublicACLAlreadyBoundException(String s)



## 6.38 DbjPublicACLNotBoundException クラス

---

DbjPublicACLNotBoundException クラスは、指定したパブリック ACL がバインドされていない場合のエラーを表します。

### (1) スーパークラス

DbjPublicACLOperationException

### (2) コンストラクター一覧

DbjPublicACLNotBoundException()

DbjPublicACLNotBoundException(String s)

## 6.39 DbjPublicACLNotFoundException クラス

---

DbjPublicACLNotFoundException クラスは、指定したパブリック ACL が存在しない場合のエラーを表します。

### (1) スーパークラス

DbjPublicACLOperationException

### (2) コンストラクター一覧

DbjPublicACLNotFoundException()

DbjPublicACLNotFoundException(String s)

## 6.40 DbjPublicACLOperationException クラス

---

DbjPublicACLOperationException クラスは、パブリック ACL の操作エラーを表すスーパークラスです。特定のパブリック ACL 操作のエラーには、サブクラスがスローされます。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjPublicACLOperationException()

DbjPublicACLOperationException(String s)

## 6.41 DbjPublicACLOutOfRangeException クラス

---

DbjPublicACLOutOfRangeException クラスは、文書空間オブジェクトのパブリック ACL の個数が制限値を超えて追加された場合のエラーを表します。

### (1) スーパークラス

DbjPublicACLOperationException

### (2) コンストラクター一覧

DbjPublicACLOutOfRangeException()

DbjPublicACLOutOfRangeException(String s)

## 6.42 DbjReferenceTypeMismatchException クラス

---

DbjReferenceTypeMismatchException クラスは、リファレンスファイル管理機能で、オブジェクトのリファレンス種別とメソッドが要求するリファレンス種別が一致しない場合のエラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjReferenceTypeMismatchException()

DbjReferenceTypeMismatchException(String s)

## 6.43 DbjRenditionCountOutOfRangeException クラス

---

DbjRenditionCountOutOfRangeException クラスは、レンディション数が制限値を超えた場合のエラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjRenditionCountOutOfRangeException()

DbjRenditionCountOutOfRangeException(String s)

## 6.44 DbjRenditionIsEmptyException クラス

---

DbjRenditionIsEmptyException クラスは、レンディションが空の場合のエラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjRenditionIsEmptyException()

DbjRenditionIsEmptyException(String s)

## 6.45 DbjRenditionNotConvertedException クラス

---

DbjRenditionNotConvertedException クラスは、レンディションが更新処理されていない場合のエラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjRenditionNotConvertedException()

DbjRenditionNotConvertedException(String s)



## 6.46 DbjRenditionNotFoundException クラス

---

DbjRenditionNotFoundException クラスは、指定レンディションが存在しない場合のエラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjRenditionNotFoundException()

DbjRenditionNotFoundException(String s)

## 6.47 DbjRenditionTypeDuplicatedException クラス

---

DbjRenditionTypeDuplicatedException クラスは、レンディションタイプが重複している場合のエラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjRenditionTypeDuplicatedException()

DbjRenditionTypeDuplicatedException(String s)

## 6.48 DbjSessionException クラス

---

DbjSessionException クラスは、セッションについてのエラーを表すスーパークラスです。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjSessionException()

DbjSessionException(String s)

## 6.49 DbjSessionNotConnectException クラス

---

DbjSessionNotConnectException クラスは、文書空間と接続されていない場合のエラーを表すスーパークラスです。

### (1) スーパークラス

DbjSessionException

### (2) コンストラクター一覧

DbjSessionNotConnectException()

DbjSessionNotConnectException(String s)

## 6.50 DbjSessionOverflowException クラス

---

DbjSessionOverflowException クラスは、文書空間への同時接続数が制限値を超えた場合のエラーを表します。

### (1) スーパークラス

DbjSessionException

### (2) コンストラクター一覧

DbjSessionOverflowException()

DbjSessionOverflowException(String s)

## 6.51 DbjSubjectLengthOutOfRangeException クラス

---

DbjSubjectLengthOutOfRangeException クラスは、ACE のサブジェクトの長さが制限値を超えた場合のエラーを表します。

### (1) スーパークラス

DbjACEOperationException

### (2) コンストラクター一覧

DbjSubjectLengthOutOfRangeException()

DbjSubjectLengthOutOfRangeException(String s)

## 6.52 DbjTargetContentPathNotSetException クラス

---

DbjTargetContentPathNotSetException クラスは、リファレンスファイル管理機能で、コンテンツ格納先ベースパスが設定されていない場合のエラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjTargetContentPathNotSetException()

DbjTargetContentPathNotSetException(String s)

## 6.53 DbjUnknownError クラス

---

DbjUnknownError クラスは、Java クラスライブラリ内での未知のエラーの発生を表します。

### (1) スーパークラス

DbjError

### (2) コンストラクター一覧

DbjUnknownError()

DbjUnknownError(String s)



## 6.54 DbjXmlParseException クラス

---

DbjXmlParseException クラスは、XML 文書の構文解析エラーを表します。

### (1) スーパークラス

DbjException

### (2) コンストラクター一覧

DbjXmlParseException()

DbjXmlParseException(String s)



# 7

## 定数定義クラス詳細

この章では、定数定義クラスの定義内容について説明します。

---

7.1 DbjDef クラス

---

7.2 DbjTraceDef クラス

---

## 7.1 DbjDef クラス

---

DbjDef クラスは、Java クラスライブラリで使用する定数を定義したクラスです。

以降、DbjDef クラスの定数定義内容について、カテゴリごとに説明します。

なお、DbjDef クラスで定義されている定数のアクセス指定子は、すべて「public」です。

### 7.1.1 Category : ACL

検索実行時のアクセス制御モードを表す定数です。アクセス制御機能付き検索を実行するか、実行しないかを表します。

検索実行時のアクセス制御モードを表す定数を表 7-1 に示します。

表 7-1 検索実行時のアクセス制御モードを表す定数

定数	データ型	意味
WITH_ACL	int 型	アクセス制御機能付き検索モード
WITHOUT_ACL	int 型	アクセス制御機能なし検索モード

## 7.1.2 Category : DATATYPE

文書空間オブジェクトのプロパティのデータ型を表す定数です。

文書空間オブジェクトのプロパティのデータ型を表す定数を表 7-2 に示します。

表 7-2 文書空間オブジェクトのプロパティのデータ型を表す定数

定数	データ型	意味
DATATYPE_BOOL	int 型	BOOL 型
DATATYPE_INT	int 型	INT 型
DATATYPE_STR	int 型	STR 型
DATATYPE_STRLIST	int 型	STRLIST 型
DATATYPE_VARRAY	int 型	VARRAY 型
DATATYPE_UNKNOWN	int 型	未サポートのデータ型, またはデータ型不明

### 7.1.3 Category : DMA BOOLEAN

BOOL 型を表す定数です。

BOOL 型を表す定数を表 7-3 に示します。

表 7-3 BOOL 型を表す定数

定数	データ型	意味
DMA_TRUE	int 型	真
DMA_FALSE	int 型	偽
DMA_UNKNOWN	int 型	不定

### 7.1.4 Category : INDEXTYPE

全文検索インデクスの種別を表す定数です。

全文検索インデクスの種別を表す定数を表 7-4 に示します。

表 7-4 全文検索インデクスの種別を表す定数

定数	データ型	意味
INDEXTYPE_NOTHING	int 型	全文検索インデクスを作成しません。インデクスデータは出力されません。
INDEXTYPE_PLAIN	int 型	構造指定検索なしの全文検索インデクスを作成します。プレーンテキスト形式のインデクスデータが出力されます。
INDEXTYPE_STRUCTURED	int 型	構造指定検索用の全文検索インデクスを作成します。構造指定検索用のインデクスデータ (ESIS-B 形式のデータ) が出力されます。



### 7.1.5 Category : LINK

リンク種別を表す定数です。

リンク種別を表す定数を表 7-5 に示します。

表 7-5 リンク種別を表す定数

定数	データ型	意味
LINK_DCR	int 型	直接型リンク
LINK_RCR	int 型	参照型リンク
LINK_VTFIX	int 型	構成管理型リンク（構成管理モードは FIX モード）
LINK_VTFLOAT	int 型	構成管理型リンク（構成管理モードは FLOATING モード）
LINK_VT	int 型	構成管理型リンク（LINK_VTFIX   LINK_VTFLOAT）
LINK_REL	int 型	文書間リンク
LINK_ALL	int 型	すべてのリンク種別を組み合わせたリンク
LINK_NONE	int 型	リンクなし

### 7.1.6 Category : LOCK

ロック種別を表す定数です。

ロック種別を表す定数を表 7-6 に示します。

表 7-6 ロック種別を表す定数

定数	データ型	意味
LOCK_READ	int 型	read ロック
LOCK_WRITE	int 型	排他的な write ロック
LOCK_READFORUPDATE	int 型	read ロック (更新または削除ができません)
LOCK_NONE	int 型	ロックなし

### 7.1.7 Category : OBJTYPE

オブジェクト種別を表す定数です。

オブジェクト種別を表す定数を表 7-7 に示します。

表 7-7 オブジェクト種別を表す定数

定数	データ型	意味
OBJTYPE_ANY	int 型	任意のオブジェクト種別の論理和
OBJTYPE_NVTFOLDER	int 型	構成管理できないバージョンなしフォルダ
OBJTYPE_DOC	int 型	バージョンなし文書
OBJTYPE_VRFOLDER	int 型	バージョン付きフォルダ
OBJTYPE_VRDOC	int 型	バージョン付き文書
OBJTYPE_FOLDER	int 型	バージョンなしフォルダ
OBJTYPE_IP	int 型	独立データ
OBJTYPE_PUBLICACL	int 型	パブリック ACL
OBJTYPE_UNKNOWN	int 型	オブジェクト種別不明

### 7.1.8 Category : OPERATEMODE

コンテンツのパス操作モードを表す定数です。

コンテンツのパス操作モードを表す定数を表 7-8 に示します。

表 7-8 コンテンツのパス操作モードを表す定数

定数	データ型	意味
OPERATEMODE_NONE	int 型	コンテンツがないオブジェクトを作成します。
OPERATEMODE_USER_RELATIVE_CONTENT	int 型	コンテンツロケーションをコンテンツの相対パスで管理します。

### 7.1.9 Category : ORDER

バージョン情報の取得順序を表す定数です。

バージョン情報の取得順序を表す定数を表 7-9 に示します。

表 7-9 バージョン情報の取得順序を表す定数

定数	データ型	意味
ORDER_ASC	int 型	最も古いバージョンから順に取得します。
ORDER_DESC	int 型	最も新しいバージョンから順に取得します。
ORDER_NONE	int 型	順序を指定しません。

## 7.1.10 Category : PERM

パーミッションを表す定数です。

パーミッションを表す定数を表 7-10 に示します。

表 7-10 パーミッションを表す定数

定数	データ型	意味
PERM_NONE	int 型	アクセス権なし
PERM_PRIM_READ_PROPS	int 型	基本プロパティ参照権
PERM_PRIM_WRITE_PROPS	int 型	基本プロパティ更新権
PERM_PRIM_READ_CONTENTS	int 型	基本コンテンツ参照権
PERM_PRIM_WRITE_CONTENTS	int 型	基本コンテンツ更新権
PERM_PRIM_LINK	int 型	基本リンク権
PERM_PRIM_VERSION	int 型	基本バージョン管理権
PERM_PRIM_DELETE	int 型	基本削除権
PERM_CHANGE_PERM	int 型	アクセス制御情報変更権
PERM_CREATE	int 型	オブジェクト作成権
PERM_READ_PROPS	int 型	プロパティ参照権 次のパーミッションと等価 • PERM_PRIM_READ_PROPS
PERM_READ	int 型	参照権 次のパーミッションの論理和と等価 • PERM_PRIM_READ_PROPS • PERM_PRIM_READ_CONTENTS
PERM_WRITE_PROPS	int 型	プロパティ更新権 次のパーミッションの論理和と等価 • PERM_PRIM_READ_PROPS • PERM_PRIM_WRITE_PROPS
PERM_READ_WRITE	int 型	参照更新権 次のパーミッションの論理和と等価 • PERM_READ • PERM_PRIM_WRITE_PROPS • PERM_PRIM_WRITE_CONTENTS
PERM_DELETE	int 型	削除権 次のパーミッションの論理和と等価 • PERM_PRIM_READ_PROPS • PERM_PRIM_DELETE
PERM_LINK	int 型	リンク権 次のパーミッションの論理和と等価 • PERM_PRIM_READ_PROPS • PERM_PRIM_LINK
PERM_VERSION	int 型	バージョン権 次のパーミッションの論理和と等価 • PERM_READ_WRITE • PERM_PRIM_VERSION

定数	データ型	意味
PERM_FULL_CONTROL	int 型	フルコントロール 次のパーミッションの論理和と等価 • PERM_READ_WRITE • PERM_PRIM_LINK • PERM_PRIM_VERSION • PERM_PRIM_DELETE

### 7.1.11 Category : PRIV

文書空間に対するユーザの特権を表す定数です。

文書空間に対するユーザの特権を表す定数を表 7-11 に示します。

表 7-11 文書空間に対するユーザの特権を表す定数

定数	データ型	意味
PRIV_NONE	int 型	セキュリティ管理者ではありません。
PRIV_SECURITY_ADMINISTRATOR	int 型	セキュリティ管理者です。



## 7.1.12 Category : REFERENCETYPE

リファレンス種別を表す定数です。

リファレンス種別を表す定数を表 7-12 に示します。

表 7-12 リファレンス種別を表す定数

定数	データ型	意味
REFERENCETYPE_NONE	int 型	コンテンツはありません。
REFERENCETYPE_USER_RELATIVE_CONTENT	int 型	ユーザ管理領域に格納したコンテンツ ( 相対パス ) です。

### 7.1.13 Category : RELATIONEND

文書間リンク設定情報一覧の取得条件を表す定数です。

文書間リンク設定情報一覧の取得条件を表す定数を表 7-13 に示します。

表 7-13 文書間リンク設定情報一覧の取得条件を表す定数

定数	データ型	意味
RELATIONEND_HEAD	int 型	文書間リンクで関連付けているリンク先オブジェクトの一覧を取得します。
RELATIONEND_TAIL	int 型	文書間リンクで関連付けられているリンク元オブジェクトの一覧を取得します。
RELATIONEND_STATUS_EXIST	int 型	関連付けの対象オブジェクトが存在しているリンク設定情報の一覧を取得します。
RELATIONEND_STATUS_NOT_EXIST	int 型	すでに関連付けの対象オブジェクトが削除されていて、リンクだけが残っているリンク設定情報の一覧を取得します。 関連付けの対象オブジェクトが削除されているリンク設定情報はすべて取得されます。
RELATIONEND_STATUS_ALL	int 型	すべてのリンク設定情報の一覧を取得する場合に指定します。 関連付けの対象オブジェクトが削除されているリンク設定情報はすべて取得されます。

### 7.1.14 Category : RENDSTATUS

レンディションステータスを表す定数です。

レンディションステータスを表す定数を表 7-14 に示します。

表 7-14 レンディションステータスを表す定数

定数	データ型	意味
RENDSTATUS_NO_SUBREND	int 型	指定したサブレンディションに対応するマスターレンディションのコンテンツは登録済みですが、サブレンディションのコンテンツが未登録です。
RENDSTATUS_SUBREND_EXIST	int 型	指定したサブレンディションに対応するマスターレンディションおよびサブレンディションのコンテンツが登録されています。
RENDSTATUS_MASTERREND_UPDATE	int 型	指定したサブレンディションに対応するマスターレンディションが更新されていますが、サブレンディションが更新されていないため、マスターレンディションとサブレンディションの内容が不一致です。
RENDSTATUS_CONVERT_NOTREQUIRED	int 型	レンディション変換が不要な状態です。この状態の場合、DocumentBroker Rendering Option によるレンディション変換の対象になりません。
RENDSTATUS_CONVERT_REQUIRED	int 型	レンディション変換が必要な状態です。この状態の場合、DocumentBroker Rendering Option によるレンディション変換の対象になります。
RENDSTATUS_CONVERT_ERROR	int 型	DocumentBroker Rendering Option によるレンディション変換時にエラーが発生した状態です。この状態になると、DocumentBroker Rendering Option によるレンディション変換はできなくなります。
RENDSTATUS_MASTERD	int 型	指定したレンディションがマスターレンディションの場合、設定されます。

### 7.1.15 Category : SUBJECTTYPE

サブジェクト種別を表す定数です。

サブジェクト種別を表す定数を表 7-15 に示します。

表 7-15 サブジェクト種別を表す定数

定数	データ型	意味
SUBJECTTYPE_USR	int 型	ユーザサブジェクトです。サブジェクトに設定した内容はユーザ識別子です。
SUBJECTTYPE_GRP	int 型	グループサブジェクトです。サブジェクトに設定した内容はグループ識別子です。
SUBJECTTYPE_SYS	int 型	システムサブジェクトです。サブジェクトに設定した内容は、システムサブジェクトを表す定数です。

## 7.1.16 Category : SYSSUBJECT

システムサブジェクトを表す定数です。

システムサブジェクトを表す定数を表 7-16 に示します。

表 7-16 システムサブジェクトを表す定数

定数	データ型	意味
SYSSUBJECT_SELF	String 型	対象オブジェクトの所有者を表すサブジェクトです。 <ul style="list-style-type: none"> <li>対象オブジェクトがパブリック ACL の ACL の場合は、そのパブリック ACL をバインドしている文書やフォルダを表すオブジェクトの所有者を示します。</li> <li>対象オブジェクトが文書やフォルダを表すオブジェクトのローカル ACL およびセキュリティ ACL の場合は、そのオブジェクトの所有者を示します。</li> <li>対象オブジェクトがパブリック ACL のセキュリティ ACL の場合は、パブリック ACL の所有者を示します。</li> </ul>
SYSSUBJECT_EVERYONE	String 型	すべてのユーザを表すサブジェクトです。

### 7.1.17 Category : XMLPARSE

XML 文書の構文解析レベルを表す定数です。

XML 文書の構文解析レベルを表す定数を表 7-17 に示します。

表 7-17 XML 文書の構文解析レベルを表す定数

定数	データ型	意味
XMLPARSE_NO_EXTERNAL_ENTITIES	int 型	DTD の解析をしないで、外部エンティティは無視します。
XMLPARSE_WITH_EXTERNAL_ENTITIES	int 型	DTD を含めた構文解析をして外部エンティティを読み込みますが、検証は実行しません。
XMLPARSE_VALIDATE	int 型	DTD を使用して検証を実行します。

## 7.1.18 Category : Others

そのほかの内容を表す定数です。

そのほかの内容を表す定数を表 7-18 に示します。

表 7-18 そのほかの内容を表す定数

定数	データ型	意味
MAX_NUM	int 型	検索結果の取得件数を最大とします。
INITIAL_KEY	int 型	キャッシュ検索を実行しません。
INDEXPATH_SAME	String 型	全文検索インデクス作成用ファイルのパスを DbjUploadInfo#setIndexPath メソッドの filePath プロパティに指定したファイルのパスとします。
CURRENT_VERSION	String 型	アクセス対象のバージョンをカレントバージョンとします。
DEFAULT_DOCSPCID	String 型	文書空間識別子のデフォルト値 (673d2be0-d1fd-11d0-ab59-08002be29e1d) です。

## 7.2 DbjTraceDef クラス

---

DbjTraceDef クラスは、Java クラスライブラリのトレースクラスで使用する定数を定義したクラスです。

以降、DbjTraceDef クラスの定数定義内容について、カテゴリごとに説明します。

なお、DbjTraceDef クラスで定義されている定数のアクセス指定子は、すべて「public」です。



## 7.2.1 Category : TRACELEVEL

トレース出力時のトレースレベルを表す定数です。トレース出力時のトレースレベルを表す定数を表 7-19 に示します。

表 7-19 トレース出力時のトレースレベルを表す定数

定数	データ型	意味
ERROR	int 型	障害監視レベル
MANAGE	int 型	通常運用レベル
HINT	int 型	障害調査レベル
DEBUG	int 型	デバッグレベル

## 7.2.2 Category : TRACEOUTPUT

トレース出力時のトレース出力先を表す定数です。トレース出力時のトレース出力先を表す定数を表 7-20 に示します。

表 7-20 トレース出力時のトレース出力先を表す定数

定数	データ型	意味
PROMPT	int 型	コマンドプロンプト（標準出力・標準エラー出力）に出力します。
TRACE	int 型	アプリケーショントレースファイルに出力します。
ERRORLOG	int 型	アプリケーションエラーログファイルに出力します。
SYSTEM	int 型	イベントログファイルまたは syslog ファイルに出力します。

# 8

## ライブラリ情報取得クラス詳細

この章では、ライブラリ情報取得クラスの DbjLibInfo クラス、およびメソッドについて説明します。

---

### 8.1 DbjLibInfo クラス

## 8.1 DbjLibInfo クラス

---

DbjLibInfo クラスは、DocumentBroker Development Kit および DocumentBroker Runtime のバージョン情報を扱うクラスです。このクラスはインスタンスを生成できません（コンストラクタは、private 属性です）。

以降、DbjLibInfo クラスのメソッドについて説明します。

## 8.1.1 getVersion (バージョンの取得)

### (1) 機能

DocumentBroker Development Kit および DocumentBroker Runtime のバージョン番号およびリビジョン番号を取得します。バージョン番号およびリビジョン番号は、次に示す計算式によって値を求めた結果の整数値で取得します。なお、バージョン番号およびリビジョン番号が XX-YY の場合、XX をバージョン番号、YY をリビジョン番号とします。ただし、リビジョン番号は 2 けたまでです。

計算式

$$(A \times 100) + B$$

(凡例)

A : バージョン番号

B : リビジョン番号

例えば、バージョン番号 - リビジョン番号が 02-10 の場合、バージョン番号は 2、リビジョン番号は 10 になります。したがって、このメソッドの実行によって 210 が返却されます。

### (2) 形式

```
static int getVersion()
```

### (3) 引数

なし

### (4) 戻り値

バージョン番号およびリビジョン番号

### (5) 例外

なし



# 9

## トレースクラス詳細

この章では、トレースクラスのクラスとそのメソッドについて説明します。

---

### 9.1 DbjTrace クラス

## 9.1 DbjTrace クラス

---

DbjTrace クラスは、トレース情報を扱うクラスです。以降、DbjTrace クラスのメソッドについて説明します。



## 9.1.1 arg (パラメタ情報の出力)

### (1) 機能

メソッドの入り口での入力パラメタ, または外部 API に渡すパラメタ情報を出力します。引数で指定したパラメタ名およびパラメタに設定した値を出力します。このメソッドは, メソッドの入り口または外部 API の実行前に実行してください。

### (2) 形式

int 型のパラメタ情報を出力する場合は, 形式 1 ~ 形式 3 の形式で指定してください。String 型のパラメタ情報を出力する場合は, 形式 4 ~ 形式 6 の形式で指定してください。

#### (a) 形式 1

```
void arg(
    String      name,
    int         value
    DbjTraceDef level,
    int         output
)
```

#### (b) 形式 2

```
void arg(
    String      name,
    int         value
    DbjTraceDef level
)
```

形式 2 の場合, トレース情報の出力先はアプリケーショントレースファイルになります。

#### (c) 形式 3

```
void arg(
    String      name,
    int         value
)
```

形式 3 の場合, トレース情報の出力先はアプリケーショントレースファイルになります。また, トレースレベルは DbjTraceDef.DEBUG になります。

#### (d) 形式 4

```
void arg(
    String      name,
    String      value
    DbjTraceDef level,
    int         output
)
```

#### (e) 形式 5

```
void arg(
    String      name,
    String      value
    DbjTraceDef level
)
```

形式 5 の場合, トレース情報の出力先はアプリケーショントレースファイルになります。

#### (f) 形式 6

```
void arg(
    String      name,
    String      value
)
```

)

形式 6 の場合、トレース情報の出力先はアプリケーショントレースファイルになります。また、トレースレベルは `DbjTraceDef.DEBUG` になります。

### (3) 引数

name ( 入力 )

メソッド入力のパラメタ名、または外部 API に渡すパラメタ名を指定します。

value ( 入力 )

パラメタに設定した値を指定します。

level ( 入力 )

トレースレベルを指定します。次に示すトレースレベルのうち、どれかを指定してください。

- `DbjTraceDef.ERROR`  
障害監視レベル
- `DbjTraceDef.MANAGE`  
通常運用レベル
- `DbjTraceDef.HINT`  
障害調査レベル
- `DbjTraceDef.DEBUG`  
デバッグレベル

output ( 入力 )

トレース情報の出力先を指定します。次に示す出力先を指定できます。

- `DbjTraceDef.PROMPT`  
コマンドプロンプト ( 標準出力・標準エラー出力 ) に出力します。
- `DbjTraceDef.TRACE`  
アプリケーショントレースファイルに出力します。
- `DbjTraceDef.ERRORLOG`  
アプリケーションエラーログファイルに出力します。

前記以外の出力先を指定した場合、トレース情報は出力されません ( 実行したメソッドは正常終了します )。

複数の出力先を指定する場合は、OR 演算子を用いて定数を連結してください。例えば、

`DbjTraceDef.TRACE` と `DbjTraceDef.ERRORLOG` に同時出力する場合は、`DbjTraceDef.TRACE | DbjTraceDef.ERRORLOG` と指定してください。

### (4) 戻り値

なし

### (5) 例外

なし

## 9.1.2 call (外部 API の呼び出し情報の出力)

### (1) 機能

外部 API の呼び出し時の情報を出力します。引数で指定した呼び出し元のクラス名およびメソッド名を出力します。このメソッドは、外部 API の呼び出し時に実行してください。

### (2) 形式

クラス名およびメソッド名を出力する場合は、形式 1 ~ 形式 3 の形式で指定してください。メソッド名だけを出力する場合は、形式 4 ~ 形式 6 の形式で指定してください。

#### (a) 形式 1

```
void call(
    String      className,
    String      methodName,
    DbjTraceDef level,
    int         output
)
```

#### (b) 形式 2

```
void call(
    String      className,
    String      methodName,
    DbjTraceDef level
)
```

形式 2 の場合、トレース情報の出力先はアプリケーショントレースファイルになります。

#### (c) 形式 3

```
void call(
    String      className,
    String      methodName
)
```

形式 3 の場合、トレース情報の出力先はアプリケーショントレースファイルになります。また、トレースレベルは DbjTraceDef.DEBUG になります。

#### (d) 形式 4

```
void call(
    String      methodName,
    DbjTraceDef level,
    int         output
)
```

#### (e) 形式 5

```
void call(
    String      methodName,
    DbjTraceDef level
)
```

形式 5 の場合、トレース情報の出力先はアプリケーショントレースファイルになります。

#### (f) 形式 6

```
void call(
    String      methodName
)
```

形式 6 の場合、トレース情報の出力先はアプリケーショントレースファイルになります。また、トレースレベルは DbjTraceDef.DEBUG になります。

### (3) 引数

#### className (入力)

呼び出しクラスの名称を 256 バイト以内で指定します。256 バイトを超えた名称を指定した場合は、256 バイトまでの文字列が設定されます。

#### methodName (入力)

呼び出しメソッドの名称を 256 バイト以内で指定します。256 バイトを超えた名称を指定した場合は、256 バイトまでの文字列が設定されます。

#### level (入力)

トレースレベルを指定します。次に示すトレースレベルのうち、どれかを指定してください。

- DbjTraceDef.ERROR  
障害監視レベル
- DbjTraceDef.MANAGE  
通常運用レベル
- DbjTraceDef.HINT  
障害調査レベル
- DbjTraceDef.DEBUG  
デバッグレベル

#### output (入力)

トレース情報の出力先を指定します。次に示す出力先を指定できます。

- DbjTraceDef.TRACE  
アプリケーショントレースファイルに出力します。

前記以外の出力先を指定した場合、トレース情報は出力されません（実行したメソッドは正常終了します）。

### (4) 戻り値

なし

### (5) 例外

なし

### 9.1.3 DbjTrace (コンストラクタ)

#### (1) 機能

DbjTrace オブジェクトを作成します。

#### (2) 形式

```
DbjTrace(  
    String      upName,  
    String      className,  
    String      methodName  
)
```

#### (3) 引数

##### upName (入力)

アプリケーション名を指定します。

ここで指定したアプリケーション名を `init` メソッドの引数 `upName` に指定してください。異なるアプリケーション名を指定した場合、トレース情報は出力されません。また、アプリケーション名に `null` を指定した場合もトレース情報は出力されません。

##### className (入力)

クラス名を 256 バイト以内で指定します。256 バイトを超えた名称を指定した場合は、256 バイトまでの文字列が設定されます。

`null` を指定した場合、クラス名は設定されません。この場合、メソッド名だけが出力されます。

##### methodName (入力)

メソッド名を 256 バイト以内で指定します。256 バイトを超えた名称を指定した場合は、256 バイトまでの文字列が設定されます。

`null` を指定した場合、`enter` または `exit` メソッドを実行しても、トレース情報は出力されません。

#### (4) 戻り値

なし

#### (5) 例外

なし

### 9.1.4 enter (メソッドの入り口情報の出力)

#### (1) 機能

メソッドの入り口情報を出力します。DbjTrace メソッドで指定したクラス名およびメソッド名を出力します。このメソッドは、メソッドの入り口で実行してください。

#### (2) 形式

##### (a) 形式 1

```
void enter(
    DbjTraceDef    level,
    int            output
)
```

##### (b) 形式 2

```
void enter(
    DbjTraceDef    level
)
```

形式 2 の場合、トレース情報の出力先はアプリケーショントレースファイルになります。

##### (c) 形式 3

```
void enter()
```

形式 3 の場合、トレース情報の出力先はアプリケーショントレースファイルになります。また、トレースレベルは DbjTraceDef.DEBUG になります。

#### (3) 引数

##### level (入力)

トレースレベルを指定します。次に示すトレースレベルのうち、どれかを指定してください。

- DbjTraceDef.ERROR  
障害監視レベル
- DbjTraceDef.MANAGE  
通常運用レベル
- DbjTraceDef.HINT  
障害調査レベル
- DbjTraceDef.DEBUG  
デバッグレベル

##### output (入力)

トレース情報の出力先を指定します。次に示す出力先を指定できます。

- DbjTraceDef.PROMPT  
コマンドプロンプト (標準出力・標準エラー出力) に出力します。
- DbjTraceDef.TRACE  
アプリケーショントレースファイルに出力します。

前記以外の出力先を指定した場合、トレース情報は出力されません (実行したメソッドは正常終了します)。

複数の出力先を指定する場合は、OR 演算子を用いて定数を連結してください。例えば、DbjTraceDef.PROMPT と DbjTraceDef.TRACE に同時出力する場合は、DbjTraceDef.PROMPT | DbjTraceDef.TRACE と指定してください。

(4) 戻り値

なし

(5) 例外

なし

## 9.1.5 error (エラー情報の出力)

### (1) 機能

ユーザアプリケーションプログラムにエラーが発生したときのエラー情報を出力します。引数で指定したメッセージまたは例外クラスを出力します。このメソッドは、ユーザアプリケーションプログラムのエラー発生時に実行してください。

### (2) 形式

エラー情報をメッセージにして出力する場合は、形式1～形式3の形式で指定してください。通常の例外の情報を出力する場合は、形式4～形式6の形式で指定してください。Java クラスライブラリの例外の情報を出力する場合は、形式7～形式9の形式で指定してください。

#### (a) 形式1

```
void error(
    String      message,
    DbjTraceDef level,
    int         output
)
```

#### (b) 形式2

```
void error(
    String      message,
    DbjTraceDef level
)
```

形式2の場合、トレース情報の出力先はアプリケーショントレースファイルおよびアプリケーションエラーログファイルになります。

#### (c) 形式3

```
void error(
    String      message
)
```

形式3の場合、トレース情報の出力先はアプリケーショントレースファイルおよびアプリケーションエラーログファイルになります。また、トレースレベルは DbjTraceDef.ERROR になります。

#### (d) 形式4

```
void error(
    Exception   ex,
    DbjTraceDef level,
    int         output
)
```

#### (e) 形式5

```
void error(
    Exception   ex,
    DbjTraceDef level
)
```

形式5の場合、トレース情報の出力先はアプリケーショントレースファイルおよびアプリケーションエラーログファイルになります。

#### (f) 形式6

```
void error(
    Exception   ex
)
```



形式 6 の場合、トレース情報の出力先はアプリケーショントレースファイルおよびアプリケーションエラーログファイルになります。また、トレースレベルは DbjTraceDef.ERROR になります。

## (g) 形式 7

```
void error(
    DbjException    ex,
    DbjTraceDef     level,
    int             output
)
```

## (h) 形式 8

```
void error(
    DbjException    ex,
    DbjTraceDef     level
)
```

形式 8 の場合、トレース情報の出力先はアプリケーショントレースファイルおよびアプリケーションエラーログファイルになります。

## (i) 形式 9

```
void error(
    DbjException    ex
)
```

形式 9 の場合、トレース情報の出力先はアプリケーショントレースファイルおよびアプリケーションエラーログファイルになります。また、トレースレベルは DbjTraceDef.ERROR になります。

## (3) 引数

## message ( 入力 )

出力するメッセージを指定します。

## ex ( 入力 )

発生した例外クラスを指定します。

Exception クラスおよびそのサブクラスを指定した場合は、`ex.getMessage()` および `ex.printStackTrace()` の内容が出力されます。

## level ( 入力 )

トレースレベルを指定します。次に示すトレースレベルのうち、どれかを指定してください。

- DbjTraceDef.ERROR  
障害監視レベル
- DbjTraceDef.MANAGE  
通常運用レベル
- DbjTraceDef.HINT  
障害調査レベル
- DbjTraceDef.DEBUG  
デバッグレベル

## output ( 入力 )

トレース情報の出力先を指定します。次に示す出力先を指定できます。

- DbjTraceDef.PROMPT  
コマンドプロンプト ( 標準出力・標準エラー出力 ) に出力します。
- DbjTraceDef.TRACE  
アプリケーショントレースファイルに出力します。
- DbjTraceDef.ERRORLOG

## 9. トレースクラス詳細

アプリケーションエラーログファイルに出力します。

前記以外の出力先を指定した場合、トレース情報は出力されません（実行したメソッドは正常終了します）。

複数の出力先を指定する場合は、OR 演算子を用いて定数を連結してください。例えば、DbjTraceDef.TRACE と DbjTraceDef.ERRORLOG に同時出力する場合は、DbjTraceDef.TRACE | DbjTraceDef.ERRORLOG と指定してください。

### (4) 戻り値

なし

### (5) 例外

なし

## 9.1.6 exit (メソッドの出口情報の出力)

### (1) 機能

メソッドの出口情報を出力します。DbjTrace メソッドで指定したクラス名およびメソッド名を出力します。このメソッドは、メソッドの出口で実行してください。

クラス名およびメソッド名については、DbjTrace メソッドで指定した値が設定されます。トレースレベルおよび出力先については、enter メソッドで指定した値が設定されます。したがって、enter メソッドを実行する前に exit メソッドを実行した場合、トレース情報は出力されません。

### (2) 形式

```
void exit()
```

### (3) 引数

なし

### (4) 戻り値

なし

### (5) 例外

なし

## 9.1.7 hint ( 下位のメソッドのエラー情報の出力 )

### (1) 機能

下位のメソッドでエラーが発生したときのエラー情報を出力します。引数で指定したメッセージを出力します。このメソッドは、下位のメソッドのエラー発生時に実行してください。

### (2) 形式

#### (a) 形式 1

```
void hint(
    String      message,
    DbjTraceDef level,
    int         output
)
```

#### (b) 形式 2

```
void hint(
    String      message,
    DbjTraceDef level
)
```

形式 2 の場合、トレース情報の出力先はアプリケーショントレースファイルになります。

#### (c) 形式 3

```
void hint(
    String      message
)
```

形式 3 の場合、トレース情報の出力先はアプリケーショントレースファイルになります。また、トレースレベルは DbjTraceDef.HINT になります。

### (3) 引数

message ( 入力 )

出力するメッセージを指定します。

level ( 入力 )

トレースレベルを指定します。次に示すトレースレベルのうち、どれかを指定してください。

- DbjTraceDef.ERROR  
障害監視レベル
- DbjTraceDef.MANAGE  
通常運用レベル
- DbjTraceDef.HINT  
障害調査レベル
- DbjTraceDef.DEBUG  
デバッグレベル

output ( 入力 )

トレース情報の出力先を指定します。次に示す出力先を指定できます。

- DbjTraceDef.TRACE  
アプリケーショントレースファイルに出力します。

前記以外の出力先を指定した場合、トレース情報は出力されません ( 実行したメソッドは正常終了します )。

(4) 戻り値

なし

(5) 例外

なし

## 9.1.8 init ( トレースクラスの初期化 )

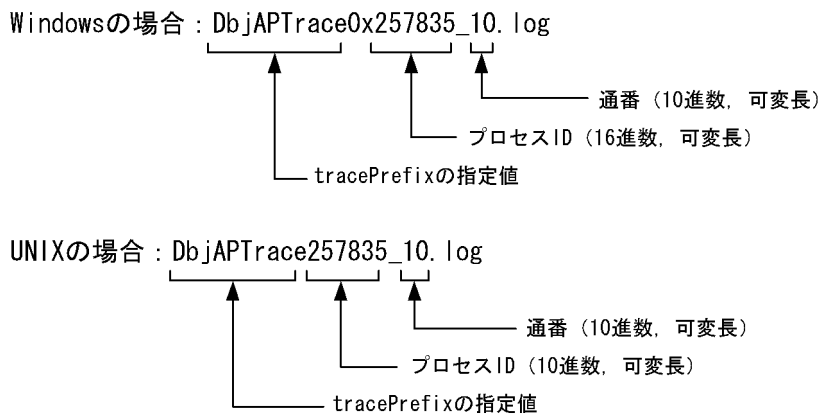
### (1) 機能

トレースクラスを初期化します。

このメソッドで指定する引数 `tracePrefix` を基に、アプリケーショントレースファイル名およびアプリケーションエラーログファイル名が次のように決まります。

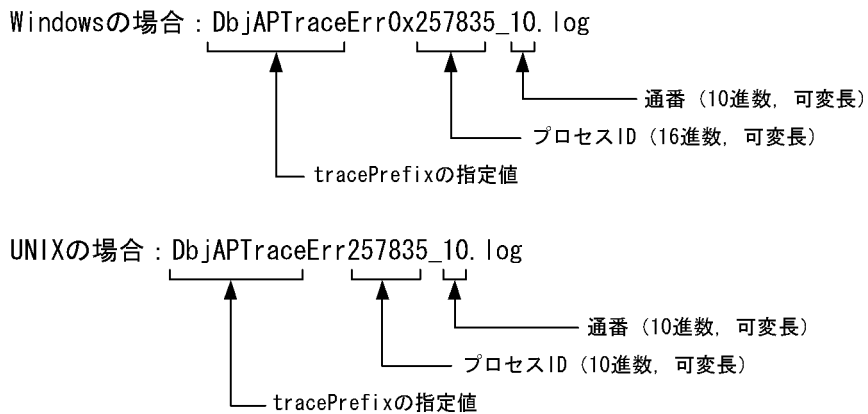
アプリケーショントレースファイルの場合

アプリケーショントレースファイル名の例を次に示します。



アプリケーションエラーログファイルの場合

アプリケーションエラーログファイル名の例を次に示します。



#### ! 注意事項

- DbjTrace クラスを使用してトレース情報を出力するには、最初に `init` メソッドを実行してトレースクラスを初期化してください。init メソッドの実行前や、init メソッドで例外が送出された場合に、init 以外のメソッドを実行すると、そのメソッドは処理を行わないで正常終了します。
- init メソッドが正常終了したあとに、同じ `upName` および `tracePrefix` を指定した `init` メソッドを実行できません。同じ `upName` および `tracePrefix` を指定した `init` メソッドを実行した場合、処理を行わないで正常終了します。

## (2) 形式

```
void init(
    String      upName
    String      tracePrefix
)
```

## (3) 引数

### upName (入力)

アプリケーション名を 16 バイト以内で指定します。16 バイトを超えた名称を指定した場合は、16 バイトまでの文字列が設定されます。null を指定した場合は、「DbjAP」が設定されます。

### tracePrefix (入力)

アプリケーショントレースファイル名およびアプリケーションエラーログファイル名のプリフィクス (先頭部分) を 128 バイト以内で指定します。128 バイトを超えた名称を指定した場合は、128 バイトまでの文字列が設定されます。null を指定した場合は、「DbjAPTrace」が設定されます。

## (4) 戻り値

なし

## (5) 例外

### IllegalArgumentException

指定した引数に誤りがある場合、この例外が送出されます。例えば、次に示す場合にこの例外が送出されます。

- 正常終了した init メソッドと同じ upName または tracePrefix を指定し、再度 init メソッドを実行した場合

### IOException

ファイル操作に失敗した場合、この例外が送出されます。例えば、次に示す場合にこの例外が送出されます。

- アプリケーショントレースファイルまたはアプリケーションエラーログファイルの格納先ディレクトリがない場合
- アプリケーショントレースファイルまたはアプリケーションエラーログファイルを作成する権限がない場合
- アプリケーショントレースファイルまたはアプリケーションエラーログファイルのパス長 (格納先ディレクトリとファイルのパス長の合計) が最大パス長を超えている場合

## 9.1.9 msg (メッセージの出力)

### (1) 機能

メッセージを出力します。引数で指定したメッセージ ID およびメッセージを出力します。このメソッドは、メッセージの出力が必要なときに実行してください。

出力先にイベントログファイルを指定した場合 (Windows 限定)

引数 messageID の終端文字の指定によって、イベントログの種類が決まります。引数 messageID の終端文字の指定とイベントログの種類の対応を表 9-1 に示します。

表 9-1 引数 messageID の終端文字の指定とイベントログの種類の対応

引数 messageID の終端文字の指定	イベントログの種類
-E	エラー
-W	警告
-I	情報
上記以外	

### (2) 形式

メッセージ ID およびメッセージを出力する場合は、形式 1 ~ 形式 3 の形式で指定してください。メッセージだけを出力する場合は、形式 4 ~ 形式 6 の形式で指定してください。

#### (a) 形式 1

```
void msg(
    String      messageID,
    String      message,
    DbjTraceDef level,
    int         output
)
```

#### (b) 形式 2

```
void msg(
    String      messageID,
    String      message,
    DbjTraceDef level
)
```

形式 2 の場合、トレース情報の出力先はアプリケーショントレースファイルになります。

#### (c) 形式 3

```
void msg(
    String      messageID,
    String      message
)
```

形式 3 の場合、トレース情報の出力先はアプリケーショントレースファイルになります。また、トレースレベルは DbjTraceDef.ERROR になります。

#### (d) 形式 4

```
void msg(
    String      message,
    DbjTraceDef level,
    int         output
)
```



## (e) 形式 5

```
void msg(
    String      message,
    DbjTraceDef level
)
```

形式 5 の場合、トレース情報の出力先はアプリケーショントレースファイルになります。

## (f) 形式 6

```
void msg(
    String      message
)
```

形式 6 の場合、トレース情報の出力先はアプリケーショントレースファイルになります。また、トレースレベルは DbjTraceDef.ERROR になります。

## (3) 引数

## messageID ( 入力 )

出力するメッセージ ID を 11 バイト以内で指定します。11 バイトを超えた場合は、11 バイトまでの文字列が設定されます。null を指定した場合、メッセージ ID は設定されません。

## message ( 入力 )

出力するメッセージを指定します。

## level ( 入力 )

トレースレベルを指定します。次に示すトレースレベルのうち、どれかを指定してください。

- DbjTraceDef.ERROR  
障害監視レベル
- DbjTraceDef.MANAGE  
通常運用レベル
- DbjTraceDef.HINT  
障害調査レベル
- DbjTraceDef.DEBUG  
デバッグレベル

## output ( 入力 )

トレース情報の出力先を指定します。次に示す出力先を指定できます。

- DbjTraceDef.PROMPT  
コマンドプロンプト ( 標準出力・標準エラー出力 ) に出力します。
- DbjTraceDef.TRACE  
アプリケーショントレースファイルに出力します。
- DbjTraceDef.ERRORLOG  
アプリケーションエラーログファイルに出力します。
- DbjTraceDef.SYSTEM  
イベントログファイルまたは syslog ファイルに出力します。

前記以外の出力先を指定した場合、トレース情報は出力されません ( 実行したメソッドは正常終了します )。

複数の出力先を指定する場合は、OR 演算子を用いて定数を連結してください。例えば、

DbjTraceDef.TRACE と DbjTraceDef.ERRORLOG に同時出力する場合は、DbjTraceDef.TRACE | DbjTraceDef.ERRORLOG と指定してください。

9. トレースクラス詳細

(4) 戻り値

なし

(5) 例外

なし

## 9.1.10 returned (外部 API からのリターン情報の出力)

### (1) 機能

外部 API から制御が戻ってきたときの情報を出力します。call メソッドの引数で指定したクラス名およびメソッド名を出力します。このメソッドは、外部 API から制御が戻ってきたときに実行してください。

クラス名、メソッド名、トレースレベルおよび出力先については、call メソッドで指定した値が設定されます。したがって、call メソッドを実行する前に returned メソッドを実行した場合、トレース情報は出力されません。

### (2) 形式

```
void returned()
```

### (3) 引数

なし

### (4) 戻り値

なし

### (5) 例外

なし



# 10 メッセージの一覧

この章では、Java クラスライブラリが出力するメッセージについて説明します。

---

10.1 メッセージの形式と見方

---

10.2 メッセージの詳細

---

## 10.1 メッセージの形式と見方

---

この節では、Java クラスライブラリが出力するメッセージの形式と見方について説明します。

### (1) メッセージの形式

Java クラスライブラリで出力されるメッセージは、クライアント共有トレースファイルに出力されます。詳細はマニュアル「DocumentBroker Version 3 クラスライブラリ Java 解説」の「8.2 メッセージ情報」を参照してください。

メッセージの形式を次に示します。

KMBJnnnnn-T：メッセージテキスト

KMBJ

「KMBJ」は Java クラスライブラリのメッセージを示します。

nnnnn

メッセージの通し番号です。

T

タイプコードです。次のように対応しています。

I (Information)

Java クラスライブラリの動作を通知するメッセージです。メッセージの発行後も処理を続行します。

W (Warning)

警告レベルのトラブルが発生したことを通知するメッセージです。メッセージの発行後も処理を続行します。

E (Error)

エラーレベルのトラブルが発生したことを通知するメッセージです。このメッセージが発行されたときは、メソッドの実行を中断します。ただし、致命的なエラーでなければ、処理を再開できます。

### (2) メッセージテキスト中の可変値について

メッセージテキスト中の %s, %n (n は挿入句の順番) はそれぞれテキスト, 数値型の可変値です。

### (3) メッセージの見方

このマニュアルでは、メッセージを次に示す形式で説明しています。

#### **KMBJnnnnn-T**

---

日本語のメッセージテキスト

英文のメッセージテキスト

要因

メッセージが通知された要因

対処

メッセージに対するユーザの対処方法

#### (4) 注意

##### (a) メッセージテキストについて

実際に出力されるメッセージテキストは、日本語または英文のどちらか一方です。デフォルトでは、日本語になっています。

メッセージテキストを、日本語から英文または英文から日本語に変更する方法については、マニュアル「DocumentBroker Version 3 クラスライブラリ Java 解説」の「7.4.2 環境変数の設定」を参照してください。

##### (b) 対処に「システム管理者に連絡してください」とある場合について

「システム管理者に連絡してください」とは、システム管理者が対処するか、システム管理者が弊社社員または弊社担当営業に連絡することを示します。

エラーコードなどは、システム管理者、弊社社員または弊社担当営業が使う情報です。

システム管理者、弊社社員または弊社担当営業に連絡される際には、極力、次に示す情報もお伝えくださいます様お願いします。

- いつエラーが発生したのか
- エラー発生直前にどのような操作をしたのか
- エラー発生時に画面などがどのように動作したのか
- エラー発生時に出力されたメッセージの内容

できるだけ正確な内容でお伝えください。特にメッセージ番号と可変値は必要な情報です。

DocumentBroker サーバを運用中に発生したエラーメッセージの詳細については、マニュアル「DocumentBroker Version 3 メッセージ」を参照してください。

## 10.2 メッセージの詳細

---

この節では、Java クラスライブラリでエラー発生時に出力される各エラーメッセージの詳細について説明します。

### KMBJ00001-E

---

指定された操作は、指定されたオブジェクトに対しては、サポートされていません。操作とオブジェクトの型を見直してください。

The specified operation is not supported by the specified object, Try out the operation with the appropriate object type.

**要因**

指定したオブジェクトに対する操作はサポートされていません。

**対処**

オブジェクトの型を見直して、再実行してください。

### KMBJ00002-E

---

内部エラーが発生しました。システム管理者に連絡してください。

An Internal Error has occurred, contact the System Administrator.

**要因**

内部エラーが発生しました。

**対処**

システム管理者に連絡してください。

### KMBJ00003-E

---

メモリが不足しているため、指定された操作を実行できません。システム管理者に連絡してください。

The specified operation cannot be carried out since the system is running out of memory, Contact System Administrator.

**要因**

メモリが不足しているため、指定された操作を実行できません。

**対処**

メモリを増設するか、メモリを大量に消費しているアプリケーションを終了して、利用できるメモリを増やしてから、処理を再実行してください。

### KMBJ00004-E

---

ログインされていません。

The User has not legged in, login and try again.

**要因**

ログインされていません。

**対処**



ログインしてから，操作を実行してください。

## KMBJ00005-E

---

セッションが切断されています。接続してから，再実行してください。

The Session got disconnected, try again after reconnecting.

### 要因

セッションが切断されています。

### 対処

再接続して，再実行してください。

## KMBJ00006-E

---

データベースでタイムアウトが発生しました。SQLCODE: %4, SQLSTATE: %5 再実行してください。

A Data Base time out has occurred, SQL CODE: %4, SQL STATE: %5. Perform the operation again.

### 要因

データベースでタイムアウトが発生しました。SQLCODE: %4, SQLSTATE: %5

### 対処

再実行してください。

## KMBJ00007-E

---

データベースでデッドロックが発生しました。SQLCODE: %4, SQLSTATE: %5 システム管理者に連絡してください。

A dead lock has occurred SQLCODE: %4, SQLSTATE: %5. Contact the System Administrator

### 要因

データベースでデッドロックが発生しました。SQLCODE: %4, SQLSTATE: %5

### 対処

システム管理者に連絡してください。

## KMBJ00008-E

---

セッションが接続されていません。接続してから，実行してください。

The Session is not connected. Try after Connecting.

### 要因

セッションが接続されていません。

### 対処

接続してから，再実行してください。

## KMBJ00009-E

---

CORBA エラーが発生しました。CORBA%4 システム管理者に連絡してください。

A CORBA Error has been occurred, CORBA%4. Contact System administrator.

要因

CORBA エラーが発生しました。

対処

ユーザアプリケーションプログラムが TPBroker に接続できません。ユーザアプリケーションプログラムを終了させてください。

## KMBJ00010-E

---

指定されたオブジェクトは不正です。適切なオブジェクトを指定して実行してください。

The specified object is invalid, Try out the operation with the appropriate object type.

要因

指定されたオブジェクトは、サポートされていない型のオブジェクトです。

対処

適切な型のオブジェクトを指定して、再実行してください。

## KMBJ00011-E

---

サーバに接続できません。システム管理者に連絡してください。

Cannot connect to the Server. Contact System Administrator.

要因

サーバに接続しているユーザ数が、上限を超えています。

対処

しばらくたってから接続するか、文書空間へのユーザ接続数の上限値を変更してください。

## KMBJ00012-E

---

指定された文書空間識別子は、存在しません。有効な文書空間識別子を指定してください。

Specified Document space Id does not exist. Specify a valid value.

要因

ログイン時に指定された文書空間識別子は、存在しません。

対処

有効な文書空間識別子を指定してください。

## KMBJ00013-E

---

文書空間識別子が指定されていません。文書空間識別子を指定してください。

Specified Document space Id is null. Specify a valid value.

要因

ログイン時に文書空間識別子が指定されていません。

対処

有効な文書空間識別子を指定してください。

## KMBJ00014-E

---

ユーザ名が指定されていません。ユーザ名を指定してください。

Specified User Name is null. Specify a valid value.

要因

ログイン時にユーザ名が指定されていません。

対処

有効なユーザ名を指定してください。

## KMBJ00015-E

---

パスワードが指定されていません。パスワードを指定してください。

Specified Password is null. Specify a valid value.

要因

ログイン時にパスワードが指定されていません。

対処

正しいパスワードを指定してください。

## KMBJ00016-E

---

ユーザ : %4 のログインが失敗しました。ログインの引数が不正です。正しい値を指定してください。

Login failed for the user: %4. Invalid Login parameters. Enter correct values.

要因

指定ユーザは認証されませんでした。

対処

有効なユーザ名とパスワードを指定してください。

## KMBJ00017-E

---

アクセス制御機能が未サポートです。システム管理者に連絡してください。

Access control function is not supported. Contact System Administrator.

要因

アクセス制御機能がサポートされていません。

対処

この操作は、実行できません。

## KMBJ00018-E

---

メソッドの実行順序が不正です。正しい順序でメソッドを使用してください。

The sequence in which one method follows another is invalid. Use methods in valid sequence.

要因

メソッドの実行順序が不正です。

- トランザクションが開始されていない状態でトランザクションを有効または無効にしようとし

た。

- トランザクション処理中に、ほかのトランザクションを開始しようとしてしました。

対処

この操作は、実行できません。

## KMBJ00019-E

---

指定されたリレーションのタイプが不正です。正しいリレーションのタイプを指定してください。

The Relation Type specified is invalid. Specify proper Relation type.

要因

DbjObj#getRelList メソッドで指定されたリレーションタイプが不正です。

対処

正しいリレーションタイプを指定してください。

## KMBJ00020-E

---

指定されたリレーションステータスが不正です。正しいリレーションステータスを指定してください。

The Relation status specified is invalid. Specify proper Relation state.

要因

DbjObj#getRelList メソッドで指定されたリレーションステータスが不正です。

対処

正しいリレーションステータスを指定してください。

## KMBJ00021-E

---

指定された操作を実行する権限がありません。システム管理者に連絡してください。

No access right to carry out the specified operation. Contact System Administrator.

要因

指定された操作を実行する権限がありません。

対処

アクセス制御情報のパーミッションを変更して、再実行してください。

## KMBJ00022-E

---

この操作で使用するオブジェクトは、既に削除されています。

The Object used in this operation is already deleted. Specify a valid object.

要因

この操作で使用するオブジェクトは、すでに削除されています。

対処

存在するオブジェクトを指定して、再実行してください。

## KMBJ00023-E

---

ロックの種類が不正です。正しい種類のロックを指定してください。

Invalid Lock Type. Specify proper lock type.

要因

指定されたロックの種類が不正です。

対処

正しい種類のロックを指定してください。

## KMBJ00024-E

---

プロパティ識別子が不正です。正しいプロパティ識別子を指定してください。

Invalid Property Identifier. Specify proper Identifier value.

要因

不正なプロパティ識別子が含まれています。

対処

プロパティ識別子を見直して、再実行してください。

## KMBJ00025-E

---

要求された要素が見つかりません。有効なオブジェクトを指定してください。

The Requested Elements not found. Specify valid objects.

要因

要求された要素が見つかりません。

対処

存在するオブジェクトを指定してください。

## KMBJ00026-E

---

セッションの状態が不正です。システム管理者に連絡してください。

Invalid session status. Contact System Administrator.

要因

セッションの状態が不正です。

対処

一つのセッションに対して1度に一つの処理要求しか出さないように見直してください。

## KMBJ00027-E

---

OIID が設定されていません。正しい OIID を設定してください。

OIID is not Set. Set proper value for OIID.

要因

OIID が設定されていません。

対処

正しい OIID を設定してください。

## KMBJ00028-E

---

予期しないエラーが発生しました。システム管理者に連絡してください。

An Unexpected Error has occurred. Contact the System Administrator.

要因

予期しないエラーが発生しました。

対処

システム管理者に連絡してください。

## KMBJ00029-E

---

引数 %4 が不正です。有効な引数を指定してください。

Argument No %4 is not a valid one. Specify a valid argument.

要因

引数 %4 が不正です。

対処

有効な引数を指定してください。

## KMBJ00030-E

---

データベースでエラーが発生しました。システム管理者に連絡してください。SQLCODE: %4, SQLSTATE: %5.

Data Base Error has occurred. Contact the System Administrator. SQLCODE: %4, SQLSTATE: %5.

要因

データベースでエラーが発生したため操作を実行できません。SQLCODE: %4, SQLSTATE: %5

対処

システム管理者に連絡してください。

## KMBJ00031-E

---

クラス識別子が不正です。有効なクラス識別子を指定してください。

Invalid Class Id. Specify valid value for Class Id.

要因

指定されたクラス識別子が不正です。

対処

有効なクラス識別子を指定してください。

## KMBJ00032-E

---

OIID が不正です。有効な OIID を指定してください。

Invalid OIID. Specify valid value for OIID.

要因

指定された OIID が不正です。

対処

有効な OIID を指定してください。

## KMBJ00033-E

---

不正なパラメタでサーバを操作しました。

Invoked server operation with invalid parameter.

要因

不正なパラメタでサーバを操作しました。

対処

内部エラーが発生しました。システム管理者に連絡してください。

## KMBJ00034-E

---

このセッションでは、この操作はサポートされていません。システム管理者に連絡してください。

This operation is not supported in this session. Contact System Administrator.

要因

このセッションでは、この操作はサポートされていません。

対処

指定された操作は、実行できません。

## KMBJ00035-E

---

指定された操作は、アクセス権がないため、実行できません。システム管理者に連絡してください。

No access to the specified operation. Contact System Administrator.

要因

指定された操作に対するアクセス権がありません。

対処

アクセス権を変更して、再実行してください。

## KMBJ00036-E

---

指定されたパブリック ACL は、存在しません。有効な値を指定してください。

Specified public ACL does not exist. Specify a valid value.

要因

指定されたパブリック ACL は、存在しません。

対処

存在するパブリック ACL を指定してください。

## KMBJ00037-E

---

バインドされていないパブリック ACL です。バインドされているパブリック ACL を指定してください。

Unbound public ACL. Specify bound public ACL.

要因

指定されたパブリック ACL は、指定されたオブジェクトにバインドされていません。

対処

バインドされているパブリック ACL を指定して、再実行してください。

## KMBJ00038-E

---

保持できるレンディション数の上限に達しました。システム管理者に連絡してください。

Reached the maximum value of rendition count. Contact System Administrator.

要因

保持できるレンディション数の上限に達しました。

対処

不要なレンディションを削除して、新しいレンディションを追加してください。

## KMBJ00039-E

---

マスタレンディションが設定されていません。設定してから操作を実行してください。

Master Rendition is not Set. Set the value and carry out the operation.

要因

マスタレンディションの値が設定されていません。

対処

操作を実行する前に、マスタレンディションの値を設定してください。

## KMBJ00040-E

---

指定されたレンディションは既に存在しています。

Specified Rendition type already exists.

要因

指定されたレンディションはすでに存在しています。

対処

存在しないレンディションタイプのレンディションを指定してください。

## KMBJ00041-E

---

この操作は、マスタレンディションに対して実行できません。ほかの値を指定してください。

Operation cannot be carried out on master rendition. Specify another value.

要因

この操作は、マスタレンディションに対して実行できません。

対処

サブレンディションを指定して、再実行してください。



## KMBJ00042-E

---

コンテンツを更新できません。システム管理者に連絡してください。

Contents not updated. Contact System Administrator.

要因

コンテンツが更新されていません。

対処

コンテンツを更新して、再実行してください。

## KMBJ00043-E

---

存在しないレンディションタイプです。有効な値を指定してください。

Non existing Rendition type. Specify a valid value.

要因

指定されたレンディションタイプは、存在しません。

対処

存在するレンディションを指定して、再実行してください。

## KMBJ00044-E

---

バージョン識別子が予約されていません。

Version identifier is not reserved.

要因

バージョン識別子が予約されていないため、指定された操作は実行できません。

対処

チェックイン時に割り当てられた正しいバージョン識別子を設定して、再実行してください。

## KMBJ00045-E

---

アクセス権がありません。システム管理者に連絡してください。

No Access permission. Contact System Administrator.

要因

指定された操作を実行するためのパーミッションがありません。

対処

パーミッションを見直して、再実行してください。

## KMBJ00046-E

---

アクセス制御エレメントが不正です。有効な値を指定してください。

Invalid Access Control Element. Specify valid value.

要因

指定されたアクセス制御エレメントが不正です。

対処

有効なアクセス制御エレメントを指定して、再実行してください。

## KMBJ00047-E

---

サブジェクト種別が不正です。有効な値を指定してください。

Invalid subject classification. Specify valid value.

要因

指定されたサブジェクト種別が不正です。

対処

有効なサブジェクト種別を指定して、再実行してください。

## KMBJ00048-E

---

アクセス制御エレメントの個数の上限に達しました。システム管理者に連絡してください。

Reached the maximum value of Access Control Element count. Contact System Administrator.

要因

アクセス制御エレメントの個数の上限に達しています。

対処

アクセス制御エレメントは、64 個までにしてください。

## KMBJ00049-E

---

サブジェクトの最大長に達しました。システム管理者に連絡してください。

Reached the maximum length for subject. Contact System Administrator.

要因

サブジェクトの長さが許容範囲を超えています。

対処

サブジェクトの長さは、1 ~ 254 文字の範囲にしてください。

## KMBJ00050-E

---

バインドするパブリック ACL の個数の上限に達しました。システム管理者に連絡してください。

Reached the maximum value for public ACL to be bound. Contact System Administrator.

要因

バインドするパブリック ACL の個数の上限を超えています。

対処

バインドするパブリック ACL の個数は、10 個までにしてください。

## KMBJ00051-E

---

唯一のバージョンであるため、削除できません。

Cannot delete last Version Identifier.

要因

指定されたバージョンは唯一のバージョンであるため、削除できません。

対処

この操作は、実行できません。

## KMBJ00052-E

---

セッションは有効です。再度ログインする必要はありません。

Session still valid .No need to login again.

要因

すでにログインしています。

対処

セッションは有効です。再度ログインする必要はありません。

## KMBJ00053-E

---

指定された引数 %4 が null です。有効な値を指定してください。

Specified argument %4 is null. Provide valid value.

要因

指定された引数 %4 が null です。

対処

有効な値を指定してください。

## KMBJ00054-E

---

ログアウト中にエラーが発生しました。システム管理者に連絡してください。

Error occurred while logging out. Contact System Administrator.

要因

ログアウト中にエラーが発生しました。

対処

システム管理者に連絡してください。

## KMBJ00055-E

---

セッションの状態をチェック中にエラーが発生しました。システム管理者に連絡してください。

Error occurred while checking session status. Contact System Administrator.

要因

セッションの状態をチェック中にエラーが発生しました。

対処

システム管理者に連絡してください。

## KMBJ00056-E

---

オブジェクトの型 %4 が不正です。有効な値を指定してください。

Invalid Object type %4. Specify valid value.

要因

指定されたオブジェクトの型が不正です。%4

対処

有効なオブジェクトの型を指定して、再実行してください。

## KMBJ00057-E

---

インデクスが不正です。有効な値を指定してください。

Invalid Index . Specify valid value.

要因

指定されたインデクスが不正です。

対処

有効なインデクスを指定してください。

## KMBJ00058-E

---

指定されたプロパティ %4 が見つかりません。有効なプロパティを指定してください。

The specified Property %4 is not found. Specify a valid Property value.

要因

指定されたプロパティ %4 が見つかりません。

対処

存在するプロパティを指定して、再実行してください。

## KMBJ00059-E

---

ファイル %4 が見つかりません。有効なパス・ファイルを指定してください。

File %4 not Found. Specify a valid path / file.

要因

指定されたファイル %4 が見つかりません。

対処

存在するファイルを指定してください。

## KMBJ00060-E

---

リンクタイプが不正です。有効なリンクタイプを指定してください。

Invalid Link Type. Specify valid link type.

要因

指定されたリンクタイプが不正です。

対処

有効なリンクタイプを指定してください。

## KMBJ00061-E

---

要求されたファイルに対してアクセス権がありません。システム管理者に連絡してください。

No Access permission for the requested file. Contact System Administrator.

### 要因

要求されたファイルに対してアクセス権がありません。

### 対処

アクセス権を見直して、再実行してください。

## KMBJ00062-E

---

指定されたクラス名 %4 が見つかりません。有効なクラス名を指定してください。

The specified class name %4 is not found. Specify valid class name.

### 要因

指定されたクラス名 %4 が見つかりません。

### 対処

有効なクラス名を指定して、再実行してください。

## KMBJ00063-E

---

環境変数 EDMCLASSDEFPATH に空白が指定されています。正しい値を指定してください。

Blank specified for EDMCLASSDEFPATH environment variable. Set the value properly.

### 要因

環境変数 EDMCLASSDEFPATH に空白が指定されています。

### 対処

環境変数 EDMCLASSDEFPATH に正しい値を指定してください。

## KMBJ00064-E

---

環境変数 EDMCLASSDEFPATH に値が設定されていないか、値が不正です。正しい値を指定してください。

EDMCLASSDEFPATH variable not set or the path is invalid. Set the value properly.

### 要因

環境変数 EDMCLASSDEFPATH に値が設定されていないか、値が不正です。

### 対処

環境変数 EDMCLASSDEFPATH に正しい値を指定してください。

## KMBJ00065-E

---

指定されたファイルに対する読み取り権がありません。システム管理者に連絡してください。

No Read permission for the specified file. Contact System Administrator.

### 要因

クラス定義情報ファイルに対する読み取り権がありません。

対処

ファイルのアクセス権を見直して、再実行してください。

## KMBJ00066-E

---

指定されたファイルはクラス定義情報ファイルではありません。有効なファイルを指定してください。

The specified file is not a Class-Definition Information file. Specify a valid file.

要因

指定されたファイルはクラス定義情報ファイルではありません。

対処

クラス定義情報ファイルを指定してください。

## KMBJ00067-E

---

環境変数 EDMCLASSDEFPATH に指定されたディレクトリが存在しないか、アクセスできません。有効な値を指定してください。

The directory specified in the EDMCLASSDEFPATH environment variable does not exist or cannot be referenced. Specify a valid value.

要因

環境変数 EDMCLASSDEFPATH に指定されたディレクトリが存在しないか、アクセスできません。

対処

操作を実行する前に、有効なディレクトリパスを指定してください。

## KMBJ00068-E

---

ファイルの内容が不正です。変換処理が必要です。

Invalid File status. Conversion process is to be done on the file.

要因

指定されたファイルの内容は変換処理されていないため、空の状態です。

対処

変換処理を実行して、空の状態を回避してください。

## KMBJ00069-E

---

XML ファイルの内容が不正です。XML の構文解析でエラーが発生しました。

Improper contents in the XML file. Error occurred during the XML file syntax analysis process.

要因

XML の構文解析でエラーが発生しました。

対処

XML ファイルの内容を確認してください。

## KMBJ00070-E

---

パラメタのクラス名又はプロパティの組み合わせが不正です。有効な値を指定してください。

The parameters Class Name and/or Property Set values are invalid.Specify proper values.

**要因**

パラメタのクラス名またはプロパティの組み合わせが不正です。

**対処**

有効なクラス名とプロパティの組み合わせを指定してください。

## KMBJ00071-W

---

@ を付けたプロパティが指定されています。

Properties with @ Mark are specified.

**要因**

@ を付けたプロパティが、バージョン付きオブジェクトに対して指定されています。

**対処**

@ を付けたプロパティを、バージョン付きオブジェクトに対して指定しないでください。

## KMBJ00072-E

---

サポートしていない URL プロトコルです。file:/// で始まっているか確認してください。

Unsupported URL protocol is used. Confirm whether URL starts by file:///.

**要因**

サポートしていない URL プロトコルです。

**対処**

file:/// で始まっているか確認してください。

## KMBJ00073-E

---

ファイル転送サービスへのセッションが切断されました。ネットワーク環境を確認してください。又は、時間をおいて、再度実行してください。

The file transfer service session is disconnected.Confirm network environment or re-execute after some time.

**要因**

ファイル転送サービスへのセッションが切断されました。

**対処**

ネットワーク環境を確認してください。または、時間をおいて、再度実行してください。

## KMBJ00074-E

---

ファイル転送サービスに同時に接続できるセッション数の上限に達しています。時間をおいて、Connect ユーザ数が減った後、再度実行してください。又は、ファイル転送サービス環境定義ファイルの FtpSessionMax エントリを見直してください。

The limit of number of sessions that can be connected simultaneously to file transfer service has reached. Execute the process again after some time when number of user connected reduces or change FtpSessionMax entry of FTP service environment definition file.

要因

ファイル転送サービスに同時に接続できるセッション数の上限に達しています。

対処

時間をおいて、Connect ユーザ数が減ったあと、再度実行してください。または、ファイル転送サービス環境定義ファイルの FtpSessionMax エントリを見直してください。

## KMBJ00075-E

---

ファイル転送サービスが利用できません。サーバのバージョンに対応したクライアントを利用してください。システム管理者に連絡してください。

File transfer service is not available. Use client corresponding to server version. Contact System Administrator.

要因

ファイル転送サービスが利用できません。

対処

サーバのバージョンに対応したクライアントを利用してください。

## KMBJ00076-E

---

使用できるファイル転送サービスがありません。

File transfer service cannot be used.

要因

使用できるファイル転送サービスがありません。

対処

1. ネットワーク環境を確認してください。
2. \_HIEDMS\_FTPMODE=STATIC を指定してファイル転送サービスが開始されているかどうかを、確認してください。
3. 時間をおいて、再度実行してください。
4. ファイル転送サービスのセットアップコマンドが実行されているかどうかを、確認してください。

## KMBJ00077-E

---

ファイルの URL が不正です。正しい URL を指定してください。

The URL for the file is invalid. Provide correct URL.

要因

ファイルの URL が不正です。

対処

URL の内容を見直してください。

## KMBJ00078-E

---

現在のオブジェクトからデータベースへの論理的な接続が失われました。操作は完了できません。



Logical connection to database is lost. Operation cannot be performed.

要因

現在のオブジェクトからデータベースへの論理的な接続が失われました。操作は完了できません。

対処

文書空間に再接続してください。

## KMBJ00079-E

---

指定されたバージョンは存在しません。バージョン識別子を見直してください。

The specified version does not exist. Reconsider the specified version Id.

要因

指定されたバージョンは存在しません。

対処

正しいバージョンを指定して、操作を再度実行してください。

## KMBJ00080-E

---

使用できるサービスプロセスがありません。システム管理者に連絡してください。

There is no usable service process. Contact System Administrator.

要因

使用できるサービスプロセスがありません。

対処

時間をおいて、再度実行してください。

## KMBJ00081-E

---

ネットワーク環境でエラーが発生しました。システム管理者に連絡してください。

Error occurred in the network environment. Contact System Administrator.

要因

ネットワーク環境でエラーが発生しました。

対処

ネットワーク環境を確認してください。

## KMBJ00082-E

---

サーバのバージョンが対応していないため、接続できません。システム管理者に連絡してください。

The connection to incompatible server is not possible. Contact System Administrator.

要因

サーバのバージョンが対応していないため、接続できません。

対処

サーバのバージョンに対応したクライアントライブラリを利用してください。

## KMBJ00083-E

---

制約違反の操作です。システム管理者に連絡してください。

An operation violating the restrictions is been performed. Contact System Administrator.

要因

制約違反の操作です。

対処

この操作は実行できません。

## KMBJ00084-E

---

この操作を実行するために必要なネットワークが利用できません。システム管理者に連絡してください。

The network cannot be used to perform this operation. Contact System Administrator.

要因

この操作を実行するために必要なネットワークが利用できません。

対処

この操作は実行できません。

## KMBJ00085-E

---

オブジェクトが初期化されていません。初期化してから再実行してください。

Object is not initialized. Re-execute after initialization.

要因

オブジェクトが初期化されていません。

対処

初期化を実行してください。

## KMBJ00086-E

---

DMA 規約で禁止されている操作です。

This operation is prohibited by DMA agreement.

要因

DMA 規約で禁止されている操作です。

対処

この操作は実行できません。

## KMBJ00087-E

---

開始インデックスが不正です。正しい開始インデックスを指定してください。

The start index is invalid. Provide proper start index.

要因

開始インデックスが不正です。

対処

正しい開始インデックスを指定してください。

## KMBJ00088-E

---

ブロックサイズに 0 は指定できません。正しいブロックサイズを指定してください。

The block size cannot be zero. Provide proper block size.

### 要因

ブロックサイズに 0 は指定できません。

### 対処

正しいブロックサイズを指定してください。

## KMBJ00089-W

---

fetch カウントが許容範囲を超えています。

The Fetch Count has gone out of range.

### 要因

fetch カウントが許容範囲を超えています。

## KMBJ00090-W

---

構成管理型のコンテインメント (FIX モード又は FLOATING モード) は、解除できません。

Cannot unlink VT Fix / Float links.

### 要因

構成管理型のコンテインメント (FIX モードまたは FLOATING モード) を解除しようとした。

## KMBJ00091-W

---

レンディション定義の初期化に失敗しました。

Mime Map Manager Initialization failed.

### 要因

レンディション定義ファイルの読み込み中に、エラーが発生しました。

### 対処

システム管理者に連絡してください。

## KMBJ00092-E

---

指定されたプロパティの値が不正です。

Value specified by the user for property is invalid.

### 要因

不正な値のプロパティが指定されました。

### 対処

正しい値を指定してください。

## KMBJ00093-E

---

文書は、既にチェックアウトされています。

The document is already checked out.

要因

文書は、すでにチェックアウトされています。

対処

文書をチェックインしてから、操作してください。

## KMBJ00094-E

---

指定されたフェッチカウントが不正です。正しいフェッチカウントを指定してください。

Specified Fetch Count is invalid. Specify a valid Fetch Count.

要因

指定されたフェッチカウントが不正です。

対処

正しいフェッチカウントを指定してください。

## KMBJ00095-E

---

文書空間オブジェクトは、指定されたパブリック ACL で、既にバインドされています。

Document space object is already bound with the specified public ACL.

要因

文書空間オブジェクトは、指定されたパブリック ACL で、すでにバインドされています。

対処

指定したパブリック ACL を見直してください。

## KMBJ00096-E

---

指定された edmSQL 文が不正です。理由：%4 正しい edmSQL 文を指定してください。

Specified edmSQL statement is incorrect, Reason: %4. Specify valid edmSQL statement.

要因

指定された edmSQL 文が不正です。理由：%4。

対処

正しい edmSQL 文を指定してください。

## KMBJ00097-E

---

指定された ? パラメタの値が不正です。正しい ? パラメタの値を指定してください。

Specified '?' parameter value is incorrect. Specify valid '?' parameter value.

要因

指定された ? パラメタの値が不正です。

対処

正しい? パラメタの値を指定してください。

## KMBJ00098-E

---

引数で指定されたパラメタ数は、edmSQL 文で指定された? パラメタ数より大きいです。正しいパラメタを指定してください。

The number of parameters specified as argument is greater than the number of '?' parameters specified in edmSQL statement. Specify valid parameter.

要因

引数で指定されたパラメタ数は、edmSQL 文で指定された? パラメタ数より大きいです。

対処

正しいパラメタを指定してください。

## KMBJ00099-E

---

フェッチカウントの最大値が不正です。正しい値を指定してください。

The value of max fetch count is invalid. Provide a valid value.

要因

フェッチカウントの最大値が不正です。

対処

正しい値を指定してください。

## KMBJ00100-E

---

XML ライブラリのローディング処理に失敗しました。

An error occurred during Loading of the library.

要因

XML ライブラリのローディング処理に失敗しました。

対処

XML 管理機能に必要な環境変数が設定されているかどうか確認してください。

## KMBJ00101-E

---

指定されたアクセス制御モードが不正です。正しい値を指定してください。

The value specified for ACL mode is invalid. Provide a valid value.

要因

不正なアクセス制御モードが指定されました。

対処

正しいアクセス制御モードを指定してください。

## KMBJ00102-E

---

指定されたコンパレータの値が不正です。正しい値を指定してください。

Specified comparator is invalid. Provide valid comparator value.

要因

指定されたコンパレータの値が不正です。

対処

正しい値を指定してください。

## KMBJ00103-E

---

指定されたカーソルの位置が不正です。正しい値を指定してください。

Specified cursor level is invalid. Provide a valid value.

要因

指定されたカーソルの位置が不正です。

対処

正しい値を指定してください。

## KMBJ00104-E

---

指定されたオブジェクトに対して、実行された操作が不正です。名前なし検索結果集合に対して、カラム名の指定はできません。

The operation performed is invalid for the specified object. The resultset object is unnamed, and hence, column name cannot be specified.

要因

名前なし検索結果集合に対して、カラム名が指定されました。

対処

カラム名ではなく、列インデックスで指定してください。

## KMBJ00105-E

---

パブリック ACL のリストの要素に null が指定されました。

A null element is specified in the public ACL list.

要因

パブリック ACL のリストの要素の一つ以上が、null です。

対処

正しい要素を指定してください。

## KMBJ00106-E

---

GetVersionObjList メソッドに指定した order が不正です。正しい order を指定してください。

Order specified in GetVersionObjList() is invalid. Specify correct order.

要因

GetVersionObjList メソッドに不正な order が指定されました。

対処

正しい order を指定してください。

## KMBJ00107-E

---

指定されたクラス名は、255 文字を超えています。

The length of specified class name exceeds 255 characters.

要因

指定されたクラス名が、255 文字を超えています。

対処

255 文字以下のクラス名を指定してください。

## KMBJ00108-E

---

指定されたレンディションタイプは空です。正しい値を指定してください。

Specified rendition type is empty. Provide a valid value.

要因

空のレンディションタイプが指定されました。

対処

正しい値を指定してください。

## KMBJ00109-E

---

?パラメタリストの要素に指定されたオブジェクトタイプが不正です。正しいオブジェクトタイプを指定してください。

The specified object type passed in the element of '?' parameter list is invalid. Specify a valid object type.

要因

不正なオブジェクトタイプが、?パラメタリストの要素に指定されました。

対処

正しいオブジェクトタイプを指定してください。

## KMBJ00110-E

---

指定されたインデックスのパスが不正です。正しいインデックスのパスを指定してください。

The specified index path is invalid. Specify a valid index path.

要因

不正なインデックスのパスが指定されました。

対処

正しいインデックスパスを指定してください。

## KMBJ00111-E

---

リストに指定された値が不正です。正しい値を指定してください。

Specified value(s) passed in the list is/are invalid. Specify valid values.

要因

リストに不正な値が指定されました

対処

リストに正しい値を指定してください。

## KMBJ00112-E

---

指定されたフォルダオブジェクト %4 が不正です。正しいフォルダオブジェクトを指定してください。

Specified folder object %4 is invalid. Specify a valid folder object.

要因

不正なフォルダオブジェクトが指定されました。

対処

正しいフォルダオブジェクトを指定してください。

## KMBJ00113-E

---

文書のアップロード情報 (DbjUploadInfo) が null です。正しい文書のアップロード情報を指定してください。

The DbjUploadInfo passed is null. Specify a valid DbjUploadInfo.

要因

文書のアップロード情報 (DbjUploadInfo) として null が指定されました。

対処

正しい文書のアップロード情報を指定してください。

## KMBJ00114-E

---

指定されたパスが不正です。正しいパスを指定してください。

The specified file path(s) is/are invalid. Specify valid path.

要因

不正なパスが指定されました。

対処

正しいパスを指定してください。

## KMBJ00115-E

---

不正な引数が指定されました。(Method=%4)

IllegalArgument is specified(Method=%4).

要因

指定された引数が不正です。

対処



正しい引数を指定してください。

## KMBJ00116-W

---

logout メソッドが呼ばれていないため、自動的に logout メソッドを実行しました。

Logout method has not been called, so the Logout method has been called automatically.

### 要因

logout メソッドが実行されていないため、logout メソッドが実行されました。

## KMBJ00117-E

---

指定されたリンク識別子が不正です。正しい値を指定してください。

Specified Link ID is invalid. Specify valid link ID.

### 要因

不正なリンク識別子が指定されました。

### 対処

正しいリンク識別子を指定してください。

## KMBJ00118-W

---

レンディション定義ファイル (mime.properties) が、見つかりません。

File "mime.properties" was not found.

### 要因

レンディション定義ファイル (mime.properties) が、見つかりません。

### 対処

レンディション定義ファイル (mime.properties) を作成してください。

## KMBJ00119-E

---

DbjUploadInfo インタフェースのメソッドで指定されたファイルのパスが不正です。正しいパスを指定してください。

The file path specified in DbjUploadInfo is null. Specify a valid file path.

### 要因

DbjUploadInfo インターフェースのメソッドに不正なパスが指定されました。

### 対処

正しいパスを指定してください。

## KMBJ00120-E

---

指定された操作が失敗しました。

The specified operation has failed.

### 要因

指定された操作が失敗しました。次のどれかの要因が考えられます。

1. 操作が失敗しました。
2. メタ情報ファイルが不正です。
3. メモリ不足で実行できません。

対処

次のどれかの方法で対処してください。

1. 時間をおいて再度実行するか、処理を見直してください。
2. メタ情報ファイルを見直してください。
3. メモリを増設してください。またはメモリを大量に消費しているアプリケーションを停止させて、処理を再実行してください。

## KMBJ00121-E

---

ログインが失敗しました。

Login failed.

要因

DocumentSpace 構成定義ファイルの ProcessOrbBoaOption エントリに、-OathreadMax オプションを 129 以上の値で指定している環境で、同時に 129 以上のクライアントから DbjSession#login が実行されました。

対処

時間をおいて再度実行するか、-OathreadMax オプションの値を 128 以下にしてください。

## KMBJ00122-E

---

制限時間を超えました。

The restricted time is exceeded.

要因

制限時間を超えました。

対処

ログインしてから、再度実行してください。

## KMBJ00123-E

---

XML の定義情報が不正です。

The XML definition information is invalid.

要因

XML の定義情報が不正です。

対処

XML の定義情報を見直してください。

## KMBJ00124-E

---

接続先サーバでエラーが発生しました。(Major Code:%4, Minor Code:%5)

Error occurred in connected server(Major Code : %4 and Minor Code:%5).

## 要因

DocumentBroker でエラーが発生しました。

## 対処

システム管理者に連絡してください。

## KMBJ00125-E

---

コンテンツの操作でエラーが発生しました。

Error occurred in the contents operation.

## 要因

コンテンツの操作でエラーが発生しました。

## 対処

システム管理者に連絡してください。

## KMBJ00126-E

---

オブジェクトとコンテンツの状態が不整合です。

The state of contents and the object is mismatched.

## 要因

オブジェクトとコンテンツの状態が不整合です。

## 対処

システム管理者に連絡してください。

## KMBJ00127-E

---

リファレンスタイプが不正です。正しいリファレンスタイプを指定してください。

Specified reference type is invalid. Specify the valid reference type.

## 要因

リファレンス種別が不正です。

## 対処

システム管理者に連絡してください。

## KMBJ00128-E

---

相対パスのベースとなるパスが指定されていません。相対パスのベースとなるパスを指定してください。

The path used as the base of relative path is not specified. Specify the path used as the base of relative path.

## 要因

相対パスのベースとなるコンテンツ格納先ベースパスが設定されていません。

## 対処

システム管理者に連絡してください。

## KMBJ00129-E

---

指定したコンテンツタイプが不正です。正しい文書オブジェクトを指定してください。

Specified content type is invalid. Specify valid document object.

要因

指定したコンテンツ種別が不正です。

対処

システム管理者に連絡してください。

## KMBJ00132-E

---

カレントバージョンのコンテンツが存在しません。

The contents of current version is not found.

要因

カレントバージョンのコンテンツが存在しません。

対処

システム管理者に連絡してください。

## KMBJ00133-E

---

コンテンツが登録されていません。

The contents is not registered.

要因

コンテンツが登録されていません。

対処

コンテンツを登録してから処理を行ってください。

## KMBJ00134-E

---

開始インデックスが不正です。0以上の開始インデックスを指定してください。

The start index is invalid. Provide start index of 0 or more.

要因

開始インデックスが不正です。

対処

0以上の開始インデックスを指定してください。

# 付録

---

付録 A 文書空間で使用する文字コード種別が UTF-8 の場合の機能差異

---

付録 B 用語解説

---

---

## 付録 A 文書空間で使用する文字コード種別が UTF-8 の場合の機能差異

ここでは、文書空間で使用する文字コード種別が UTF-8 の場合のクラス、インターフェースおよびメソッドの機能差異について説明します。なお、文書空間で使用する文字コード種別が Shift-JIS の場合は、すべてのクラス、インターフェースおよびメソッドを使用できます。

### (1) ファクトリクラスのメソッド

文書空間で使用する文字コード種別が UTF-8 の場合、次のメソッドは使用できません。

DbjFactory インターフェース

- createXmlTranslator メソッド
- createXmlUploadInfo メソッド

そのほかのメソッドはすべて使用できます。

### (2) パラメタクラスのメソッド

文書空間で使用する文字コード種別が UTF-8 の場合、次のメソッドは使用できません。

DbjMappedProp インターフェース

- getClassList メソッド
- propSet メソッド

DbjXmlUploadInfo インターフェース

- getFilterFilePath メソッド
- getIndexType メソッド
- getMappingId メソッド
- getParseLevel メソッド
- setFilterFilePath メソッド
- setIndexType メソッド
- setMappingId メソッド
- setParseLevel メソッド
- clearParseMessage メソッド
- getParseMessage メソッド

そのほかのメソッドはすべて使用できます。

### (3) 文書管理クラスのメソッド

文書空間で使用する文字コード種別が UTF-8 の場合、次のメソッドは使用できません。

DbjXmlTranslator インターフェース

- getMappedProperties メソッド

そのほかのメソッドはすべて使用できます。

### (4) メタクラスのメソッド

文書空間で使用する文字コード種別が UTF-8 の場合、すべてのメソッドが使用できます。

(5) 例外クラス

文書空間で使用する文字コード種別が UTF-8 の場合、すべてのクラスが使用できます。

(6) 定数定義クラス

文書空間で使用する文字コード種別が UTF-8 の場合、すべてのクラスが使用できます。

(7) ライブラリ情報取得クラスのメソッド

文書空間で使用する文字コード種別が UTF-8 の場合、すべてのメソッドが使用できます。

(8) トレースクラスのメソッド

文書空間で使用する文字コード種別が UTF-8 の場合、すべてのメソッドが使用できます。

---

## 付録 B 用語解説

Java クラスライブラリで使用する用語について説明します。

ここで説明していない Java クラスライブラリの用語については、マニュアル「DocumentBroker Version 3 システム導入・運用ガイド」を参照してください。

---

### (記号)

#### ?パラメタ

edmSQL 文中のパラメタ値を、検索条件を指定するときには固定しないでおいて、検索実行時にアプリケーションから値を設定するために使用するパラメタです。edmSQL 文中の ? パラメタによって値を渡す個所には、定数を指定する代わりに「?」を指定しておきます。

Java クラスライブラリでは、DbjQParam インターフェースのサブインターフェースによって、? パラメタの値を設定できます。

---

### (英字)

#### ACE

Access Control Element の略です。アクセス制御エレメントのことです。

#### ACFlag

Access Control Flag の略です。アクセス制御フラグのことです。

#### ACL

Access Control List の略です。アクセス制御リストのことです。

#### AND 条件

検索条件同士の論理積を求める結合条件です。例えば、「著者が『日立太郎』で、文書のコンテンツ中に『コンピュータ』という文字列を含む文書を検索する」というような場合に使用できます。

#### API ( Application Programming Interface )

アプリケーションプログラムとのインターフェースです。

#### ContainerVersion オブジェクト

DMA クラスの edmClass\_ContainerVersion クラスまたはそのサブクラスを基に作成された DMA オブジェクトです。

#### Container オブジェクト

DMA クラスの dmaClass\_Container クラス、または edmClass\_ContainerVersion クラス以外の dmaClass\_Container クラスのサブクラスを基に作成された DMA オブジェクトです。

#### CORBA ( Common Object Request Broker Architecture )

OMG ( Object Management Group ) が提唱するオブジェクト間の通信メカニズムを提供する ORB ( Object Request Broker ) の標準アーキテクチャです。

#### DMA ( Document Management Alliance )

文書管理インターフェースの標準化を図る団体 AIIM ( Association for Information and Image Management International ) によって定義される共通インターフェースです。



### dmaClass\_ConfigurationHistory クラス

バージョン管理に使用する DMA クラスです。

dmaClass\_ConfigurationHistory クラスまたはそのサブクラスは、バージョンングオブジェクトのトップオブジェクトクラスになります。

### dmaClass\_Container クラス

オブジェクトをディレクトリに格納するイメージでリンクしてまとめて管理したり、オブジェクトに分類を付けるイメージでリンクして関連付けて管理したりできるフォルダを表す DMA クラスです。

dmaClass\_Container クラスまたはそのサブクラスは、構成管理できないバージョンなしフォルダのトップオブジェクトクラスになります。

### dmaClass\_DocVersion クラス

文書を表す DMA クラスです。

dmaClass\_DocVersion クラスまたはそのサブクラスは、バージョンなし文書のトップオブジェクトクラスになります。また、構成管理できないバージョン付き文書のバージョンオブジェクトのトップオブジェクトクラスになります。

### DMA オブジェクト

DMA クラスを基に作成されたオブジェクトです。

### DMA クラス

DMA のオブジェクトモデルに基づいたクラス、DocumentBroker で定義しているクラスおよびそのサブクラスです。

Java クラスライブラリの文書空間オブジェクトクラスのプロパティは、DMA クラスに定義します。

### DocVersion オブジェクト

DMA クラスの dmaClass\_DocVersion クラス、または edmClass\_VersionTracedDocVersion クラス以外の

dmaClass\_DocVersion クラスのサブクラスを基に作成された DMA オブジェクトです。

### edmClass\_ContainerVersion クラス

オブジェクトのまとまりをバージョン管理できるフォルダを表す DMA クラスです。オブジェクトをディレクトリに格納するイメージでリンクしてまとめて管理したり、オブジェクトに分類を付けるイメージでリンクして関連付けて管理したりできるフォルダを表す DMA クラスです。また、構成管理型リンクを利用して構成要素であるバージョン付き文書の特定期間を固定してリンクしたり、常に最新バージョンをトレースしてリンクしたりもできます。

edmClass\_ContainerVersion クラスまたはそのサブクラスは、バージョンなしフォルダのトップオブジェクトクラスになります。また、バージョン付きフォルダのバージョンオブジェクトのトップオブジェクトクラスになります。

### edmClass\_IndependentPersistence クラス

独立データを表す DMA クラスです。

edmClass\_IndependentPersistence クラスまたはそのサブクラスは、独立データのトップオブジェクトクラスになります。

### edmClass\_PublicACL クラス

パブリック ACL を表す DMA クラスです。文書やフォルダなどのオブジェクトとは独立に作成・参照でき、複数の文書やフォルダなどのオブジェクトから共有できるアクセス制御情報を表すクラスです。

パブリック ACL のトップオブジェクトクラスになります。

### edmClass\_Relationship クラス

文書間リンクを表す DMA クラスです。単独では作成できない、文書に従属する Relationship オブジェクトの基となる DMA クラスです。

### edmClass\_VersionTracedDocVersion クラス

文書を表す DMA クラスです。

edmClass\_VersionTracedDocVersion クラスまたはそのサブクラスは、バージョン付き文書のバージョンオブジェクトのトップオブジェクトクラスになります。

## edmSQL

文書空間オブジェクトを検索する場合に使用する、検索条件式を表現するための文法です。SQL の文法に基づいています。

## edmSQL 検索

検索条件に、SQL の文法に基づいた文法で記述できる edmSQL 文を指定して実行する検索のことです。

## getter メソッド

パラメタクラスのインターフェースで扱うオブジェクトのプロパティの値を取得するためのメソッドです。

## GUID

Globally Unique Identifier の略です。DMA のクラス、プロパティなどに与えるユニークな識別子です。GUID は、「X」を 0 ~ 9, a ~ f (小文字), および A ~ F (大文字) で表される 16 進数とした「XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX (8 けた -4 けた -4 けた -4 けた -12 けた)」の形式で表されます。

## Java クラスライブラリ

DocumentBroker サーバにアクセスするユーザアプリケーションプログラムを Java 言語で作成するために、DocumentBroker Development Kit および DocumentBroker Runtime が提供しているインターフェースです。

## MIME 形式

MIME::text/plain, MIME::text/html など、コンテンツのレンディションタイプを表す形式です。なお、DocumentBroker では、「MIME::」は省略して指定します。

## NOT 条件

指定したキーワードとの不一致を求める検索条件です。例えば、「作成者が『日立』ではない文書を検索する」というような場合に使用できます。

## OIID

Object Instance Identifier の略です。データベースに格納されたオブジェクトの存在や格納位置などを明確にするために、各オブジェクトに与えるユニークな識別子です。

## OR 条件

検索条件同士の論理和を求める結合条件です。例えば、「作成者が『日立太郎』である文書空間オブジェクトか、作成者の所属が『日立製作所』である文書空間オブジェクトを検索する」という場合、「文書のコンテンツ中に『コンピュータ』という文字列を含むか、『インターネット』という文字列を含む文書を検索する」という場合に使用できます。

## Proxy オブジェクト

文書空間オブジェクトを操作する場合に使用する代理オブジェクトです。Java クラスライブラリでは、文書空間オブジェクトを操作するとき、文書空間オブジェクトを直接操作するのではなく、Proxy オブジェクトを操作することで間接的に操作します。

Proxy オブジェクトはメモリ空間に存在します。不要になった場合は、Java のガベージコレクションによって削除されます。

## setter メソッド

パラメタクラスのインターフェースで扱うオブジェクトのプロパティの値を設定するためのメソッドです。

## VARRAY 型

文書空間オブジェクトのプロパティのデータ型の一つです。VARRAY 型のプロパティは、プロパティのデータ型に従った複数の値を可変長な一次元配列として持ちます。

## VersionTracedDocVersion オブジェクト

DMA クラスの edmClass\_VersionTracedDocVersion クラスまたはそのサブクラスを基に作成された DMA オブジェクトです。

## W3C ( World Wide Web Consortium )

HTML や XML など、WWW に関する技術の標準化を推進する非営利団体です。

## XML インデクスデータ作成機能

XML 文書のコンテンツからインデクスデータを作成して、XML 文書のプレーンテキスト形式の全文検索インデクスまたは構造指定検索用の全文検索インデクスを登録する機能です。XML 文書の構文解析やタグ情報の除去もあわせて実行できます。

## XML 構造指定検索

XML 文書の DTD で記述した論理構造 ( エLEMENT ) や ELEMENT の属性をキーにして、XML 文書を全文検索する方法です。例えば、「タイトルに『コンピュータ』という単語が含まれ、章に『XML』という単語が含まれる文書を検索する」というような場合に使用します。また、「『document』というELEMENT に設定されている属性『status』の属性値が『public』である文書を検索する」というような場合にも使用できます。

## XML トランスレーターオブジェクト

DbjXmlTranslator インターフェースを取得したときにメモリ空間に作成される概念的なオブジェクトです。

DbjXmlTranslator インターフェースの機能を実行できます。

## XML プロパティマッピング機能

XML 文書を構文解析して、DocumentBroker サーバのプロパティマッピング定義ファイルの定義に従って XML 文書内のタグ間の文字列または属性値を XML 文書のプロパティに割り当てて設定する機能です。

## XML プロパティマッピング結果情報

XML プロパティマッピング機能で作成される情報です。DbjMappedProp インターフェースで扱います。

## XML 文書のアップロード情報

XML 文書作成またはコンテンツ更新時に使用する情報です。XML 文書作成時に、XML プロパティマッピング機能および XML インデクスデータ作成機能を実行するために使用できます。また、XML 文書のコンテンツ更新時に XML インデクスデータ作成機能を実行するために使用できます。DbjXmlUploadInfo インターフェースで扱います。

## (ア行)

---

### アクセス権

文書空間オブジェクトを作成したり、すでに作成されているオブジェクトにアクセスしたりする権利です。

### アクセス制御エレメント ( ACE : Access Control Element )

アクセス制御リスト ( ACL ) の要素です。一つのサブジェクトと一つのパーミッションの組で構成され、指定されたサブジェクトに対して指定されたパーミッションの範囲のアクセス権を与えることを示す情報です。

### アクセス制御機能

DocumentBroker の文書空間での文書空間オブジェクトの作成や、管理されている文書やフォルダなどの文書空間オブジェクトに対する操作を、ユーザやグループごとに許可または制限する機能です。

### アクセス制御機能付き検索

アクセス制御機能を利用した文書空間で検索を実行した場合に、ユーザにアクセス権がない文書空間オブジェクトを検索結果として取得しない検索です。

### アクセス制御情報

アクセス制御されている文書空間で、ユーザがメソッドを実行する際に、アクセス権の判定に使用される情報です。

### アクセス制御情報変更権

文書空間オブジェクトに設定されているアクセス制御情報 ( ACFlag および ACL ) を変更する権利です。また、パブ

リック ACL をアクセス制御対象の文書空間オブジェクトに設定することを許可する権利も含まれます。なお、パブリック ACL のアクセス情報変更権には、パブリック ACL のユーザ定義プロパティの値を変更する権利が含まれます。

### アクセス制御フラグ (ACFlag : Access Control Flag)

文書空間オブジェクトの所有者、プライマリグループおよび全ユーザという区分でパーミッションを設定できるアクセス制御情報の一つです。

### アクセス制御リスト (ACL : Access Control List)

任意のユーザまたはグループにパーミッションを設定できるアクセス制御情報の一つです。アクセス制御エレメント (ACE) のリストで構成されます。

### アンバインド

パブリック ACL とのバインドを解除することです。

### 異表記展開検索

全文検索条件として指定する検索タームまたは検索タームの異表記を含む文書を検索する全文検索のオプションです。例えば、検索タームとして「バイオリン」を指定した場合に、「ヴァイオリン」という検索タームの異表記を含む文書も検索できます。

### インターフェース

Java クラスライブラリの機能を実行するためのメソッドが定義されている、クラスの仕様です。Java クラスライブラリでは、オブジェクトを操作するとき、まず、インターフェースを取得して、そのインターフェースで定義されているメソッドを実行する、という手順で操作します。

### 永続オブジェクト

データベースに格納されたオブジェクトです。

### 永続プロパティ

データベースに存在するプロパティです。

### オーナーオブジェクトプロパティ

リンク Proxy オブジェクトのプロパティです。ターゲットリンクオブジェクトによってリンクが設定されている文書空間オブジェクトのうち、リンク元である文書空間オブジェクトの Proxy オブジェクトが設定されます。

### オブジェクト作成権

オブジェクト作成権限で、文書空間オブジェクトを作成する権利を与えるパーミッションです。ユーザ権限定義ファイルに指定します。

### オブジェクト作成権限

文書空間オブジェクトを作成する権限で、ユーザ権限の一つです。ユーザ権限定義ファイルで定義します。オブジェクト作成権限を与えられたユーザおよびグループに属するユーザは、文書空間オブジェクトを作成するメソッドを実行できます。

### オブジェクト種別

文書空間オブジェクトの種類を表す定数です。オブジェクト種別には、バージョンなし文書、バージョン付き文書、バージョンなしフォルダ、バージョン付きフォルダ、独立データおよびパブリック ACL があります。

### オブジェクト種別プロパティ

Proxy オブジェクトのプロパティです。ターゲットオブジェクトのオブジェクト種別が設定されます。

### オブジェクト操作権限

文書空間内のすべての文書空間オブジェクトを、与えられた権限の範囲で操作する権利です。ユーザ権限の一つです。ユーザ権限定義ファイルで定義します。例えば、オブジェクト操作権限として基本プロパティ参照権を与えられたユーザおよびグループに属するユーザは、文書空間内のすべての文書空間オブジェクトのプロパティを参照できます。

## オブジェクトリファレンス

DMA オブジェクトへのリファレンスを示すプロパティの値です。例えば、`dmaProp_ParentContainer` プロパティの値がこれに当たります。Java クラスライブラリのメソッドでこの値を使用することはできませんが、`edmSQL` 文中では使用することができます。

## (カ行)

---

### 下位オブジェクト

フォルダを使用した文書管理をする場合に、フォルダからリンクを設定されている、リンク先オブジェクトです。例えば、文書をフォルダに格納する管理をしている場合は、文書が下位オブジェクトです。

### 概念検索

ユーザが任意に指定した文章や文字列を手がかりにして、その条件と似た概念を持つ文書を検索する方法です。全文検索の一種です。概念検索で指定する条件のことを種文章といいます。

### 可変長配列

VARRAY 型のプロパティの値です。Java クラスライブラリでは、`DbjVArray` インターフェースで扱います。

### 仮のバージョン識別子

チェックアウト中の仮のバージョンを識別するための識別子です。  
チェックアウト中のオブジェクトを参照・更新するときに使用します。この識別子はチェックアウト時に `DocumentBroker` によって設定される識別子であり、仮のバージョンに該当するオブジェクトの `OIID` とは異なります。

### カレントバージョン

バージョン付き文書またはバージョン付きフォルダが持つ複数のバージョンの中の、最新バージョンのことです。

### 基本コンテンツ更新権

基本パーミッションの一つで、バージョンなし文書またはバージョン付き文書のコンテンツを更新する権利を与えるパーミッションです。基本プロパティ参照権を含みます。また、全文検索インデクスを作成、削除する権利も含みます。

### 基本コンテンツ参照権

基本パーミッションの一つで、バージョンなし文書またはバージョン付き文書のコンテンツを参照する権利を与えるパーミッションです。基本プロパティ参照権を含みます。

### 基本削除権

基本パーミッションの一つで、文書空間オブジェクトを削除する権利を与えるパーミッションです。基本プロパティ参照権を含みます。

### 基本バージョン管理権

基本パーミッションの一つで、バージョン管理されている文書空間オブジェクトのバージョンを追加、削除する権利を与えるパーミッションです。基本プロパティ参照権を含みます。

### 基本パーミッション

ユーザおよびグループが文書空間オブジェクトに対して実行できる操作の範囲を定めるパーミッションの基本単位です。オブジェクト操作権限、アクセス制御フラグおよびアクセス制御エレメントで、ユーザおよびグループに許可する操作の範囲を定める場合に使用します。例えば、あるユーザに対して、文書の更新と削除を許可する場合は、更新と削除を許可するために、基本コンテンツ更新権と基本削除権という二つのパーミッションを設定します（一つのパーミッションで一つの権利を与えます）。基本パーミッションには、基本プロパティ参照権、基本プロパティ更新権、基本コンテンツ参照権、基本コンテンツ更新権、基本リンク権、基本バージョン管理権および基本削除権があります。

### 基本プロパティ更新権

基本パーミッションの一つで、文書空間オブジェクトのプロパティを更新する権利を与えるパーミッションです。基本プロパティ参照権を含みます。

## 基本プロパティ参照権

基本パーミッションの一つで、文書空間オブジェクトのプロパティを参照する権利を与えるパーミッションです。そのほかのすべての基本パーミッションに含まれます。フォルダに対して設定すると、フォルダのリンクをたどることも許可します。

## 基本リンク権

基本パーミッションの一つで、リンクを設定・解除する権利を与えるパーミッションです。基本プロパティ参照権を含みます。

## キャッシュキー

キャッシュ付き検索を実行する場合に、2回目以降の検索条件が前回の検索条件と同じかどうかを判断するためのキーです。検索条件が同じ場合、検索結果キャッシュに対して検索が実行されます。検索条件が異なる場合、検索結果キャッシュは破棄されて、文書空間に対して検索が再度実行されます。

## キャッシュ検索

検索結果キャッシュを使用する検索です。文書空間に対して検索を実行したときに、取得した検索結果をキャッシュに保存しておき、同じ検索条件の2回目以降の検索では検索結果キャッシュから検索結果を取得します。同じ検索条件で複数回検索を実行する場合に、検索結果取得処理が高速になります。例えば、10,000件検索結果がある場合に、これを検索結果キャッシュとして保存しておき、ユーザアプリケーションプログラムではこれを100件ずつ取得する、という検索ができます。

## キャッシュ名

検索結果キャッシュの名称です。検索結果キャッシュは、キャッシュ名によって識別されます。

## 近傍条件検索

全文検索条件に複数の検索タームを指定した場合に、検索ターム間の距離を条件にして検索する方法です。例えば、「『文書管理』という検索タームと『ドキュメント』という検索タームを含み、これらの検索タームがこのとおりの順序で出現し、かつ検索ターム間に入る文字が5文字以内である文書を検索する」というような場合に使用できます。

## 組み合わせパーミッション

基本パーミッションを複数組み合わせた権利を与えるパーミッションの単位です。アクセス制御フラグおよびアクセス制御エレメントでユーザおよびグループに許可する操作の範囲を定める場合に使用します。組み合わせパーミッションには、プロパティ参照権、参照権、プロパティ更新権、参照更新権、削除権、リンク権、バージョン管理権およびフルコントロールがあります。例えば、あるユーザに対して、ある文書の参照更新権という組み合わせパーミッションを設定すると、そのユーザは基本プロパティ参照権、基本プロパティ更新権、基本コンテンツ参照権および基本コンテンツ更新権が設定されたのと同じ範囲の操作を、その文書に対して実行できます。

## クラス定義情報ファイル

DocumentBroker サーバで定義されている DMA オブジェクトのクラスまたはそのサブクラスのクラス名を取得したり、プロパティ名から GUID、データ型などの情報を取得したりするために使用するファイルです。DocumentBroker サーバの EDMCrtSimMeta コマンドで作成できます。DMA クラスやプロパティに名前アクセスする場合に Java クラスライブラリが参照します。

## クラスディスクリプション

DMA クラスに関するメタ情報です。DMA クラスごとに定義されています。

## 継承

既存のクラスを利用して新しいクラスを定義するオブジェクト指向の技術です。

## 検索結果キャッシュ

キャッシュ付き検索実行時に作成される検索結果のキャッシュです。キャッシュ付き検索では、例えば、10,000件の検索結果がある場合に、これを検索結果キャッシュとして保存しておき、ユーザアプリケーションプログラムではこれを100件ずつ取得する、という検索ができます。

### 検索結果集合

検索実行時に作成される、検索条件に一致する文書空間オブジェクトの集合です。要素がプロパティ値である、行と列の二次元データに、列名や列のデータ型などのメタデータを付けたものとして表されます。検索結果集合は、DbjResultSet インターフェースで扱います。

### 検索結果取得情報

キャッシュ付き検索実行時に、検索結果を何件検索結果キャッシュとして保存するか、また、検索結果キャッシュから何件ユーザアプリケーションプログラムに返却するかを指定する情報です。DbjFetchInfo インターフェースで扱います。

### 更新系メソッド

操作対象になるオブジェクトの状態を変化させるメソッドです。文書のコンテンツの更新や、文書やフォルダのプロパティの更新を実行するメソッドなどが該当します。

### 構成管理型リンク

フォルダにバージョン付きオブジェクトを関連付けるときに、バージョン付きオブジェクトの特定のバージョンを固定して関連付けたり、常に最新バージョンをトレースして関連付けたりできるリンクの種別です。構成管理モードとして、FIX モードと FLOATING モードがあります。

### 構造指定検索

XML 文書を管理している場合に、文書の論理構造や構造の属性をキーとして検索する方法です。例えば、「タイトルに『コンピュータ』という単語が含まれ、章に『XML』という単語が含まれる文書を検索する」というような場合に使用します。また、「『document』というエレメントに設定されている属性『status』の属性値が『public』である文書を検索する」というような場合にも使用できます。

### コレクション

java.util.Collection インターフェースの仕様に従うオブジェクトです。要素オブジェクトの集合を表します。

### コンテンツ

文書のデータ部分を指します。DMA で規定されているコンテンツ管理モデルに従ってアクセスされるオブジェクトの実体（例えば、report.doc、document.htm など）です。

### コンテンツ情報

文書のコンテンツ管理モデルで管理されているコンテンツに関する情報です。ファイル名およびレンディションタイプが管理されています。DbjContentInfo インターフェースで扱います。

## ( サ行 )

---

### 削除権

組み合わせパーミッションの一つです。基本削除権と同じ操作を許可するパーミッションです。

### サブインターフェース

あるインターフェースから派生するインターフェースのことです。上位インターフェースで定義されているメソッドを継承します。

### サブクラス

あるクラスから派生するクラスのことです。または、それ自身がサブクラスとして参照されているクラスのことです。

### サブジェクト

アクセス権を与えるユーザまたはグループです。

### サブジェクト種別

アクセス権を与えるサブジェクトが、ユーザなのか、グループなのかまたはシステムなのかを識別するための情報です。

## サブレンディション

マルチレンディション文書のマスタレンディション以外のレンディションです。なお、サブレンディションは、登録後にマスタレンディションに変更できます。

## 参照型リンク

一つのオブジェクトから複数のバージョンなしフォルダまたはバージョン付きフォルダを親として関連付けるリンクの種類です。一つの文書に対して複数の分類を付けるイメージでの管理を実現します。

## 参照系メソッド

操作対象になるオブジェクトの状態を変化させないメソッドです。文書のコンテンツの参照や、文書やフォルダのプロパティの参照を実行するメソッドなどが該当します。

## 参照権

組み合わせパーミッションの一つです。基本コンテンツ参照権と同じ操作を許可するパーミッションです。

## 参照更新権

組み合わせパーミッションの一つです。基本プロパティ参照権、基本プロパティ更新権、基本コンテンツ参照権および基本コンテンツ更新権を組み合わせたパーミッションです。すなわち、参照更新権を設定することで、プロパティを参照、更新する権利とコンテンツを参照、更新する権利を設定できます。

## サンプル Web アプリケーション

DocumentBroker Development Kit および DocumentBroker Runtime がサンプルとして提供しているアプリケーションプログラムです。Java クラスライブラリを使用して開発されたコンポーネントで構成されています。ユーザアプリケーションプログラムを開発する際に、アプリケーションプログラムのアーキテクチャを参考にできます。また、Java クラスライブラリのクラス、インターフェース、メソッドなどの具体的な使用方法について参考にできます。

## システム管理者

DocumentBroker の運用、管理および保守をするユーザです。DocumentBroker の実行環境を設定することができます。

## 上位オブジェクト

フォルダを使用した文書管理をする場合のリンク元オブジェクトです。つまり、フォルダのことです。

## 状態フラグ

マルチレンディション文書で、マスタレンディションに対するサブレンディションのコンテンツの状態を表すフラグです。マスタレンディションとサブレンディションのコンテンツの状態が一致している、マスタレンディションのコンテンツが更新されたのに対してサブレンディションのコンテンツが更新されていない、またはサブレンディションのコンテンツが存在しない、という 3 種類の状態が表されます。dbrProp\_RenditionStatus プロパティの下位 2 バイトに設定されます。

## 所有者

文書空間オブジェクトの所有者として設定されているユーザです。アクセス制御フラグでアクセス権を与えられます。所有者に設定されているユーザは、その文書空間オブジェクトのアクセス制御フラグで所有者に与えられたパーミッションの範囲の操作を、その文書空間オブジェクトに対して実行できます。また、その文書空間オブジェクトの所有者およびセキュリティ ACL の値を変更できます。

## スーパーインターフェース

あるインターフェースのインターフェース定義に使用されたインターフェースです。派生したインターフェースでは、スーパーインターフェースで定義されているメソッドを継承します。

## スーパークラス

あるクラスのクラス定義に使用されたクラスを、派生したクラスのスーパークラスといいます。

## セキュリティ ACL

文書空間オブジェクトに設定されたアクセス制御情報へのアクセスを制御するためのアクセス制御リストです。任意の



ユーザまたはグループにアクセス制御情報変更権を設定できます。

### セキュリティ運用者

DocumentBroker のアクセス制御の運用情報の管理者です。セキュリティ定義ファイルを保守します。

### セキュリティ管理者

アクセス制御機能を利用した文書空間で、アクセス権判定を受けることなく、すべての文書空間オブジェクトに自由にアクセスする特権を持ち、文書空間のすべての文書空間オブジェクトを保守するユーザです。セキュリティ定義ファイルに定義します。

### セキュリティ定義ファイル

アクセス制御の運用情報を定義するファイルです。セキュリティ管理者、ユーザ権限定義ファイル名、および文書空間オブジェクト作成時にアクセス制御フラグにデフォルトで設定されるパーミッションを定義します。

### セッション

文書空間に接続している間のことです。文書空間に接続することを、セッションの確立といいます。文書空間との接続を解除することを、セッションの切断といいます。

### セッションオブジェクト

セッションを確立する機能とセッション内のトランザクションを制御する機能を持つ、Java クラスライブラリのオブジェクトです。DbjSession インターフェースの機能を実行できます。

### セット

java.util.Set インターフェースの仕様に従うオブジェクトです。重複のない、要素オブジェクトの集合を表します。

### 全文検索

文書に含まれるキーワードを条件（全文検索条件）として、キーワードを含む文書を検索する方法です。

### 全文検索インデクス

文書を全文検索するために、データベースに登録するインデクスです。全文検索の対象になるテキストデータに対応する edmProp\_TextIndex プロパティ、edmProp\_StIndex プロパティ、edmProp\_ConceptTextIndex プロパティおよび edmProp\_ConceptStIndex プロパティに相当します。

### 全文検索機能付き文書クラス

dmaClass\_DocVersion クラスのサブクラスに全文検索に必要なプロパティを追加したサブクラスです。プロパティの追加は、ユーザが行います。

バージョンなし文書作成時のトップオブジェクトクラスの DMA クラス名として、またはバージョン付き文書作成時のバージョンオブジェクトのトップオブジェクトクラスの DMA クラス名として指定します。

edmSQL で全文検索を実行するときには、この DMA クラスを検索対象として FROM 句に指定します。

### 属性検索

文書空間オブジェクトのプロパティを対象にした検索です。例えば、「『文書名』が『X』で『作成者』が『A』の文書」のような条件を設定して検索する方法です。

## ( 夕行 )

---

### ターゲット OIID プロパティ

Proxy オブジェクトのプロパティです。ターゲットオブジェクトの OIID が設定されます。

### ターゲットオブジェクト

Proxy オブジェクトが対象にする文書空間オブジェクトのことです。

### ターゲットオブジェクトプロパティ

リンク Proxy オブジェクトのプロパティです。ターゲットリンクオブジェクトが関連付けている二つの文書空間オブ

ジェクトのうち、リンク先オブジェクトの Proxy オブジェクトが設定されます。  
例えば、フォルダと文書間の直接型リンクをターゲットリンクオブジェクトとするリンク Proxy オブジェクトの場合、ターゲットオブジェクトプロパティには文書の Proxy オブジェクトが設定されます。

### ターゲットバージョン

バージョン付きオブジェクトの複数のバージョンオブジェクトのうち、操作の対象になるバージョンです。

### ターゲットバージョン識別子プロパティ

Proxy オブジェクトのプロパティです。ターゲットバージョンであるバージョンオブジェクトのバージョン識別子が設定されます。

### ターゲットプロパティ値集合プロパティ

Proxy オブジェクトのプロパティです。文書空間オブジェクトからロードしたプロパティ値集合が設定されます。

### ターゲットリンクオブジェクト

リンク Proxy オブジェクトが対象にするリンクオブジェクトのことです。

### ターゲットリンク識別子プロパティ

リンク Proxy オブジェクトのプロパティです。ターゲットリンクオブジェクトのリンク識別子が設定されます。

### 種文章

概念検索の検索条件に指定する文章です。概念検索では、種文章を特徴付ける単語が種文章から抽出され、さらに抽出された特徴タームの中から種文章の概念を表す（実際の検索に使用する）タームが選出されます。このタームが、検索タームとして使用されます。

### チェックアウト

バージョン付き文書またはバージョン付きフォルダにバージョンを追加するために、仮のバージョンを追加することです。

### チェックアウト情報

バージョン付きオブジェクトのチェックアウトに関する情報です。チェックアウトしているか、チェックアウトしている場合はだれがチェックアウトしているか、また、チェックアウトしたときに設定された仮のバージョン識別子は何かについての情報です。DbjCheckOutInfo インターフェースで扱います。

### チェックイン

バージョン付き文書またはバージョン付きフォルダをチェックアウトして追加した仮のバージョンを最新バージョンとして確定することです。

### 直接型リンク

一つのフォルダまたはバージョン付きフォルダを親として複数のオブジェクトを関連付けるリンクの種別です。ディレクトリにファイルを格納するイメージでの管理を実現します。

### 定数定義クラス

Java クラスライブラリのメソッドで指定する定数が定義されている DbjDef クラスおよび DbjTraceDef クラスのことです。

### ディレクトリサービス

ネットワーク上にあるユーザや組織の情報などの資源とその属性を記憶し、検索できるようにしたシステムです。DocumentBroker では、Active Directory や Sun Java System Directory Server などの製品を使用した LDAP 対応のディレクトリサービスと連携できます。LDAP 対応のディレクトリサービスとして使用できる製品の詳細については、マニュアル「DocumentBroker Version 3 システム導入・運用ガイド」を参照してください。

### 同義語展開検索

全文検索条件として指定する検索タームまたは検索タームの同義語を含む文書を検索する方法です。例えば、検索ター

ムとして「パソコン」を指定した場合に、「電子計算機」、「パーソナルコンピュータ」、「PC」など、検索タームと同じ意味を持つ単語を含む文書も検索できます。

### 独立データ

独立したデータを表すオブジェクトです。プロパティだけを持つことができる文書空間オブジェクトです。  
edmClass\_IndependentPersistence クラスまたはそのサブクラスを基に作成した DMA オブジェクトをトップオブジェクトとする文書空間オブジェクトです。

### 特権

アクセス制御機能を利用した文書空間で、アクセス権判定を受けることなく、すべてのオブジェクトに自由にアクセスする権利です。セキュリティ定義ファイルにセキュリティ管理者として定義されたユーザに与えられます。特権の有無は、ログイン時にセキュリティ定義ファイルが参照され、ログインユーザごとに作成されるユーザ情報に保持されます。

### トップオブジェクト

文書空間オブジェクトを構成する DMA オブジェクトのうち、代表的な DMA オブジェクトです。例えば、バージョンなし文書の場合は、DocVersion オブジェクトです。  
また、バージョン付きオブジェクトの場合、バージョンングオブジェクトのトップオブジェクトとバージョンオブジェクトのトップオブジェクトがあります。例えば、バージョン付き文書の場合は、ConfigurationHistory オブジェクトと VersionTracedDocVersion オブジェクト（または DocVersion オブジェクト）がトップオブジェクトです。

### トップオブジェクトクラス

トップオブジェクトの基になる DMA クラスです。文書空間オブジェクトのプロパティの定義元になります。

### トランザクション

文書空間オブジェクトの処理単位です。トランザクション単位で、文書空間オブジェクトへの操作を確定または取り消すことができます。  
トランザクションの単位は、ユーザが明示的に指定できます。指定しなかった場合は、メソッド単位でトランザクションが分割されます。

### トレース情報

Java クラスライブラリで発生した障害の原因を追求するための情報です。

## (ナ行)

---

### 名前付き検索結果

列名が付いている検索結果集合です。列名には、edmSQL 文で SELECT 句に指定した項目の名前（プロパティ名など）が設定されます。

### 名前なし検索結果

列名が付いていない検索結果集合です。名前なし検索結果は、あとから列名を追加して、名前付き検索結果集合にすることができます。

## (ハ行)

---

### バージョンングオブジェクト

バージョン付きオブジェクトの一連のバージョンを統括する文書空間オブジェクトです。DMA オブジェクトの ConfigurationHistory オブジェクトをトップオブジェクトとする文書空間オブジェクトです。

### バージョンオブジェクト

バージョン付きオブジェクトの個々のバージョンを表す文書空間オブジェクトです。  
バージョン付き文書の場合は、DMA オブジェクトの VersionTracedDocVersion オブジェクト（または DocVersion オブジェクト）をトップオブジェクトとする文書空間オブジェクトです。バージョン付きフォルダの場合は、DMA オブ

ジェットの ContainerVersion オブジェクトをトップオブジェクトとする文書空間オブジェクトです。

### バージョン管理

文書空間オブジェクトを更新する場合に、古いオブジェクトを残して新しい状態のオブジェクトを追加して、一連の履歴を管理することです。

### バージョン管理権

組み合わせパーミッションの一つです。基本プロパティ参照権、基本プロパティ更新権、基本コンテンツ参照権、基本コンテンツ更新権および基本バージョン管理権を組み合わせたパーミッションです。

### バージョン識別子

複数のバージョンを持つ文書空間オブジェクトの、バージョンを識別するための識別子です。特定のバージョンを指定して操作するときを使用します。この識別子は DocumentBroker によってバージョンごとに設定される識別子であり、バージョンオブジェクトの OIID とは異なります。

### バージョン付きフォルダ

複数のバージョンを保持できるフォルダを表す文書空間オブジェクトです。

### バージョン付き文書

複数のバージョンを保持できる文書を表す文書空間オブジェクトです。

### バージョンなしフォルダ

バージョン管理しないフォルダを表す文書空間オブジェクトです。

### バージョンなし文書

バージョン管理しない文書を表す文書空間オブジェクトです。

### パーミッション

文書空間オブジェクトの作成、プロパティ参照、コンテンツ更新などの実行できる操作の範囲を表す値です。オブジェクト作成権限を与えるパーミッション、オブジェクトの操作の範囲を定めるパーミッションがあります。

### バインド

文書やフォルダからパブリック ACL を参照することです。

### パブリック ACL

文書空間オブジェクトとして存在するアクセス制御情報です。複数の文書空間オブジェクトで共有できます。

### パラメタクラス

Java クラスライブラリ固有のデータをメソッドで受け渡す場合に使用するインターフェースの総称です。

### ファクトリオブジェクト

DbjFactory インターフェースの機能を実行するためのオブジェクトです。パラメタクラスのオブジェクトおよびセッションオブジェクトが作成できます。

### ファクトリクラス

パラメタクラスのオブジェクトの生成、セッションオブジェクトの生成、および文書空間メタ情報アクセスインターフェースの取得を実行する、クラスならびにインターフェースです。

### プライマリグループ

アクセス制御フラグ (ACFlag) でパーミッションを与えるグループです。

### フラッシュ

Proxy オブジェクトのターゲットプロパティ値集合プロパティに設定した値を、文書空間オブジェクトのプロパティに反映することです。

## フルコントロール

組み合わせパーミッションの一つです。すべての基本パーミッションを組み合わせたパーミッションです。文書空間オブジェクトに対するすべての操作を許可します。

## プロパティ

オブジェクトに関する付加情報です。プロパティに設定される値をプロパティ値といいます。

## プロパティ更新権

組み合わせパーミッションの一つです。基本プロパティ更新権と同じ操作を許可するパーミッションです。

## プロパティ参照権

組み合わせパーミッションの一つです。基本プロパティ参照権と同じ操作を許可するパーミッションです。

## プロパティ値集合

プロパティ名とプロパティ値の対応を要素として持つ集合です。DbjPropSet インターフェースで扱います。

DbjPropSet インターフェースは、java.util.Map インターフェースを継承しています。キーをプロパティ名、値をプロパティ値とするマップとして表されます。

Proxy オブジェクトにロードしたプロパティの値を操作したり、文書空間オブジェクトに設定するためのプロパティの値を設定したりするときに使用します。

## プロパティ定義元の DMA クラス

Java クラスライブラリで扱う文書空間オブジェクトのプロパティが実際に定義されている DMA のクラスです。

## プロパティディスクリプション

文書空間オブジェクトのプロパティについてのメタ情報です。プロパティごとに定義されています。

## 文書

dmaClass\_DocVersion クラスまたはそのサブクラスをトップオブジェクトクラスとする文書空間オブジェクトです。コンテンツを保持できます。

## 文書管理クラス

DocumentBroker の文書管理モデルに基づいた管理を実現するための機能を提供するインターフェースの総称です。

## 文書間リンク

文書と文書を関連付けるリンクの種別です。

## 文書空間

DMA オブジェクトモデルを実装するリポジトリです。

## 文書空間アクセスオブジェクト

セッションを確立した文書空間を、概念的なオブジェクトとして扱うためのオブジェクトです。DbjDocSpace インターフェースの機能を実行できます。

## 文書空間オブジェクト

DocumentBroker の文書管理で使用する、文書空間に存在するオブジェクトの総称です。

バージョンなし文書、バージョン付き文書、バージョンなしフォルダ、バージョン付きフォルダ、独立データおよびパブリック ACL があります。

## 文書空間識別子

セッションを確立する文書空間を特定するための識別子です。GUID 文字列で表されます。

## 文書のアップロード情報

文書のコンテンツを更新するときに、コンテンツとして登録するファイルのパス名やファイル名、レンディションタイプなどを設定します。DbjUploadInfo インターフェースで扱います。

## 変換フラグ

マルチレンディション文書のサブレンディションのコンテンツを、レンディション変換の対象にするかどうかを表すフラグです。DocumentBroker Rendering Option を使用してレンディション変換を実行する場合に使用します。また、DocumentBroker Rendering Option によるレンディション変換でエラーが発生した場合には、エラーを示すフラグとしても使われます。dbrProp\_RenditionStatus プロパティの上位 2 バイトに設定されます。

## (マ行)

---

### マスタレンディション

マルチレンディション文書の主要なレンディションです。マルチレンディション文書の参照・更新時には、レンディション形式を指定しますが、レンディション形式を指定しない場合は、マスタレンディションが対象になります。なお、マスタレンディションとして扱うレンディションは、登録後に変更できます。

### マップ

java.util.Map インターフェースの仕様に従うオブジェクトです。キーと値が対応付けられた要素オブジェクトの集合を表します。

### マルチレンディション機能

一つの文書に、同一内容の複数の異なる形式のコンテンツを登録する機能です。

### マルチレンディション文書

複数のレンディションを登録している文書のことです。一つの同じ内容を表す複数の形式のコンテンツを保持する文書です。バージョンなし文書クラスまたはバージョン付き文書クラスを使用して操作します。

### メソッド

データを操作するために定義されている方法です。Java クラスライブラリでは、インターフェースおよびクラスごとにメソッドが定義されています。

### メタクラス

文書空間のメタ情報を扱うためのインターフェースの総称です。

### メッセージ情報

Java クラスライブラリで提供するメソッドを実行してエラーが発生したときに取得できるエラーメッセージです。

## (ヤ行)

---

### ユーザ管理システム

DocumentBroker にログインするユーザのユーザ名や所属グループなどの情報を管理しているシステムです。LDAP 対応のディレクトリサービスなどが使用できます。

なお、ログイン時の認証に必要な情報、およびアクセス制御機能に必要な情報を取得する場合は、DocumentBroker サーバを介してユーザ管理システムにアクセスします。この場合にどのユーザ管理システムと連携するかについては、DocumentBroker サーバで定義されています。

### ユーザ権限

文書空間にオブジェクトを作成する権利（オブジェクト作成権限）と、文書空間内のすべてのオブジェクトに対する操作の範囲（オブジェクト操作権限）をユーザまたはグループ単位で定めるアクセス制御情報の一つです。ユーザ権限定義ファイルに定義します。ユーザ権限の内容は、ログイン時にユーザ権限定義ファイルが参照され、ログインユーザごとに作成されるユーザ情報に保持されます。

### ユーザ権限定義ファイル

ユーザ権限（オブジェクト作成権限およびオブジェクト操作権限）を定義するためのファイルです。

## ユーザ情報

ログインユーザのユーザ識別子、所属グループ、特権およびユーザ権限を表す情報です。ログイン時にユーザごとに生成され、アクセス権判定に使用されます。

## ユーザ定義プロパティ

ユーザが業務に応じて追加するプロパティです。

## (ラ行)

---

### ライブラリ情報取得クラス

DocumentBroker Development Kit および DocumentBroker Runtime のバージョン情報を返却するメソッドを提供するクラスです。

### ランキング検索

全文検索条件に対する適応度をスコアとして算出して、スコアを基に検索結果一覧をソートして出力する検索です。

### リスト

java.util.List インターフェースの仕様に従うオブジェクトです。順序付けのある要素オブジェクトの集合を表します。

### リレーション種別

文書間リンクで関連付けられた文書をたどる場合に、リンク先の文書を取得するか、リンク元の文書を取得するかを指定するための種別です。

### リンク Proxy オブジェクト

リンクオブジェクトを操作する場合に使用する代理オブジェクトです。  
リンク Proxy オブジェクトは、メモリ空間に存在します。不要になった場合は、Java のガベージコレクションによって削除されます。

### リンクオブジェクト

文書空間オブジェクトの関連付けに使用するオブジェクトです。要素は、dmaClass\_Relationship クラスのサブクラスを基に作成された DMA オブジェクトです。文書空間およびデータベースに存在する永続オブジェクトです。

### リンク権

組み合わせパーミッションの一つです。基本リンク権と同じ操作を許可するパーミッションです。

### リンク先オブジェクト

リンクを設定する先になる文書空間オブジェクトです。

### リンク識別子

文書空間オブジェクト間（フォルダとフォルダ間またはフォルダと文書間）のリンクを識別するための識別子です。リンクを解除したり、リンクのプロパティの参照または更新したり、構成管理モードを変更したりする場合に使用します。この識別子は関連付けをしたときに DocumentBroker によって設定される識別子です。同じフォルダに、同じオブジェクトを 2 度リンク付けした場合は、それぞれ異なるリンク識別子が設定されます。

### リンク種別

リンクの種類です。直接型リンク、参照型リンク、構成管理型リンク、文書間リンクがあります。なお、構成管理型リンクは、構成管理モードによって、さらに 2 種類の種別に分けられます。

### リンク設定情報

リンクの設定に関する情報です。リンク種別、リンクオブジェクトのプロパティ、リンク先になるオブジェクトに関する情報などがあります。DbjSetLinkInfo インターフェースのサブインターフェースで表されます。なお、DbjSetLinkInfo インターフェースのサブインターフェースは、リンク種別ごとに存在します。

### リンクプロパティ値集合プロパティ

リンク Proxy オブジェクトのプロパティです。リンクオブジェクトからロードしたプロパティ値集合が設定されます。

### リンク元オブジェクト

リンクを設定する元になる文書空間オブジェクトです。

### 例外クラス

Java クラスライブラリで発生する例外のうち、Java クラスライブラリ固有の例外を扱うクラスの総称です。  
java.lang.Exception クラスまたは java.lang.Error クラスを継承しています。

### 列名

検索結果集合の列に付ける名前です。edmSQL 文の SELECT 句に指定したプロパティ名が設定されます。

### レンディション

文書のコンテンツの形式およびそのコンテンツを併せた概念です。

### レンディション情報

レンディションの情報です。レンディションタイプおよびレンディションのプロパティです。DbjRenditionInfo インターフェースで扱います。

### レンディション情報リスト

レンディション情報を要素としたリストです。DbjRenditionList インターフェースで扱います。

### レンディションタイプ

Word などのアプリケーションで編集したファイル、HTML 形式のファイル、GIF などの画像データのファイルのように、登録した文書のコンテンツのファイルの形式を表す文字列です。レンディションごとに設定します。  
DocumentBroker では、レンディションタイプとして、MIME 形式を指定することを推奨しています。

### レンディション定義ファイル

ファイルの拡張子とレンディションタイプ (MIME 形式) の対応を定義するファイルです。Java クラスライブラリがコンテンツとして登録するファイルの拡張子によってレンディションタイプ (MIME 形式) を自動的に判別して設定する場合に参照されます。

### レンディション変換

マルチレンディション文書のマスタレンディションのコンテンツを変換して、サブレンディションのコンテンツを作成、登録することです。

### ローカル ACL

文書空間オブジェクトごとに設定できるアクセス制御リスト (ACL) です。VARRAY 型のプロパティとして設定されます。

### ロード

文書空間オブジェクトのプロパティを、Proxy オブジェクトのターゲットプロパティ値集合プロパティに読み込むことです。

### ログアウト

文書空間とのセッションを切断することです。

### ログイン

文書空間とのセッションを確立することです。ログインするときには、ユーザ識別子とパスワードによる認証処理も実行されます。

### ロック指定検索

検索結果集合として取得した文書空間オブジェクトに、指定した種別のロックを設定する検索です。



### ロック種別

排他制御を実行するために設定する，ロックの種類です。read ロックと write ロックがあります。

### ロック種別プロパティ

Proxy オブジェクトのプロパティです。その Proxy オブジェクトのインターフェースのメソッドで設定するロック種別が設定されます。

## (ワ行)

---

### ワイルドカード

文字の代わりに指定する記号です。検索する単語の一部しかわからない場合，わかっている部分にワイルドカードを付けて検索条件のキーワードとして指定します。



---

# 索引

## 記号

---

? パラメタ 550  
? パラメタ値の取得 133

## A

---

absolute 157  
ACE 550  
ACE の作成 43  
ACE のプロパティ値集合の取得 74  
ACE のプロパティ値集合の設定 78  
ACFlag 550  
ACL 550  
aclMode ( changeSearchACLMode ) 243  
addPropSet 219  
addPropVals 220  
addRendition 294  
afterLast 158  
AND 条件 550  
API ( Application Programming Interface ) 550  
arg 495

## B

---

beforeFirst 159  
begin 370  
bindObjectList ( createPublicACL ) 255  
bindPublicACL 296  
BOOL 型の ? パラメタの作成 44  
BOOL 型を表す定数 469

## C

---

cacheKey ( createFetchInfo ) 45  
cacheKey ( setCacheKey ) 100  
cacheKey プロパティ 91  
cacheName ( createFetchInfo ) 45  
cacheName ( setCacheName ) 101  
cacheName プロパティ 91  
cacheTotal ( setCacheTotal ) 102  
cacheTotal プロパティ 91  
call 497  
cancelCheckOut 298  
changeMap ( changePropNames ) 115, 222  
changeMap ( changeToVTFFix ) 302  
changeMasterRendition 300  
changePropName 114, 221  
changePropNames 115, 222

changeSearchACLMode 243  
changeToVTFFix 270, 283, 302  
changeToVTFFloat 271, 284, 304  
checkIn 306  
checkOut 308  
checkOutUserId プロパティ 84  
checkOutVersionId プロパティ 84  
checkOut プロパティ 84  
checkSession 371  
childLinkList ( link ) 337  
childList ( unlink ) 350  
childObjs ( link ) 337  
classList プロパティ 108  
className ( call ) 498  
className ( createDocument ) 244  
className ( createFolder ) 247  
className ( createIndependentData ) 249  
className ( createPublicACL ) 255  
className ( DbjTrace ) 499  
className ( getClassDesc ) 394  
classNameVersion ( createVrDocument ) 257  
classNameVersion ( createVrFolder ) 260  
classNameVersioning ( createVrDocument ) 257  
classNameVersioning ( createVrFolder ) 260  
clearParseMessage 230  
columnIndex ( getColumnMetaName ) 164  
columnIndex ( getColumnName ) 165  
columnIndex ( getColumnTypes ) 166  
columnIndex ( getColumnVals ) 167  
columnIndex ( getIntegerVal ) 168  
columnIndex ( getIntVal ) 169  
columnIndex ( getObjectRef ) 170  
columnIndex ( getObjectVal ) 171  
columnIndex ( getStringVal ) 176  
columnIndex ( getTableName ) 177  
columnIndex ( getVArrayRef ) 178  
columnIndex ( getVArrayVal ) 179  
columnIndex ( isNull ) 185  
columnIndex ( setColumnMetaName ) 190  
columnIndexOfOiid ( createObjList ) 254  
columnMetaName ( setColumnMetaName ) 190  
columnName ( findColumnName ) 161  
columnName ( getColumnVals ) 167  
columnName ( getIntegerVal ) 168  
columnName ( getIntVal ) 169  
columnName ( getObjectRef ) 170  
columnName ( getObjectVal ) 171

columnName ( getStringVal ) 176  
 columnName ( getVArrayRef ) 178  
 columnName ( getVArrayVal ) 179  
 columnName ( isNull ) 185  
 columnNameOfOid ( createObjList ) 254  
 commit 372  
 comp ( createFetchInfo ) 45  
 comp ( setComparator ) 103  
 Comparator インターフェースの取得 96  
 Comparator インターフェースの設定 103  
 comparator プロパティ 92  
 ContainerVersion オブジェクト 550  
 Container オブジェクト 550  
 contentLocation プロパティ  
 ( DbjReferenceContentInfo ) 134  
 contentOperateMode ( createReferencePathInfo )  
 52  
 contentOperateMode ( setContentOperateMode )  
 143  
 contentOperateMode プロパティ  
 ( DbjReferencePathInfo ) 137  
 CORBA ( Common Object Request Broker  
 Architecture ) 550  
 createACE 43  
 createBooleanQParam 44  
 createDocument 244  
 createFetchInfo 45  
 createFolder 247  
 createIndependentData 249  
 createInteger32QParam 47  
 createLinkObjList 251  
 createObjConnection 252  
 createObjList 253  
 createObjQParam 49  
 createOIDQParam 48  
 createPropSet 50  
 createPublicACL 255  
 createPublicACLIdElm 51  
 createReferencePathInfo 52  
 createReferenceUploadInfo 53  
 createSeedDocQParam 54  
 createSession 55  
 createSetDCRLinkInfo 56  
 createSetRCRLinkInfo 57  
 createSetRelLinkInfo 58  
 createSetVTFixLinkInfo 59  
 createSetVTFloatLinkInfo 60  
 createStringQParam 61  
 createUploadInfo 62  
 createVArray 63

createVrDocument 257  
 createVrFolder 260  
 createXmlTranslator 64  
 createXmlUploadInfo 65  
 CURRENT\_VERSION 485

## D

---

DATATYPE\_BOOL 468  
 DATATYPE\_INT 468  
 DATATYPE\_STR 468  
 DATATYPE\_STRLIST 468  
 DATATYPE\_UNKNOWN 468  
 DATATYPE\_VARRAY 468  
 DbjAccessControlException クラス 411  
 DbjAccessControlNotSupportedException クラス  
 412  
 DbjACEOperationException クラス 413  
 DbjACE インターフェース 69  
 DbjACE インターフェースのメソッド一覧 6  
 DbjACLOutOfRangeException クラス 414  
 DbjAlreadyCheckOutException クラス 415  
 DbjBooleanQParam インターフェース 83  
 DbjCheckOutException クラス 416  
 DbjCheckOutInfo インターフェース 84  
 DbjCheckOutInfo インターフェースのメソッド一覧  
 7  
 DbjClassDesc インターフェース 388  
 DbjClassDesc インターフェースのメソッド一覧 21  
 DbjContentInfo インターフェース 88  
 DbjContentInfo インターフェースのメソッド一覧 7  
 DbjContentNotRegisteredException クラス 417  
 DbjContentTypeMismatchException クラス 418  
 DbjCORBAException クラス 419  
 DbjDBDeadLockException クラス 420  
 DbjDBException クラス 421  
 DbjDBLockTimeoutException クラス 422  
 DbjDef クラス 466  
 DbjDisconnectedSessionException クラス 423  
 DbjDocSpace インターフェース 242  
 DbjDocSpace インターフェースのメソッド一覧 15  
 DbjError クラス 424  
 DbjException クラス 425  
 DbjFactory0200 クラス 38  
 DbjFactory0200 クラスのメソッド一覧 2  
 DbjFactory インターフェース 41  
 DbjFactory インターフェースのメソッド一覧 2  
 DbjFetchInfo インターフェース 91  
 DbjFetchInfo インターフェースのメソッド一覧 7  
 DbjFileAccessException クラス 426

- DbjFileNotFoundException クラス 427
- DbjFileReferenceCurrentContentNotfoundException クラス 428
- DbjFileReferenceMismatchStatusException クラス 429
- DbjFileReferenceOperationFailedException クラス 430
- DbjIllegalCacheStartIndexException クラス 431
- DbjIllegalDocSpaceIdException クラス 432
- DbjIllegalObjectTypeException クラス 433
- DbjIllegalPropValException クラス 434
- DbjInitializeError クラス 435
- DbjInteger32QParam インターフェース 107
- DbjInternalError クラス 436
- DbjIOException クラス 437
- DbjIsMasterRenditionException クラス 438
- DbjLastVersionException クラス 439
- DbjLibInfo クラス 490
- DbjLibInfo クラスのメソッド 32
- DbjLinkObjList インターフェース 282
- DbjLinkObjList インターフェースのメソッド一覧 16
- DbjLinkObj インターフェース 269
- DbjLinkObj インターフェースのメソッド一覧 16
- DbjMappedProp インターフェース 108
- DbjMappedProp インターフェースのメソッド一覧 8
- DbjMasterRenditionNotSetException クラス 440
- DbjMetaManager インターフェース 400
- DbjMetaManager インターフェースのメソッド一覧 22
- DbjMeta インターフェース 393
- DbjMeta インターフェースのメソッド一覧 21
- DbjNotAuthenticatedException クラス 441
- DbjNotCheckoutException クラス 442
- DbjNotLoginException クラス 443
- DbjObjectNotFoundException クラス 444
- DbjObjList インターフェース 359
- DbjObjList インターフェースのメソッド一覧 19
- DbjObjQParam インターフェース 111
- DbjObj インターフェース 293
- DbjObj インターフェースのメソッド一覧 17
- DbjOIIDQParam インターフェース 112
- DbjOutOfMemoryError クラス 445
- DbjPropDesc インターフェース 402
- DbjPropDesc インターフェースのメソッド一覧 22
- DbjPropSet インターフェース 113
- DbjPropSet インターフェースのメソッド一覧 8
- DbjPublicACLAlreadyBoundException クラス 446
- DbjPublicACLIdElm インターフェース 127
- DbjPublicACLIdElm インターフェースのメソッド一覧 9
- DbjPublicACLNotBoundException クラス 447
- DbjPublicACLNotFoundExpection クラス 448
- DbjPublicACLOperationException クラス 449
- DbjPublicACLOutOfRangeException クラス 450
- DbjQParam インターフェース 132
- DbjQParam インターフェースのメソッド一覧 9
- DbjReferenceContentInfo インターフェース 134
- DbjReferenceContentInfo インターフェースのメソッド一覧 9
- DbjReferencePathInfo インターフェース 137
- DbjReferencePathInfo インターフェースのメソッド一覧 10
- DbjReferenceTypeMismatchException クラス 451
- DbjReferenceUploadInfo インターフェース 147
- DbjReferenceUploadInfo インターフェースのメソッド一覧 10
- DbjRenditionCountOutOfRangeExpection クラス 452
- DbjRenditionInfo インターフェース 150
- DbjRenditionInfo インターフェースのメソッド一覧 10
- DbjRenditionIsEmptyException クラス 453
- DbjRenditionList インターフェース 153
- DbjRenditionList インターフェースのメソッド一覧 11
- DbjRenditionNotConvertedException クラス 454
- DbjRenditionNotFoundExpection クラス 455
- DbjRenditionTypeDuplicatedException クラス 456
- DbjResultSet インターフェース 156
- DbjResultSet インターフェースのメソッド一覧 11
- DbjSeedDocQParam インターフェース 191
- DbjSessionException クラス 457
- DbjSessionNotConnectException クラス 458
- DbjSessionOverflowException クラス 459
- DbjSession インターフェース 369
- DbjSession インターフェースのメソッド一覧 19
- DbjSetDCRLinkInfo インターフェース 192
- DbjSetLinkInfo インターフェース 193
- DbjSetLinkInfo インターフェースのメソッド一覧 12
- DbjSetRCRLinkInfo インターフェース 199
- DbjSetRelLinkInfo インターフェース 200
- DbjSetVTFixLinkInfo インターフェース 201
- DbjSetVTFFloatLinkInfo インターフェース 202
- DbjSetVTLinkInfo インターフェース 203
- DbjStringQParam インターフェース 204
- DbjSubjectLengthOutOfRangeExpection クラス 460
- DbjTargetContentPathNotSetException クラス 461

DbjTrace 499  
 DbjTraceDef クラス 486  
 DbjTrace クラス 494  
 DbjTrace クラスのメソッド 33, 494  
 DbjUnknownError クラス 462  
 DbjUploadInfo インターフェース 205  
 DbjUploadInfo インターフェースのメソッド一覧 12  
 DbjVArray インターフェース 217  
 DbjVArray インターフェースのメソッド一覧 13  
 DbjVerObjList インターフェース 381  
 DbjVerObjList インターフェースのメソッド一覧 20  
 DbjVerObj インターフェース 379  
 DbjVerObj インターフェースのメソッド一覧 20  
 DbjXmlParseException クラス 463  
 DbjXmlTranslator インターフェース 384  
 DbjXmlTranslator インターフェースのメソッド一覧 20  
 DbjXmlUploadInfo インターフェース 228  
 DbjXmlUploadInfo インターフェースのメソッド一覧 13  
 DEBUG 487  
 DEFAULT\_DOCSPCID 485  
 deleteRendition 310  
 deleteRootPath ( createReferencePathInfo ) 52  
 deleteRootPath ( setDeleteRootPath ) 144  
 deleteRootPath プロパティ  
 ( DbjReferencePathInfo ) 137  
 deleteVersion 311  
 distinct 160  
 DMA ( Document Management Alliance ) 550  
 DMA\_FALSE 469  
 DMA\_TRUE 469  
 DMA\_UNKNOWN 469  
 dmaClass\_ConfigurationHistory クラス 551  
 dmaClass\_Container クラス 551  
 dmaClass\_DocVersion クラス 551  
 DMA オブジェクト 551  
 DMA クラス 551  
 docspaceId ( createSession ) 55  
 docspaceId ( createXmlTranslator ) 64  
 docspaceId ( getMeta ) 401  
 DocVersion オブジェクト 551  
 downloadContents 313

## E

---

edmClass\_ContainerVersion クラス 551  
 edmClass\_IndependentPersistence クラス 551  
 edmClass\_PublicACL クラス 551  
 edmClass\_Relationship クラス 551

edmClass\_VersionTracedDocVersion クラス 551  
 edmSQL 552  
 edmSQL 検索 552  
 enter 500  
 entry ( createReferencePathInfo ) 52  
 entry ( setEntry ) 145  
 entry プロパティ ( DbjReferencePathInfo ) 137  
 eqlStatement ( executeSearch ) 263  
 eqlStatement ( removeObjects ) 268  
 ERROR 487  
 error 502  
 ERRORLOG 488  
 ex ( error ) 503  
 executeSearch 262  
 exit 505  
 extension ( getRenditionType ) 399

## F

---

fetchCount ( createFetchInfo ) 45  
 fetchCount ( setFetchCount ) 104  
 fetchCount プロパティ 92  
 fetchInfo ( executeSearch ) 263  
 fetchInfo ( getBindObjectList ) 315  
 fetchInfo ( getChildList ) 319  
 fetchInfo ( getParentList ) 327  
 fetchInfo ( getRelList ) 330  
 fetchInfo ( getVersionObjList ) 335  
 filePath ( createReferenceUploadInfo ) 53  
 filePath ( createUploadInfo ) 62  
 filePath ( createXmlUploadInfo ) 65  
 filePath ( downloadContents ) 313  
 filePath ( getMappedProperties ) 385  
 filePath ( setFilePath ) 212  
 filePath プロパティ 205, 229  
 filterFilePath ( createXmlUploadInfo ) 65  
 filterFilePath ( setFilterFilePath ) 236  
 filterFilePath プロパティ 228  
 findColumnName 161  
 first 162  
 fixVersionId ( changeToVTFix ) 270, 302

## G

---

getBindObjectList 315  
 getCacheKey 93  
 getCacheName 94  
 getCacheTotal 95  
 getCheckoutStatus 317  
 getCheckoutUserId 85  
 getCheckoutVersionId 86

getChildList 318  
 getClassDesc 394  
 getClassList 109  
 className 321  
 getColumnCount 163  
 getColumnMetaName 164  
 columnName 165  
 getColumnType 166  
 getColumnVals 167  
 getComparator 96  
 getContentLocation 135  
 getContentOperateMode 139  
 getDataType 403  
 getDCRParent 322  
 getDeleteRootPath 140  
 getDocSpaceId 395  
 getEntry 141  
 getExtFromRenditionType 396  
 getFactory 39  
 getFetchCount 97  
 getFilePath 207  
 getFilterFilePath 231  
 getId 128  
 getIndexPath 208  
 getIndexType 232  
 getIntegerVal 116, 168  
 getIntVal 117, 169  
 getLinkId 272  
 getLinkIdList 285  
 getLinkObj 286  
 getLinkType 194, 273  
 getListRef 118  
 getLockType 323  
 getLoginUserInfo 373  
 getMappedProperties 385  
 getMappingId 233  
 getMaxFetchCount 98  
 getMeta 265, 401  
 getMetaManager 40  
 getName 389, 404  
 getObj 360  
 getObjectRef 170  
 getObjectVal 171  
 getObjType 324  
 getOiid 325  
 getOwnerObj 274  
 getOwnerObjList 287  
 getParentList 326  
 getParseLevel 234  
 getParseMessage 235  
 getPermission 70  
 getPropCount 223  
 getPropDataType 397  
 getPropDesc 398  
 getProperties 390  
 getPropNameSet 224  
 getPropSet 172  
 getPropVals 225  
 getPublicACLList 328  
 getReferencePath 374  
 getReferencePathInfo 148  
 getReferenceType 136  
 getRelList 329  
 getRenditionInfo 154  
 getRenditionList 331  
 getRenditionType 89, 151, 209, 399  
 getRenditionTypeList 155  
 getRetrievalName 90, 210  
 getRow 173  
 getRowCount 174  
 getRowVals 175  
 getSearchACLMode 266  
 getStartIndex 99  
 getStringVal 119, 176  
 getSubClasses 391  
 getSubject 72  
 getSubjectType 73  
 getSuperClass 392  
 getTableName 177  
 getTargetObj 195, 275  
 getTargetObjList 288  
 getTargetPath 142  
 getTargetVersion 332  
 getter メソッド 552  
 getVal 133  
 getVArrayClass 405  
 getVArrayRef 120, 178  
 getVArrayVal 121, 179  
 getVerObj 382  
 getVersion 491  
 getVersionId 333, 380  
 getVersionIdList 383  
 getVersioningInfo 334  
 getVersionObjList 335  
 grpSubject ( setGroupSubject ) 75  
 GUID 552

## H

---

HINT 487

hint 506

## I

---

id ( setId ) 130

Id プロパティ 127

index ( getLinkObj ) 286

index ( getObj ) 360

index ( getRenditionInfo ) 154

index ( getVerObj ) 382

indexPath ( createreferenceuploadinfo ) 53

indexPath ( createUploadInfo ) 62

indexPath ( setIndexPath ) 213

INDEXPATH\_SAME 485

indexPath プロパティ 205, 229

indexType ( createXmlUploadInfo ) 65

indexType ( setIndexType ) 237

INDEXTYPE\_NOTHING 470

INDEXTYPE\_PLAIN 470

INDEXTYPE\_STRUCTURED 470

indexType プロパティ 228

init 508

INITIAL\_KEY 485

Integer 型でのプロパティ値の取得 116

Integer 型での列データの取得 168

int 型でのプロパティ値の取得 117

int 型での列データの取得 169

INT 型の ? パラメタの作成 47

isAccessControlMode 267

isAfterLast 180

isAlsoChildren ( removeObject ) 344

isAlsoChildren ( removeObjectes ) 365

isBeforeFirst 181

isCheckOut 87

isFirst 182

isLast 183

isNamed 184

isNull 122, 185

## J

---

Java クラスライブラリ 552

Java クラスライブラリのセッションオブジェクトの  
作成 55

## K

---

KMBJ00001-E 518

KMBJ00002-E 518

KMBJ00003-E 518

KMBJ00004-E 518

KMBJ00005-E 519

KMBJ00006-E 519

KMBJ00007-E 519

KMBJ00008-E 519

KMBJ00009-E 519

KMBJ00010-E 520

KMBJ00011-E 520

KMBJ00012-E 520

KMBJ00013-E 520

KMBJ00014-E 521

KMBJ00015-E 521

KMBJ00016-E 521

KMBJ00017-E 521

KMBJ00018-E 521

KMBJ00019-E 522

KMBJ00020-E 522

KMBJ00021-E 522

KMBJ00022-E 522

KMBJ00023-E 523

KMBJ00024-E 523

KMBJ00025-E 523

KMBJ00026-E 523

KMBJ00027-E 523

KMBJ00028-E 524

KMBJ00029-E 524

KMBJ00030-E 524

KMBJ00031-E 524

KMBJ00032-E 524

KMBJ00033-E 525

KMBJ00034-E 525

KMBJ00035-E 525

KMBJ00036-E 525

KMBJ00037-E 525

KMBJ00038-E 526

KMBJ00039-E 526

KMBJ00040-E 526

KMBJ00041-E 526

KMBJ00042-E 527

KMBJ00043-E 527

KMBJ00044-E 527

KMBJ00045-E 527

KMBJ00046-E 527

KMBJ00047-E 528

KMBJ00048-E 528

KMBJ00049-E 528

KMBJ00050-E 528

KMBJ00051-E 528

KMBJ00052-E 529

KMBJ00053-E 529

KMBJ00054-E 529



KMBJ00055-E 529  
KMBJ00056-E 529  
KMBJ00057-E 530  
KMBJ00058-E 530  
KMBJ00059-E 530  
KMBJ00060-E 530  
KMBJ00061-E 531  
KMBJ00062-E 531  
KMBJ00063-E 531  
KMBJ00064-E 531  
KMBJ00065-E 531  
KMBJ00066-E 532  
KMBJ00067-E 532  
KMBJ00068-E 532  
KMBJ00069-E 532  
KMBJ00070-E 533  
KMBJ00071-W 533  
KMBJ00072-E 533  
KMBJ00073-E 533  
KMBJ00074-E 533  
KMBJ00075-E 534  
KMBJ00076-E 534  
KMBJ00077-E 534  
KMBJ00078-E 534  
KMBJ00079-E 535  
KMBJ00080-E 535  
KMBJ00081-E 535  
KMBJ00082-E 535  
KMBJ00083-E 536  
KMBJ00084-E 536  
KMBJ00085-E 536  
KMBJ00086-E 536  
KMBJ00087-E 536  
KMBJ00088-E 537  
KMBJ00089-W 537  
KMBJ00090-W 537  
KMBJ00091-W 537  
KMBJ00092-E 537  
KMBJ00093-E 538  
KMBJ00094-E 538  
KMBJ00095-E 538  
KMBJ00096-E 538  
KMBJ00097-E 538  
KMBJ00098-E 539  
KMBJ00099-E 539  
KMBJ00100-E 539  
KMBJ00101-E 539  
KMBJ00102-E 540  
KMBJ00103-E 540  
KMBJ00104-E 540  
KMBJ00105-E 540  
KMBJ00106-E 540  
KMBJ00107-E 541  
KMBJ00108-E 541  
KMBJ00109-E 541  
KMBJ00110-E 541  
KMBJ00111-E 542  
KMBJ00112-E 542  
KMBJ00113-E 542  
KMBJ00114-E 542  
KMBJ00115-E 542  
KMBJ00116-W 543  
KMBJ00117-E 543  
KMBJ00118-W 543  
KMBJ00119-E 543  
KMBJ00120-E 543  
KMBJ00121-E 544  
KMBJ00122-E 544  
KMBJ00123-E 544  
KMBJ00124-E 544  
KMBJ00125-E 545  
KMBJ00126-E 545  
KMBJ00127-E 545  
KMBJ00128-E 545  
KMBJ00129-E 545  
KMBJ00132-E 546  
KMBJ00133-E 546  
KMBJ00134-E 546

---

**L**

last 186  
level ( arg ) 496  
level ( call ) 498  
level ( enter ) 500  
level ( error ) 503  
level ( hint ) 506  
level ( msg ) 511  
link 337  
LINK\_ALL 471  
LINK\_DCR 471  
LINK\_NONE 471  
LINK\_RCR 471  
LINK\_REL 471  
LINK\_VT 471  
LINK\_VTFIX 471  
LINK\_VTFLOAT 471  
linkId ( changeToVTFix ) 302  
linkId ( changeToVTFloat ) 304  
linkIdList ( changeToVTFloat ) 304

linkIds ( unlinkByLinkId ) 352  
 linkPropDefs ( getChildList ) 318  
 linkPropDefs ( getParentList ) 326  
 linkPropDefs ( getRelList ) 329  
 linkProps ( createSetDCRLinkInfo ) 56  
 linkProps ( createSetRCRLinkInfo ) 57  
 linkProps ( createSetRelLinkInfo ) 58  
 linkProps ( createSetVTFixLinkInfo ) 59  
 linkProps ( createSetVTFloatLinkInfo ) 60  
 linkType ( getChildList ) 318  
 linkType ( getParentList ) 326  
 linkType ( link ) 337  
 linkType プロパティ 193  
 List 型でのプロパティ値の取得 118  
 lock 339, 361  
 LOCK\_NONE 472  
 LOCK\_READ 472  
 LOCK\_READFORUPDATE 472  
 LOCK\_WRITE 472  
 lockType ( executeSearch ) 263  
 lockType ( lock ) 339, 361  
 login 375  
 logout 376

## M

---

MANAGE 487  
 mappingId ( createXmlUploadInfo ) 65  
 mappingId ( getMappedProperties ) 385  
 mappingId ( setMappingId ) 238  
 mappingId プロパティ 229  
 MAX\_NUM 485  
 maxFetchCount ( createFetchInfo ) 45  
 maxFetchCount ( setMaxFetchCount ) 105  
 maxFetchCount プロパティ 92  
 message ( error ) 503  
 message ( hint ) 506  
 message ( msg ) 511  
 messageID ( msg ) 511  
 methodName ( call ) 498  
 methodName ( DbjTrace ) 499  
 MIME 形式 552  
 move 340, 362  
 msg 510

## N

---

name ( arg ) 496  
 newName ( changePropName ) 114, 221  
 next 187  
 NOT 条件 552

NULL 値かどうかの判定 185  
 NULL 値プロパティの設定 123

## O

---

obj ( createSetDCRLinkInfo ) 56  
 obj ( createSetRCRLinkInfo ) 57  
 obj ( createSetRelLinkInfo ) 58  
 obj ( createSetVTFixLinkInfo ) 59  
 obj ( createSetVTFloatLinkInfo ) 60  
 obj ( setTargetObj ) 198  
 Object 型の ? パラメタの作成 49  
 objType ( getBindObjectList ) 315  
 objType ( getChildList ) 319  
 objType ( getParentList ) 326  
 objType ( getRelList ) 329  
 OBJTYPE\_ANY 473  
 OBJTYPE\_DOC 473  
 OBJTYPE\_FOLDER 473  
 OBJTYPE\_IP 473  
 OBJTYPE\_NVTFFOLDER 473  
 OBJTYPE\_PUBLICACL 473  
 OBJTYPE\_UNKNOWN 473  
 OBJTYPE\_VRDOC 473  
 OBJTYPE\_VRFOLDER 473  
 OIID 552  
 oiid ( createObjConnection ) 252  
 OIID の ? パラメタの作成 48  
 oldName ( changePropName ) 114, 221  
 OPERATEMODE\_NONE 474  
 OPERATEMODE\_USER\_RELATIVE\_CONTENT  
 474  
 order ( getVersionObjList ) 335  
 ORDER\_ASC 475  
 ORDER\_DESC 475  
 ORDER\_NONE 475  
 OR 条件 552  
 output ( arg ) 496  
 output ( call ) 498  
 output ( enter ) 500  
 output ( error ) 503  
 output ( hint ) 506  
 output ( msg ) 511

## P

---

param ( createBooleanQParam ) 44  
 param ( createInteger32QParam ) 47  
 param ( createObjQParam ) 49  
 param ( createOIIDQParam ) 48  
 param ( createSeedDocQParam ) 54

- param ( createStringQParam ) 61
- parentFolder ( move ) 340, 362
- parentLinkList ( createDocument ) 244
- parentLinkList ( createFolder ) 247
- parentLinkList ( createVrDocument ) 258
- parentLinkList ( createVrFolder ) 260
- parseLevel ( createXmlUploadInfo ) 65
- parseLevel ( getMappedProperties ) 385
- parseLevel ( setParseLevel ) 239
- parseLevel プロパティ 229
- parseMessage プロパティ 229
- passWord ( login ) 375
- pathInfo ( downloadcontents ) 313
- pathInfo ( removeObject ) 344
- PERM\_CHANGE\_PERM 476
- PERM\_CREATE 476
- PERM\_DELETE 476
- PERM\_FULL\_CONTROL 477
- PERM\_LINK 476
- PERM\_NONE 476
- PERM\_PRIM\_DELETE 476
- PERM\_PRIM\_LINK 476
- PERM\_PRIM\_READ\_CONTENTS 476
- PERM\_PRIM\_READ\_PROPS 476
- PERM\_PRIM\_VERSION 476
- PERM\_PRIM\_WRITE\_CONTENTS 476
- PERM\_PRIM\_WRITE\_PROPS 476
- PERM\_READ 476
- PERM\_READ\_PROPS 476
- PERM\_READ\_WRITE 476
- PERM\_VERSION 476
- PERM\_WRITE\_PROPS 476
- permission ( createACE ) 43
- permission ( setPermission ) 76
- permission プロパティ 69
- previous 188
- PRIV\_NONE 478
- PRIV\_SECURITY\_ADMINISTRATOR 478
- PROMPT 488
- propDefs ( cancelCheckOut ) 298
- propDefs ( checkIn ) 306
- propDefs ( checkOut ) 308
- propDefs ( createVArray ) 63
- propDefs ( DbjLinkObj#readProperties ) 277
- propDefs ( DbjLinkObj#writeProperties ) 280
- propDefs ( DbjLinkObjList#readProperties ) 289
- propDefs ( DbjLinkObjList#writeProperties ) 291
- propDefs ( DbjObj#readProperties ) 342
- propDefs ( DbjObj#writeProperties ) 355
- propDefs ( DbjObjList#readProperties ) 363
- propDefs ( DbjObjList#writeProperties ) 367
- propDefs ( getBindObjectList ) 315
- propDefs ( getChildList ) 318
- propDefs ( getDCRParent ) 322
- propDefs ( getLoginUserInfo ) 373
- propDefs ( getParentList ) 326
- propDefs ( getPublicACLList ) 328
- propDefs ( getRelList ) 329
- propDefs ( getRenditionList ) 331
- propDefs ( getVersionObjList ) 335
- propName ( addPropVals ) 220
- propName ( getIntegerVal ) 116
- propName ( getIntVal ) 117
- propName ( getListRef ) 118
- propName ( getPropDataType ) 397
- propName ( getPropDesc ) 398
- propName ( getPropVals ) 225
- propName ( getStringVal ) 119
- propName ( getVArrayRef ) 120
- propName ( getVArrayVal ) 121
- propName ( isNull ) 122
- propName ( removeProp ) 227
- propName ( setNull ) 123
- propName ( setPropRef ) 124
- propName ( setPropVal ) 125
- propSet 74, 110, 129, 152, 196, 226, 276, 341
- propSet ( addPropSet ) 219
- propSet ( cancelCheckOut ) 298
- propSet ( checkIn ) 306
- propSet ( checkOut ) 308
- propSet ( createDocument ) 244
- propSet ( createFolder ) 247
- propSet ( createIndependentData ) 249
- propSet ( createPublicACL ) 255
- propSet ( createVrDocument ) 257
- propSet ( createVrFolder ) 260
- propSet ( DbjACE#setPropSet ) 78
- propSet ( DbjLinkObj#setPropSet ) 279
- propSet ( DbjLinkObj#writeProperties ) 280
- propSet ( DbjLinkObjList#writeProperties ) 291
- propSet ( DbjObj#setPropSet ) 346
- propSet ( DbjObj#writeProperties ) 355
- propSet ( DbjObjList#setPropSet ) 366
- propSet ( DbjObjList#writeProperties ) 367
- propSet ( DbjPublicACLIdElm#setPropSet ) 131
- propSet ( DbjSetLinkInfo#setPropSet ) 197
- propSet ( writeRenditionProperties ) 357
- propSet プロパティ ( DbjACE ) 69
- propSet プロパティ ( DbjMappedProp ) 108

propSet プロパティ ( DbjPublicACLIdElm ) 127  
 propSet プロパティ ( DbjRenditionInfo ) 150  
 propSet プロパティ ( DbjSetLinkInfo ) 193  
 Proxy オブジェクト 552  
 Proxy オブジェクトのアクセスロック種別の取得 323  
 Proxy オブジェクトのプロパティ値集合インターフェースの取得 341  
 Proxy オブジェクトのプロパティ値集合の設定 346  
 publicACLId ( createPublicACLIdElm ) 51  
 publicACLIdList ( bindPublicACL ) 296  
 publicACLIdList ( unbindPublicACL ) 348

## Q

---

qParamList ( executeSearch ) 263  
 qParamList ( removeObjects ) 268

## R

---

readProperties 277, 289, 342, 363  
 reduct 189  
 ref ( setPropRef ) 124  
 referencePath ( setReferencePath ) 378  
 referencePathInfo ( createReferenceUploadInfo ) 53  
 referencePathInfo ( setReferencePathInfo ) 149  
 referencePathInfo プロパティ ( DbjReferenceUploadInfo ) 147  
 REFERENCETYPE\_NONE 479  
 REFERENCETYPE\_USER\_RELATIVE\_CONTENT 479  
 referenceType プロパティ ( DbjReferenceContentInfo ) 134  
 RELATIONEND\_HEAD 480  
 RELATIONEND\_STATUS\_ALL 480  
 RELATIONEND\_STATUS\_EXIST 480  
 RELATIONEND\_STATUS\_NOT\_EXIST 480  
 RELATIONEND\_TAIL 480  
 relationendType 329  
 removeObject 278, 344  
 removeObjects 268, 290, 365  
 removeProp 227  
 renditionPropSet 211  
 renditionPropSet ( createReferenceUploadInfo ) 53  
 renditionPropSet ( createUploadInfo ) 62  
 renditionPropSet ( createXmlUploadInfo ) 65  
 renditionPropSet ( setRenditionPropSet ) 214  
 renditionPropSet プロパティ 205, 229  
 renditionType ( changeMasterRendition ) 300  
 renditionType ( createReferenceUploadInfo ) 53  
 renditionType ( createUploadInfo ) 62

renditionType ( createXmlUploadInfo ) 65  
 renditionType ( downloadContents ) 313  
 renditionType ( getExtFromRenditionType ) 396  
 renditionType ( setRenditionType ) 215  
 renditionType ( uploadContents ) 353  
 renditionType ( writeRenditionProperties ) 357  
 renditionTypeList ( deleteRendition ) 310  
 renditionType プロパティ ( DbjContentInfo ) 88  
 renditionType プロパティ ( DbjRenditionInfo ) 150  
 renditionType プロパティ ( DbjUploadInfo ) 206  
 renditionType プロパティ ( DbjXmlUploadInfo ) 229  
 RENDSTATUS\_CONVERT\_ERROR 481  
 RENDSTATUS\_CONVERT\_NOTREQUIRED 481  
 RENDSTATUS\_CONVERT\_REQUIRED 481  
 RENDSTATUS\_MASTERD 481  
 RENDSTATUS\_MASTERREND\_UPDATE 481  
 RENDSTATUS\_NO\_SUBREND 481  
 RENDSTATUS\_SUBREND\_EXIST 481  
 result ( createObjList ) 254  
 retrievalName ( createReferenceUploadInfo ) 53  
 retrievalName ( createUploadInfo ) 62  
 retrievalName ( createXmlUploadInfo ) 65  
 retrievalName ( setRetrievalName ) 216  
 retrievalName プロパティ ( DbjContentInfo ) 88  
 retrievalName プロパティ ( DbjUploadInfo ) 206  
 retrievalName プロパティ ( DbjXmlUploadInfo ) 229  
 returned 513  
 rollback 377  
 rowIndex ( absolute ) 157  
 rowIndex ( getPropSet ) 172  
 rowIndex ( getRowVals ) 175  
 rowIndex ( propSet ) 226

## S

---

selectItems ( executeSearch ) 262  
 setCacheKey 100  
 setCacheName 101  
 setCacheTotal 102  
 setColumnMetaName 190  
 setComparator 103  
 setContentOperateMode 143  
 setDeleteRootPath 144  
 setEntry 145  
 setFetchCount 104  
 setFilePath 212  
 setFilterFilePath 236  
 setGroupSubject 75

setId 130  
 setIndexPath 213  
 setIndexType 237  
 setMappingId 238  
 setMaxFetchCount 105  
 setNull 123  
 setParseLevel 239  
 setPermission 76  
 setPropRef 124  
 setPropSet 78, 131, 197, 279, 346, 366  
 setPropVal 125  
 setReferencePath 378  
 setReferencePathInfo 149  
 setRenditionPropSet 214  
 setRenditionType 215  
 setRetrievalName 216  
 setStartIndex 106  
 setSubject 79  
 setSubjectType 80  
 setSystemSubject 81  
 setTargetObj 198  
 setTargetPath 146  
 setTargetVersion 347  
 setter メソッド 552  
 setUserSubject 82  
 src ( createACE ) 43  
 src ( createPropSet ) 50  
 src ( createPublicACLIdElm ) 51  
 srcObjs ( createLinkObjList ) 251  
 srcObjs ( createObjList ) 253  
 startIndex ( createFetchInfo ) 45  
 startIndex ( setStartIndex ) 106  
 startIndex プロパティ 92  
 String 型でのプロパティ値の取得 119  
 String 型での列データの取得 176  
 String 型の ? パラメタの作成 61  
 subject ( createACE ) 43  
 subject ( setSubject ) 79  
 subjectType ( createACE ) 43  
 subjectType ( setSubjectType ) 80  
 SUBJECTTYPE\_GRP 482  
 SUBJECTTYPE\_SYS 482  
 SUBJECTTYPE\_USR 482  
 subjectType プロパティ 69  
 subject プロパティ 69  
 sysSubject ( setSystemSubject ) 81  
 SYSSUBJECT\_EVERYONE 483  
 SYSSUBJECT\_SELF 483  
 SYSTEM 488

---

**T**


---

tableName ( createObjList ) 254  
 tableName ( getPropSet ) 172  
 targetObj プロパティ 193  
 targetPath ( createReferencePathInfo ) 52  
 targetPath ( setTargetPath ) 146  
 targetPath プロパティ ( DbjReferencePathInfo ) 137  
 TRACE 488  
 tracePrefix ( init ) 509

---

**U**


---

unbindPublicACL 348  
 unlink 350  
 unlinkByLinkId 352  
 uploadContents 353  
 uploadInfo ( uploadContents ) 353  
 uploadList ( addRendition ) 294  
 uploadList ( createDocument ) 244  
 uploadList ( createVrDocument ) 258  
 upName ( DbjTrace ) 499  
 upName ( init ) 509  
 userName ( login ) 375  
 usrSubject ( setUserSubject ) 82

---

**V**


---

val ( setPropVal ) 125  
 vals ( addPropVals ) 220  
 value ( arg ) 496  
 VARRAY 型 552  
 VARRAY 型でのデータの参照の取得 178  
 VARRAY 型でのデータの取得 179  
 VARRAY 型でのプロパティ値の参照の取得 120  
 VARRAY 型でのプロパティ値の取得 121  
 VARRAY 型プロパティを扱うクラスのクラスディスクリプションの取得 405  
 versionId ( setTargetVersion ) 347  
 versionIds ( deleteVersion ) 311  
 VersionTracedDocVersion オブジェクト 552

---

**W**


---

W3C 553  
 WITH\_ACL 467  
 WITHOUT\_ACL 467  
 writeProperties 280, 291, 355, 367  
 writeRenditionProperties 357

## X

XMLPARSE\_NO\_EXTERNAL\_ENTITIES 484  
 XMLPARSE\_VALIDATE 484  
 XMLPARSE\_WITH\_EXTERNAL\_ENTITIES 484  
 XML インデクスデータ作成機能 553  
 XML 構造指定検索 553  
 XML トランスレータオブジェクト 553  
 XML トランスレータの取得 64  
 XML プロパティマッピング機能 553  
 XML プロパティマッピング結果情報 553  
 XML 文書の XML プロパティマッピング情報の作成  
 385  
 XML 文書のアップロード情報 553  
 XML 文書のアップロード情報オブジェクトの作成  
 65  
 XML 文書の構文解析レベルを表す定数 484  
 xmsFileName ( createXmlTranslator ) 64

## あ

アクセス権 553  
 アクセス指定子 486, 466  
 アクセス制御エレメント ( ACE:Access Control  
 Element ) 553  
 アクセス制御機能 553  
 アクセス制御機能付き検索 553  
 アクセス制御情報 553  
 アクセス制御情報変更権 553  
 アクセス制御フラグ ( ACFlag:Access Control Flag )  
 554  
 アクセス制御モードの取得 267  
 アクセス制御リスト ( ACL:Access Control List ) 554  
 アクセス対象ターゲットバージョンのバージョン識別  
 子の取得 332  
 アクセス対象ターゲットバージョンのバージョン識別  
 子の変更 347  
 アクセス対象文書空間オブジェクトの OIID の取得  
 325  
 アクセス対象文書空間オブジェクトのオブジェクト種  
 別の取得 324  
 アクセス対象文書空間オブジェクトを構成する DMA  
 クラス名の取得 321  
 アクセスロック種別の異なる複数文書空間オブジェク  
 トインターフェースの取得 361  
 アクセスロック種別の異なる文書空間オブジェクトイ  
 ンターフェースの取得 339  
 アンバインド 554

## い

異表記展開検索 554  
 インターフェース 554

## え

永続オブジェクト 554  
 永続プロパティ 554  
 エラー情報の出力 502

## お

オーナーオブジェクトプロパティ 554  
 オブジェクト作成権 554  
 オブジェクト作成権限 554  
 オブジェクト種別 554  
 オブジェクト種別プロパティ 554  
 オブジェクト種別を表す定数 473  
 オブジェクト操作権限 554  
 オブジェクトリファレンス 555

## か

カーソルが最終行かどうかの判定 183  
 カーソルが最終行の後ろかどうかの判定 180  
 カーソルが先頭行かどうかの判定 182  
 カーソルが先頭行の前かどうかの判定 181  
 カーソル行インデクスの取得 173  
 カーソルの最終行の後ろへの移動 158  
 カーソルの最終行への移動 186  
 カーソルの絶対指定行への移動 157  
 カーソルの先頭行の前への移動 159  
 カーソルの先頭行への移動 162  
 カーソルの次の行への移動 187  
 カーソルの前の行への移動 188  
 下位オブジェクト 555  
 概念検索 555  
 下位のメソッドのエラー情報の出力 506  
 外部 API からのリターン情報の出力 513  
 外部 API の呼び出し情報の出力 497  
 拡張子に対応するレンディションタイプの取得 399  
 可変長配列 555  
 可変長配列オブジェクトの作成 63  
 可変長配列オブジェクトのプロパティ値のリストの取  
 得 225  
 可変長配列オブジェクトのプロパティ名の一括変更  
 222  
 可変長配列オブジェクトのプロパティ名の変更 221  
 仮のバージョン識別子 555  
 仮のバージョン識別子の取得 86  
 カレントバージョン 555

## き

---

基本コンテンツ更新権 555  
 基本コンテンツ参照権 555  
 基本削除権 555  
 基本バージョン管理権 555  
 基本パーミッション 555  
 基本プロパティ更新権 555  
 基本プロパティ参照権 556  
 基本リンク権 556  
 キャッシュキー 556  
 キャッシュキーの取得 93  
 キャッシュキーの設定 100  
 キャッシュ検索 556  
 キャッシュ名 556  
 キャッシュ名の取得 94  
 キャッシュ名の設定 101  
 行数の取得 174  
 行データの取得 175  
 行データをプロパティ値集合として取得 172  
 近傍条件検索 556

## く

---

組み合わせパーミッション 556  
 クライアント共有トレースファイル 516  
 クラス、インターフェースおよびメソッドの説明形式 34  
 クラス定義情報ファイル 556  
 クラスディスクリプション 556  
 クラス名の取得 389

## け

---

継承 556  
 検索結果キャッシュ 556  
 検索結果キャッシュの全件数の取得 95  
 検索結果キャッシュの全件数の設定 102  
 検索結果集合 557  
 検索結果取得情報 557  
 検索結果取得情報オブジェクトの作成 45  
 検索結果の最大取得件数の取得 98  
 検索結果の最大取得件数の設定 105  
 検索結果の取得開始位置の取得 99  
 検索結果の取得開始位置の設定 106  
 検索結果の取得件数の取得 97  
 検索結果の取得件数の設定 104  
 検索実行時のアクセス制御モードの取得 266  
 検索実行時のアクセス制御モードの変更 243  
 検索実行時のアクセス制御モードを表す定数 467  
 検索条件を指定したオブジェクトの削除 268

検索の実行 262

## こ

---

更新系メソッド 557  
 構成管理型リンク 557  
 構成管理型リンクの構成管理モードを FIX モードに変更 270, 302  
 構成管理型リンクの構成管理モードを FLOATING モードに変更 271, 304  
 構成管理モードが FIX モードの構成管理型リンク設定情報の作成 59  
 構成管理モードが FLOATING モードの構成管理型リンク設定情報の作成 60  
 構造指定検索 557  
 構文解析結果メッセージのクリア 230  
 構文解析結果メッセージの取得 235  
 構文解析レベルの取得 234  
 構文解析レベルの設定 239  
 コレクション 557  
 コンストラクタ 499  
 コンテンツ 557  
 コンテンツ格納先パスの取得 142  
 コンテンツ格納先パスの設定 146  
 コンテンツ格納先ベースパスの取得 374  
 コンテンツ格納先ベースパスの設定 378  
 コンテンツ情報 557  
 コンテンツのパス操作モード 474  
 コンテンツのパス操作モードの取得 139  
 コンテンツのパス操作モードの設定 143  
 コンテンツのファイル名の取得 90  
 コンテンツのレンディションタイプの取得 89  
 コンテンツロケーションの取得 135

## さ

---

削除権 557  
 削除するディレクトリのルートパスの取得 140  
 削除するディレクトリのルートパスの設定 144  
 サブインターフェース 557  
 サブクラス 557  
 サブクラスのクラスディスクリプションの取得 391  
 サブジェクト 557  
 サブジェクト種別 557  
 サブジェクト種別の取得 73  
 サブジェクト種別の設定 80  
 サブジェクト種別を表す定数 482  
 サブジェクトの取得 72  
 サブジェクトの設定 79  
 サブジェクトをグループサブジェクトとして設定 75  
 サブジェクトをシステムサブジェクトとして設定 81

サブジェクトをユーザサブジェクトとして設定 82  
 サブレンディション 558  
 参照型リンク 558  
 参照型リンク設定情報の作成 57  
 参照系メソッド 558  
 参照権 558  
 参照更新権 558  
 サンプル Web アプリケーション 558

## し

---

システム管理者 558  
 システムサブジェクトを表す定数 483  
 指定したクラスのクラスディスクリプションの取得 394  
 指定したプロパティのデータ型の取得 397  
 指定したプロパティのプロパティディスクリプションの取得 398  
 上位オブジェクト 558  
 状態フラグ 558  
 所有者 558

## す

---

スーパーインターフェース 558  
 スーパークラス 558  
 スーパークラスのクラスディスクリプションの取得 392

## せ

---

セキュリティ ACL 558  
 セキュリティ運用者 559  
 セキュリティ管理者 559  
 セキュリティ定義ファイル 559  
 セッション 559  
 セッションオブジェクト 559  
 セッションが有効かどうかのチェック 371  
 セット 559  
 全文検索 559  
 全文検索インデクス 559  
 全文検索インデクス作成用ファイルのフルパスの取得 208  
 全文検索インデクス作成用ファイルのフルパスの設定 213  
 全文検索インデクスの種別の取得 232  
 全文検索インデクスの種別の設定 237  
 全文検索インデクスの種別を表す定数 470  
 全文検索機能付き文書クラス 559

## そ

---

属性検索 559

## た

---

ターゲット OIID プロパティ 559  
 ターゲットオブジェクト 559  
 ターゲットオブジェクトプロパティ 559  
 ターゲットバージョン 560  
 ターゲットバージョン識別子プロパティ 560  
 ターゲットプロパティ値集合プロパティ 560  
 ターゲットリンクオブジェクト 560  
 ターゲットリンク識別子プロパティ 560  
 ターゲットリンク識別子プロパティのリストの取得 285  
 種文章 560  
 種文章の ? パラメタの作成 54

## ち

---

チェックアウト 560  
 チェックアウトしたユーザのユーザ識別子の取得 85  
 チェックアウト情報 560  
 チェックアウト中かどうかの判定 87  
 チェックイン 560  
 直接型リンク 560  
 直接型リンクが設定されている文書空間オブジェクトの一括移動 362  
 直接型リンクが設定されている文書空間オブジェクトの移動 340  
 直接型リンク設定情報の作成 56  
 直接型リンクによるリンク元オブジェクトの取得 322

## て

---

定数定義クラス 560  
 定数定義クラスのカテゴリー一覧 31  
 ディレクトリサービス 560  
 データの参照の取得 170  
 データの取得 171

## と

---

同義語展開検索 560  
 登録するコンテンツのパスまたはダウンロード先のパスの取得 141  
 登録するコンテンツのパスまたはダウンロード先のパスの設定 145  
 登録する文書のファイル名の取得 210  
 登録する文書のファイル名の設定 216



登録する文書のフルパスの取得 207  
 登録する文書のフルパスの設定 212  
 登録する文書のレンディションタイプの取得 209  
 登録する文書のレンディションタイプの設定 215  
 独立データ 561  
 独立データの作成 249  
 特権 561  
 トップオブジェクト 561  
 トップオブジェクトクラス 561  
 トランザクション 561  
 トランザクションの開始 370  
 トランザクションの確定 372  
 トランザクションの取り消し 377  
 トレースクラス詳細 493  
 トレースクラスのクラス 33  
 トレースクラスのクラス, およびメソッド一覧 33  
 トレースクラスの初期化 508  
 トレースクラスのメソッド一覧 33  
 トレース出力先を表す定数 488  
 トレース情報 561  
 トレースレベルを表す定数 487

## な

---

名前付き検索結果 561  
 名前付き検索結果かどうかの判定 184  
 名前なし検索結果 561  
 名前なし列の削除 189

## は

---

バージョンニングオブジェクト 561  
 バージョンオブジェクト 561  
 バージョンオブジェクトのバージョン識別子の取得 333  
 バージョン管理 562  
 バージョン管理権 562  
 バージョン識別子 562  
 バージョン情報の取得順序を表す定数 475  
 バージョン付きオブジェクトのチェックアウト 308  
 バージョン付きオブジェクトのチェックアウト状態の取得 317  
 バージョン付きオブジェクトのチェックアウトの取り消し 298  
 バージョン付きオブジェクトのチェックイン 306  
 バージョン付きオブジェクトのバージョンニングオブジェクトの取得 334  
 バージョン付きオブジェクトのバージョン一覧の取得 335  
 バージョン付きフォルダ 562  
 バージョン付きフォルダの作成 260

バージョン付き文書 562  
 バージョン付き文書の作成 257  
 バージョンなしフォルダ 562  
 バージョンなしフォルダの作成 247  
 バージョンなし文書 562  
 バージョンなし文書の作成 244  
 バージョンの削除 311  
 バージョンの取得 491  
 パーミッション 562  
 パーミッションの取得 70  
 パーミッションの設定 76  
 パーミッションを表す定数 476  
 バインド 562  
 バインドしているパブリック ACL 一覧の取得 328  
 パブリック ACL 562  
 パブリック ACL にバインドしている文書空間オブジェクト一覧の取得 315  
 パブリック ACL の OIID エレメントの作成 51  
 パブリック ACL の OIID 文字列の取得 128  
 パブリック ACL の OIID 文字列の設定 130  
 パブリック ACL の OIID 文字列のプロパティ値集合の取得 129  
 パブリック ACL の OIID 文字列のプロパティ値集合の設定 131  
 パブリック ACL のアンバインド 348  
 パブリック ACL の作成 255  
 パブリック ACL のバインド 296  
 パラメタクラス 562  
 パラメタクラスのインターフェース, およびメソッド一覧 4  
 パラメタクラスのインターフェース一覧 4  
 パラメタクラスのインターフェースで説明する項目 34  
 パラメタクラスのインターフェースの継承関係 6  
 パラメタクラスのメソッド一覧 6  
 パラメタ情報の出力 495

## ひ

---

表名の取得 177

## ふ

---

ファクトリインターフェースの取得 39  
 ファクトリオブジェクト 562  
 ファクトリクラス 562  
 ファクトリクラスのインターフェース, およびメソッド一覧 2  
 ファクトリクラスのクラスおよびインターフェース 2  
 ファクトリクラスのクラスおよびインターフェース一覧 2

- ファクトリクラスのメソッド一覧 2
  - フィルタリング定義ファイルのフルパスの取得 231
  - フィルタリング定義ファイルのフルパスの設定 236
  - フォルダのリンク先オブジェクト(下位オブジェクト)の取得 318
  - フォルダのリンク元オブジェクトの取得 326
  - 複数の構成管理型リンクの構成管理モードを一括して FIX モードに変更 283
  - 複数の構成管理型リンクの構成管理モードを一括して FLOATING モードに変更 284
  - 複数の文書空間オブジェクトアクセスインターフェースの取得 253
  - 複数の文書空間オブジェクトプロパティ値の一括取得 363
  - 複数の文書空間オブジェクトプロパティ値の一括設定 367
  - 複数のリンクオブジェクトアクセスインターフェースの取得 251
  - プライマリグループ 562
  - フラッシュ 562
  - フルコントロール 563
  - プロパティ 563
  - プロパティ更新権 563
  - プロパティ参照権 563
  - プロパティ値(列)の追加 220
  - プロパティ値が NULL 値かどうかの判定 122
  - プロパティ値集合 563
  - プロパティ値集合(行)の追加 219
  - プロパティ値集合オブジェクトの作成 50
  - プロパティ値集合の取得 226
  - プロパティ値の参照の設定 124
  - プロパティ値の設定 125
  - プロパティ定義元の DMA クラス 563
  - プロパティディスクリプション 563
  - プロパティディスクリプションの取得 390
  - プロパティの削除 227
  - プロパティのデータ型の取得 403
  - プロパティ名の一括変更 115
  - プロパティ名の取得 404
  - プロパティ名の変更 114
  - 文書 563
  - 文書管理クラス 563
  - 文書管理クラスのインターフェース, およびメソッド一覧 14
  - 文書管理クラスのインターフェース一覧 14
  - 文書管理クラスのインターフェースの継承関係 15
  - 文書管理クラスのメソッド一覧 15
  - 文書間リンク 563
  - 文書間リンク一覧の取得 329
  - 文書間リンク設定情報一覧の取得条件を表す定数 480
  - 文書間リンク設定情報の作成 58
  - 文書空間 563
  - 文書空間アクセスオブジェクト 563
  - 文書空間オブジェクト 563
  - 文書空間オブジェクトアクセスインターフェースの取得 252
  - 文書空間オブジェクトの一括削除 365
  - 文書空間オブジェクトの削除 344
  - 文書空間オブジェクトのバージョン識別子の取得 380
  - 文書空間オブジェクトのバージョン識別子リストの取得 383
  - 文書空間オブジェクトのプロパティ値の設定 355
  - 文書空間オブジェクトのプロパティ値の読み込み 342
  - 文書空間オブジェクトのプロパティのデータ型 35
  - 文書空間オブジェクトのプロパティのデータ型を表す定数 468
  - 文書空間識別子 563
  - 文書空間識別子の取得 395
  - 文書空間で使用する文字コード種別が UTF-8 の場合の機能差異 548
  - 文書空間に対するユーザの特権を表す定数 478
  - 文書空間のメタ情報の取得 265
  - 文書のアップロード情報 563
  - 文書のアップロード情報オブジェクトの作成 62
  - 文書のコンテンツのアップロード 353
  - 文書のコンテンツのダウンロード 313
- へ
- 
- 変換フラグ 564
- ま
- 
- マスタレンディション 564
  - マスタレンディションの変更 300
  - マッピングされた DMA クラス名のリストの取得 109
  - マッピングされたプロパティ値集合の取得 110
  - マッピング定義名の取得 233
  - マッピング定義名の設定 238
  - マップ 564
  - マルチレンディション機能 564
  - マルチレンディション文書 564
- め
- 
- メソッド 564

メソッドで説明する項目 34  
 メソッドの入り口情報の出力 500  
 メソッドの出口情報の出力 505  
 メタクラス 564  
 メタクラスのインターフェース およびメソッド一覧 21  
 メタクラスのインターフェース一覧 21  
 メタクラスのメソッド一覧 21  
 メタ情報の取得 401  
 メタプロパティ数の取得 223  
 メタプロパティの取得 224  
 メタマネージャーインターフェースの取得 40  
 メッセージ情報 564  
 メッセージテキスト中の可変値 516  
 メッセージの形式 516  
 メッセージの形式と見方 516  
 メッセージの出力 510  
 メッセージの詳細 518  
 メッセージの見方 516

## ゆ

---

ユーザ管理システム 564  
 ユーザ権限 564  
 ユーザ権限定義ファイル 564  
 ユーザ情報 565  
 ユーザ情報の取得 373  
 ユーザ定義プロパティ 565

## よ

---

要素の DbjVerObj インターフェースの取得 382  
 要素のプロパティ値集合の設定 366

## ら

---

ライブラリ情報取得クラス 565  
 ライブラリ情報取得クラスのクラス 32  
 ライブラリ情報取得クラスのクラス, およびメソッド一覧 32  
 ライブラリ情報取得クラスのメソッド一覧 32  
 ランキング検索 565

## り

---

リスト 565  
 リスト要素の DbjLinkObj インターフェースの取得 286  
 リスト要素の取得 360  
 リファレンス種別の取得 136  
 リファレンスタイプを表す定数 479

リファレンスファイル文書のアップロード情報オブジェクトの作成 53  
 リファレンスファイル文書のパス情報オブジェクトの作成 52  
 リファレンスファイル文書のパス情報の取得 148  
 リファレンスファイル文書のパス情報の設定 149  
 リレーション種別 565  
 リンク Proxy オブジェクト 565  
 リンク Proxy オブジェクトのプロパティ値集合インターフェースの取得 276  
 リンク Proxy オブジェクトのプロパティ値集合の設定 279  
 リンクオブジェクト 565  
 リンクオブジェクトの一括削除 290  
 リンクオブジェクトの削除 278  
 リンクオブジェクトのプロパティ値集合の取得 196  
 リンクオブジェクトのプロパティ値集合の設定 197  
 リンク権 565  
 リンク先オブジェクト 565  
 リンク先オブジェクトとのリンク 337  
 リンク先オブジェクトとのリンクの解除 350  
 リンク先オブジェクトの取得 195, 275  
 リンク先オブジェクトの設定 198  
 リンク先オブジェクトのリストの取得 288  
 リンク識別子 565  
 リンク識別子の指定によるリンク先オブジェクトとのリンクの解除 352  
 リンク識別子の取得 272  
 リンク種別 565  
 リンク種別の取得 194, 273  
 リンク種別を表す定数 471  
 リンク設定情報 565  
 リンクプロパティ値集合プロパティ 566  
 リンクプロパティ値の一括取得 289  
 リンクプロパティ値の一括設定 291  
 リンクプロパティ値の設定 280  
 リンクプロパティの取得 277  
 リンク元オブジェクト 566  
 リンク元オブジェクトの取得 274  
 リンク元オブジェクトのリストの取得 287

## れ

---

例外クラス 566  
 例外クラスのインターフェースの継承関係 26  
 例外クラスのクラス一覧 23  
 例外クラスのクラス一覧, スーパークラス, およびコンストラクタ 23  
 列インデクスの取得 161  
 列数の取得 163

列データの取得 167  
列の重複の排除 160  
列のデータ型の取得 166  
列名 566  
列名の取得 165  
列名の取得（表名を含む） 164  
列名の設定 190  
レンジション 566  
レンジション情報 566  
レンジション情報一覧の取得 331  
レンジション情報の取得 154  
レンジション情報リスト 566  
レンジションステータスを表す定数 481  
レンジションタイプ 566  
レンジションタイプに対応する拡張子の取得 396  
レンジションタイプの取得 151  
レンジションタイプのリストの取得 155  
レンジション定義ファイル 566  
レンジションの削除 310  
レンジションの追加 294  
レンジションのプロパティ値集合の取得 152  
レンジションプロパティ値集合の取得 211  
レンジションプロパティ値集合の設定 214  
レンジションプロパティの設定 357  
レンジション変換 566

## ろ

---

ローカル ACL 566  
ロード 566  
ログアウト 376, 566  
ログイン 375, 566  
ロック指定検索 566  
ロック種別 567  
ロック種別プロパティ 567  
ロック種別を表す定数 472

## わ

---

ワイルドカード 567