



Sun Java Enterprise System 用語集



Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

Part No: 819-4629
2007 年 3 月

本製品および本書は著作権法によって保護されており、その使用、複製、頒布、および逆コンパイルを制限するライセンスのもとにおいて頒布されます。サン・マイクロシステムズ株式会社による事前の許可なく、本製品および本書のいかなる部分も、いかなる方法によっても複製することが禁じられます。フォント技術を含む第三者のソフトウェアは、著作権により保護されており、提供者からライセンスを受けているものです。

本製品の一部は Berkeley BSD システムより派生したもので、カリフォルニア大学よりライセンスを受けています。UNIX は、X/Open Company, Ltd. が独占的にライセンスしている米国ならびにほかの国における登録商標です。

Sun、Sun Microsystems、Sun のロゴマーク、docs.sun.com、AnswerBook、AnswerBook2、Java、Solaris は、米国およびその他の国における米国 Sun Microsystems, Inc. (以下、米国 Sun Microsystems 社とします) の商標もしくは登録商標です。Sun のロゴマークおよび Solaris は、米国 Sun Microsystems 社の登録商標です。すべての SPARC 商標は、米国 SPARC International, Inc. のライセンスを受けて使用している同社の米国およびその他の国における商標または登録商標です。SPARC 商標が付いた製品は、米国 Sun Microsystems 社が開発したアーキテクチャーに基づくものです。

OPEN LOOK および Sun™ Graphical User Interface は、米国 Sun Microsystems 社が自社のユーザーおよびライセンス実施権者向けに開発しました。米国 Sun Microsystems 社は、コンピュータ産業用のビジュアルまたはグラフィカルユーザーインターフェースの概念の研究開発における米国 Xerox 社の先駆者としての成果を認めるものです。米国 Sun Microsystems 社は米国 Xerox 社から Xerox Graphical User Interface の非独占的ライセンスを取得しており、このライセンスは、OPEN LOOK GUI を実装するか、または米国 Sun Microsystems 社の書面によるライセンス契約に従う米国 Sun Microsystems 社のライセンス実施権者にも適用されます。

U.S. Government Rights – Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

本書は、「現状のまま」をベースとして提供され、商品性、特定目的への適合性または第三者の権利の非侵害の黙示の保証を含みそれに限定されない、明示的であるか黙示的であるかを問わない、なんらの保証も行われないものとします。

はじめに

この用語集では、Java Enterprise System で使用される用語を定義します。

この用語集は次の要素で構成されます。

- 頭字語
- その用語の品詞。名詞、動詞など。次の略語が使われています。
 - adj. - 形容詞
 - n. - 名詞
 - v. - 動詞
- 番号付きの使用法 (複数の製品またはテクノロジーで使用される場合)
- 相互参照
- 同義語
- 対照的な概念を持つ用語
- 略語を展開したもの (略語の場合)

表記上の規則

このマニュアルでは、次のような字体や記号を特別な意味を持つものとして使用します。

表1 表記上の規則

字体または記号	意味	例
AaBbCc123	コマンド名、ファイル名、ディレクトリ名、画面上的コンピュータ出力、コード例を示します。	.login ファイルを編集します。 ls -a を使用してすべてのファイルを表示します。 machine_name% you have mail.
AaBbCc123	ユーザーが入力する文字を、画面上的コンピュータ出力と区別して示します。	machine_name% su Password:
aabbcc123	変数を示します。実際に使用する特定の名前または値で置き換えます。	ファイルを削除するには、rm filename と入力します。
『』	参照する書名を示します。	『コードマネージャー・ユーザーズガイド』を参照してください。
「」	参照する章、節、ボタンやメニュー名、強調する単語を示します。	第5章「衝突の回避」を参照してください。 この操作ができるのは、「スーパーユーザー」だけです。
\	枠で囲まれたコード例で、テキストがページ幅を超える場合に、継続を示します。	sun% grep '^#define \ XV_VERSION_STRING'

コード例は次のように表示されます。

- C シェル

```
machine_name% command y|n [filename]
```

- C シェルのスーパーユーザー

```
machine_name# command y|n [filename]
```

- Bourne シェルおよび Korn シェル

```
$ command y|n [filename]
```

- Bourne シェルおよび Korn シェルのスーパーユーザー

```
# command y|n [filename]
```

[] は省略可能な項目を示します。上記の例は、filename は省略してもよいことを示しています。

|は区切り文字(セパレータ)です。この文字で分割されている引数のうち1つだけを指定します。

キーボードのキー名は英文で、頭文字を大文字で示します(例: Shift キーを押します)。ただし、キーボードによっては Enter キーが Return キーの動作をします。

ダッシュ (-) は2つのキーを同時に押すことを示します。たとえば、Ctrl-D は Control キーを押したまま D キーを押すことを意味します。

コマンド例のシェルプロンプト

次の表は、デフォルトのシステムプロンプトとスーパーユーザープロンプトを示しています。

表2 シェルプロンプト

シェル	プロンプト
UNIX および Linux システムの C シェル	machine_name%
UNIX および Linux システムの C シェルのスーパーユーザー	machine_name#
UNIX および Linux システムの Bourne シェルおよび Korn シェル	\$
UNIX および Linux システムの Bourne シェルおよび Korn シェルのスーパーユーザー	#
Microsoft Windows のコマンド行	C:\

記号の規則

次の表は、この用語集で使用される記号の一覧です。

表3 記号の規則

記号	説明	例	意味
[]	省略可能な引数やコマンドオプションが含まれます。	ls [-l]	-l オプションは必須ではありません。
{ }	必須のコマンドオプションの選択肢を囲みます。	-d {y n}	-d オプションには y 引数か n 引数のいずれかを使用する必要があります。

表3 記号の規則 (続き)

記号	説明	例	意味
<code>\${ }</code>	変数参照を示します。	<code>\${com.sun.javaRoot}</code>	<code>com.sun.javaRoot</code> 変数の値を参照します。
-	同時に押すキーを示します。	Control-A	Control キーを押しながら A キーを押します。
+	順番に押すキーを示します。	Ctrl+A+N	Control キーを押してから放し、それに続くキーを押します。
→	グラフィカルユーザーインターフェイスでのメニュー項目の選択順序を示します。	「ファイル」→「新規」 →「テンプレート」	「ファイル」メニューから「新規」を選択します。「新規」サブメニューから「テンプレート」を選択します。

Sun のオンラインリソースへのアクセス

docs.sun.com Web サイトでは、Sun の技術文書にオンラインでアクセスできます。docs.sun.com アーカイブをブラウズすることも、特定のマニュアルのタイトルまたは主題を検索することもできます。マニュアルは、PDF および HTML 形式のオンラインファイルとして利用できます。どちらの形式でも、障害をもつユーザーのための支援技術により読み取り可能です。

次の Sun のリソースにアクセスするには、<http://www.sun.com> をご覧ください。

- Sun 製品のダウンロード
- サービスおよびソリューション
- サポート (パッチおよびアップデートを含む)
- トレーニング
- 研究
- コミュニティ (Sun Developer Network など)

サードパーティーの Web サイト

このマニュアルに掲載されているサードパーティーの URL を参照すると、追加および関連情報を入手できます。

注- このマニュアル内で述べられるサードパーティーの Web サイトが、現在利用できるかどうかについて Sun は責任を負いません。こうしたサイトやリソース上またはこれらを通じて利用できるコンテンツ、広告、製品、その他の資料について Sun は推奨しているわけではなく、Sun はいかなる責任も負いません。また、このようなサイトやリソース上で、またはサイトやリソースを通じて利用できるコンテンツ、製品、サービスの使用または依存を原因として、または使用や依存に関連して生じた、または生じた疑いのある実際の損傷や損失、あるいは損傷や損失の疑いのあるものに対して Sun は責任を負いません。

ユーザーからのご意見

Sun は当社のマニュアルの改善のために、ユーザーからのご意見ご提案を受け付けています。ご意見をいただくには、<http://docs.sun.com> のページから「コメントの送信」をクリックしてください。オンラインフォームに完全な文書のタイトルとパーツ番号を入力してください。パーツ番号はこのマニュアルの表紙またはマニュアルの URL に示されている 7 桁か 9 桁の数字です。たとえば、このマニュアルのパーツ番号は 819-4629 です。

A

抽象スキーマ (abstract schema)	(n.) Bean の持続フィールドと関係を定義する、エンティティ Bean の配備記述子の一部分。「 エンティティ Bean (entity bean) 」、「 持続性 (persistence) 」を参照してください。「 スキーマ 」も参照してください。
抽象スキーマ名 (abstract schema name)	(n.) 「 EJB QL 」照会で参照される論理名。
アクセス制御 (access control)	(1) (n.) サーバーへのアクセスを制御して、サーバーをセキュリティー保護するための方法。 (2) (n.) リソースと相互作用するための方法は、整合性、機密性、および可用性の制約を課すため、ユーザーまたはグループのコレクションに制限されます。
アクセス制御エントリ (access control entry)	「 ACE (アクセス制御エントリ) 」を参照してください。
アクセス制御命令 (access control instruction)	「 ACI (アクセス制御命令) 」を参照してください。
アクセス制御リスト (access control list)	「 ACL (アクセス制御リスト) 」を参照してください。
アクセス制御規則 (access control rules)	(n.) 特定のディレクトリエントリまたは属性のセットに対するユーザーのアクセス権を指定する規則。
アクセスドメイン (access domain)	(n.) 指定されたドメイン内からの特定の Messaging Server に対するアクセスを制限するドメイン。たとえば、アクセスドメインを使用すると、特定のアカウント宛てのメールを収集できる場所を制限できます。
アクセサ (accessor)	(n.) LDAP などのプロトコルを使用して、ディレクトリリソースと直接やりとりするコネクタ層。Identity Synchronization for Windows では、Directory Server、Active

Directory、Windows NT用に個々の実装が用意されています。アクセサは、動作についてログメッセージで多く参照されます。

アクセス権限 (access right)	(n.) アクセス権限により、付与または拒否するアクセス制御レベルを指定します。アクセス権限は、ディレクトリで実行可能な動作のタイプに関連があります。アクセス権限として、読み取り、書き込み、追加、削除、検索、比較、自己書き込み、プロキシ、およびすべてを付与または拒否できます。
アカウント (account)	(n.) 特定のユーザーまたはユーザーグループを定義する情報。この情報には、ユーザー名やグループ名、1つまたは複数の有効な電子メールアドレス、および電子メールの配信方法と配信先が含まれます。
アカウントの無効化 (account inactivation)	(n.) 単一のユーザーアカウントまたはアカウントセットを無効化し、認証動作が自動的に拒否されるようにします。
ACE (アクセス制御エントリ)	(ACE (access control entry)) (1) (n.) アクセス制御リストの単一の情報項目。アクセス制御情報とも呼ばれます。 (2) (n.) 受信したアクセス要求を評価するために Web サーバーが使用する規則の階層。 (3) (n.) カレンダー、カレンダープロパティ、およびイベントやタスクなどのカレンダーコンポーネントのアクセス制御に使用する文字列。
ACI (アクセス制御命令)	(ACI (access control instruction)) (n.) ディレクトリのエントリにアクセス権を付与する、またはアクセス権を拒否する命令。
ACID	(adj.) 「トランザクション (transaction)」で保証される4つの特性の頭字語。つまり原子性 (atomicity)、一貫性 (consistency)、隔離性 (isolation)、持続性 (durability) です。
ACL (アクセス制御リスト)	(ACL (access control list)) (1) (n.) ディレクトリへのアクセスを制御する仕組み。Directory Server の場合、ACL はディレクトリエントリの ACI 属性です。 (2) (n.) ACE の集合。ACL は、サーバーへのアクセス権を持つユーザーを定義する仕組みです。特定のファイルやディレクトリに固有の ACL 規則を定義して、1つ以上のユーザーまたはグループへのアクセスを付与または拒否できます。 (3) (n.) カレンダー、カレンダープロパティ、およびイベントやタスクなどのカレンダーコンポーネントのアクセスをまとめて制御する一連の ACE 文字列。 (4) (n.) ディレクトリにアクセスする場合の、ユーザー、グループ、またはユーザーとグループが持つアクセス権を定義する、ディレクトリに関連付けられた一連のデータ。ACL は、1つ以上の ACE 文字列で構成されます。
アカウント連携 (account federation)	「アイデンティティ連携 (identity federation)」を参照してください。

累積パッチ (accumulated patch)	(n.) 以前のパッチ (複数の場合もある) の修正版、同じパッチの旧バージョン、および現在リリース中の修正版のセットを統合するパッチプログラム。
アクティベーション (activation)	(n.) エンタープライズ Bean の状態を補助記憶装置からメモリーに転送するプロセス。「 パッシベーション (passivation) 」も参照してください。
アクティブブート環境	(n.) 現在、起動して稼働している環境。
アクティブノード (active node)	(n.) セッションデータを保管している HADB ノード。アクティブノードで障害が発生すると、スペアノードがミラーノードからデータをコピーしてアクティブになります。「 HADB ノード (HADB node) 」、「 スペアノード (spare node) 」、「 ミラーノード (mirror node) 」、および「 データ冗長ユニット (DRU) 」も参照してください。
アドレス (address)	(n.) 電子メールメッセージの送信先と送信方法を決定するメッセージ内の情報。アドレスはメッセージヘッダーとメッセージエンベロップの両方に表示されます。エンベロップのアドレスは、メッセージの通る経路と配信方法を定めます。ヘッダーアドレスは、表示のためだけに存在しています。
アドレス処理 (address handling)	(n.) アドレス指定のエラーを検出し、必要に応じてアドレスを書き換え、アドレスと受取人の照合を行うために MTA によって実行される処理。
アドレス指定プロトコル (addressing protocol)	(n.) 電子メールの利用を可能にするアドレス指定規則。RFC 822 は、インターネット上でもっとも幅広く使用されているプロトコルで、Messaging Server でサポートされています。それ以外には、X.400 や UUCP などのプロトコルがあります。
アドレストークン (address token)	(n.) 書き換え規則パターン of アドレス要素。
管理コンソール (admin console)	(n.) Java™ Enterprise System サーバーおよびそのコンポーネントの設定、管理、監視、維持、トラブルシューティングに使用される、ブラウザベースの一連のフォーム。 (n.) 管理者が Portal Server 6.0 で作業する際の Directory Server Access Management Edition GUI インタフェース。
管理対象オブジェクト (administered object)	(n.) 接続ファクトリや送信先などのあらかじめ設定された Java Enterprise System のオブジェクト。1 つ以上の JMS クライアントで使用するために管理者が作成します。 管理対象オブジェクトを使用して、Java Message Service (JMS) クライアントをプロバイダ独自の要素から孤立させます。これらのオブジェクトは、管理者によって Java Naming and Directory Interface™ (JNDI) 名前空間に配置されます。JMS クライアントは、JNDI 検索を使用してそれらのオブジェクトにアクセスします。
管理コンソール (administration console)	「 管理コンソール (admin console) 」を参照してください。

管理ドメイン
(administration
domain)

「ドメイン (domain)」を参照してください。

管理インタ
フェース
(administration
interface)
管理ノード
(administration
node)

「管理コンソール (admin console)」を参照してください。

(n.) リモート管理サーバーと通信できる Web Server ノード。「[クラスタ](#)」または「[サーバーファーム \(server farm\)](#)」内の各ノードには、そこで稼働している管理サーバーまたは管理ノードがあります。このようなノードの1つがマスターサーバーとして構成され(管理サーバーと呼ばれます)、残りがスレーブサーバーとして構成されます(管理ノードと呼ばれます)。

管理権限
(administration
privileges)

(n.) ユーザーの管理ロールを定義する一連の権限。

管理サーバー
(administration
server)

(n.) Java Enterprise System コンポーネント製品の管理機能を提供する専用のサーバー。

Administration
Server 管理者
(administration
server
administrator)

(n.) Java Enterprise System Directory Server に接続していない場合でも、サーバーの起動および停止を行う管理権限を持つユーザー。Administration Server 管理者は、ローカルサーバーグループ内のすべてのサーバーに対する制限されたサーバーに関する作業(通常はサーバーの再起動と停止のみ)を実行できます。Administration Server をインストールすると、この管理者のエントリが自動的にローカルに作成されます。この管理者は、ユーザーディレクトリのユーザーではありません。

管理ドメイン

[ドメイン \(domain\)](#)を参照してください。

管理者
(administrator)

(n.) 定義された一連の管理権限を持つユーザー。「[構成管理者 \(configuration administrator\)](#)」、「[ディレクトリマネージャー \(Directory Manager\)](#)」、「[Administration Server 管理者 \(administration server administrator\)](#)」、「[サーバー管理者 \(server administrator\)](#)」、「[ファミリーグループ管理者 \(family group administrator\)](#)」、および「[メールリスト所有者 \(mail list owner\)](#)」も参照してください。

admpw

(n.) Sun Enterprise™ Administrator Server スーパーユーザーのユーザー名とパスワードのファイル。

採用シナリオ
(adoption
scenario)

Java Enterprise System ソフトウェアの総合的な配備理由。配備するソフトウェアシステムの特性を述べ、達成目標を示します。Java Enterprise System の基本的な採用シナリオは、新規システム、置換、拡張、およびアップグレードの4つです。

連携プロバイダ
(affiliation)

(n.) 特定の認証ドメインとは無関係に形成されるプロバイダのグループ。連携プロバイダの所有者によって形成および保守されます。連携プロバイダのドキュメントには、プロバイダ ID によってまとめて識別されるプロバイダのグループが記述されて

います。連携プロバイダのメンバーは、連携プロバイダ ID により連携プロバイダのメンバーとして、またはプロバイダ ID により個々のプロバイダとして、サービスを呼び出すことが可能です。

エージェント (agent)	(1) (n.) ルーター、ホスト、あるいは X 端末などのネットワークデバイス上でネットワーク管理ソフトウェアを実行するソフトウェア。「インテリジェントエージェント (intelligent agent)」、「ノードエージェント (node agent)」も参照してください。 (2) (n.) Identity Synchronization for Windows では、Message Queue とのインタフェースとなるコネクタコンポーネント。属性を Directory Server 名と Windows 名との間で変換します。エージェントは、動作についてログメッセージで多く参照されます。
アラームイベント (alarm event)	(n.) Calendar Server ENS によって生成され送信されるイベント。アラームイベントが発生すると、指定の受信者にメッセージアラームが送信されます。
アラート (alert)	(n.) ポップアップウィンドウに即座に表示される緊急メッセージ。送信者は、メッセージの受信者を把握しており、「メッセージステータスを表示」オプションが使用されると、警告が閉じられるかクリックされたときに、緊急メッセージが読まれたことが通知されます。警告メッセージに応答が必要な場合は、警告を右クリックしてコンテキストメニューを表示し、「送信者とチャット(R)」オプションを使用します。
エイリアス ファイル (alias file)	(n.) ポストマスターエイリアスなど、ディレクトリ内に設定されていないエイリアスを設定するために使用されるファイル。
エイリアス化 (aliasing)	(n.) 異なるスキーマを持つ別の検索エンジンからリソース記述をインポートする場合に、エイリアスを使用する Java Enterprise System Portal Server の検索エンジンで、ある項目の代わりに別の項目を使用すること。
すべての ID のし きい値 (All IDs threshold)	(n.) Java Enterprise System Directory Server によって管理されるすべてのインデックスにグローバルに適用されるサイズ制限。エン트리 ID リストのサイズがこの制限値に達すると、サーバーはそのエン트리 ID リストをすべての ID のトークンで置き換えます。
すべての ID の トークン (All IDs token)	(n.) すべてのディレクトリエントリがインデックスキーに一致するとサーバーに想定させる仕組み。実際、すべての ID のトークンにより、Java Enterprise System Directory Server は、インデックスを使用しない検索を実行してインデックスキーを一致させます。
許可された属性 (allowed attributes)	(n.) 特定のオブジェクトクラスを使用してエントリ内に存在可能なオプションの属性。「属性 (attribute)」、「必須の属性 (required attributes)」も参照してください。
許可フィルタ (Allow filter)	(n.) 1 つ以上の POP、IMAP、または HTTP サービスへのアクセスが許可されるクライアントを識別する、Java Enterprise System Messaging Server のアクセス制御規則。「拒否フィルタ (Deny filter)」も参照してください。

代替アドレス (alternate address)	(n.) アカウントの二次アドレス。通常は主アドレスを変化させたものです。1つのアカウントに複数のアドレスがあると便利な場合があります。
代替ルート (alternate root)	(n.) パッケージがインストールされたクライアントの root ファイルシステムの位置。代替ルートは、通常、pkgadd -R を使用して指定されます。
AML (抽象マークアップ言語)	(AML (abstract markup language)) (n.) 特定のベンダーやモデルに依存しないモバイルデバイスマークアップ言語。
匿名アクセス (anonymous access)	(1) (n.) 「 認証 (authentication) 」なしで「 リソース (resource) 」にアクセスすること。 (2) (n.) 権限を付与されている場合に、資格を入力せずに、バインド状態に関係なく、ディレクトリ情報へのアクセスを誰にでも許可するアクセス方法。
API	(API (application programming interface)) (1) (n.) ほかのソフトウェアまたはハードウェアとの通信を行うために、コンピュータプログラムが使用する命令の集まり。同じ API を解釈するように設計された対象と通信できます。 (2) (n.) プログラムが既存のソフトウェアパッケージのサービスを呼び出す方法を定義した、呼び出し規則または命令の集まり。
APOP	(authenticated post office protocol) (n.) POP に似ていますが、APOP の認証では、プレーンテキストによるパスワードではなく暗号化したパスワードとチャレンジ文字列を使用します。
アプレットコンテナ (applet container)	(n.) アプレットプログラミングモデルのサポートが含まれた「 コンテナ (container) 」。
アプリケーションアセンブラ (application assembler)	(n.) J2EE™ 「 構成要素 (コンポーネント) 」と「 モジュール 」を配備可能なアプリケーション単位に結合する担当者。
アプリケーションクライアント (application client)	(n.) 専用の Java 仮想マシン内で実行する、第 1 層の J2EE クライアントコンポーネント。アプリケーションクライアントは、J2EE プラットフォーム API の一部にアクセスできます。
アプリケーションクライアントコンテナ (application client container)	(n.) アプリケーションクライアントコンポーネントをサポートするコンテナ。「 コンテナ (container) 」を参照してください。
アプリケーションクライアントモジュール (application client module)	(n.) 1つ以上のクラスと、アプリケーションクライアント配備記述子で構成されるソフトウェアユニット。

アプリケーションコンポーネント
(application component)

「構成要素 (コンポーネント)」を参照してください。

アプリケーションコンポーネントプロバイダ
(application component provider)
構成リソースファイル
(application configuration resource file)

(n.) コンポーネントのメソッド、JSP ページ定義、およびその他の必須の配備記述子を配備する Java クラスを提供するベンダー。

(n.) JavaServer Faces アプリケーションのリソースを構成したり、アプリケーションのナビゲーションルールを定義したり、コンバータ、バリデータ、リスナー、レンダラ、およびコンポーネントをアプリケーションに登録したりするために使用される XML ファイル。

Application Server

(n.) Sun Java Enterprise System に含まれるアプリケーションサーバー製品。

アプリケーションサーバー
(application server)

(n.) ビジネスアプリケーションが実行されるソフトウェアプラットフォーム。通常、アプリケーションサーバーは、コンポーネントのライフサイクル、位置、配置、およびトランザクションリソースアクセスなど、高レベルのサービスをアプリケーションに提供します。

アプリケーションサービス
(application service)

(n.) 複数のクライアントのためにビジネスロジックを実行するコンポーネントまたはコンポーネントアセンブリ。そのため、マルチスレッドプロセスにする必要があります。また、アプリケーションサービスは、Web サービスまたはスタンドアロンのコンテンツサーバーとしてカプセル化されたコンポーネントであったり、コンポーネントアセンブリであったりします。

アプリケーション層
(application tier)

(n.) J2EE アプリケーションの概念的な区分。

クライアント層: ユーザーインタフェース。エンドユーザーは、Web ブラウザなどのクライアントソフトウェアと対話してアプリケーションを使用します。

サーバー層: アプリケーションを構成し、アプリケーションのコンポーネント内で定義されているビジネスロジックおよびプレゼンテーションロジック。

データ層: アプリケーションがデータソースと対話できるようにするデータアクセスロジック。

近似インデックス
(approximate index)

(n.) ディレクトリ情報ツリー全体で、効果的な近似検索または「あいまい」検索を実行可能なインデックス。

アーキテクチャー
(architecture)

分散アプリケーションまたはほかのソフトウェアシステムの論理的および物理的な基本要素、およびそれらの相互関係を示す設計。「分散エンタープライズアプリケーション (distributed enterprise application)」の場合、アーキテクチャーの設計には

通常、「論理アーキテクチャー (logical architecture)」と「配備アーキテクチャー (deployment architecture)」の両方が含まれます。

アーカイブ (archiving)	(n.) オブジェクトの状態を保存および復元する処理。
A レコード (A record)	(n.) ホスト名および関連付けられた IP アドレスを含む DNS レコードの一種。A レコードは、Messaging Server がインターネット上で電子メールをルーティングするために使用します。「ドメインネームシステム (domain name system)」、「MX レコード (メール交換レコード)」も参照してください。
asant	(n.) Apache Ant (Java クラスを使用して拡張可能) を基にした構築ツール。構成ファイルは XML ベースで、さまざまなタスクが実行されるターゲットツリーを呼び出します。「構築ファイル (build file)」も参照してください。
アセンブリ (assembly)	(n.) アプリケーションの個別コンポーネントを配備可能な 1 つの単位に結合するプロセス。「配備 (deployment)」も参照してください。
非同期通信 (asynchronous communication)	(n.) メッセージの送信側がほかの作業を続ける前に、送信メソッドが戻るのを待つ必要がない通信のモード。
属性 (attribute)	<p>(1) (n.) サブレットによって設定可能な、要求オブジェクト内の名前と値のペア。XML ファイル内の要素を修正する DTD ファイルで定義済みの名前と値のペアでもあります。「プロパティ (property)」と比較してください。「パラメータ (parameter)」も参照してください。一般的には、属性はメタデータの単位です。</p> <p>(2) (n.) エントリについての記述情報を保持する名前と値のペア。属性にはタイプ (名前) と値のセットがあります。また属性タイプは、そのタイプの属性の値として保管可能な情報の種類に適用される構文も指定します。</p> <p>(3) (n.) Java Enterprise System Directory Server Access Management Edition のサービスが組織に提供するパラメータを定義します。Java Enterprise System Directory Server Access Management Edition サービスを構成する属性は、Dynamic、Policy、User、Organization、または Global のいずれかに分類されます。各サービスでこれらのタイプを使用して属性を細分することにより、サービススキーマをより一貫した方法で整理し、サービスパラメータを容易に管理できます。</p> <p>(4) (n.) Application Server で、組み込まれたサーバー構成の一部である名前と値のペア。「プロパティ (property)」と比較してください。</p>
属性プロバイダ (attribute provider)	(n.) 属性プロバイダは、属性データをホストする Web サービスです。
属性リスト (attribute list)	「オプション属性リスト (optional attribute list)」および「必須の属性リスト (required attribute list)」を参照してください。

監査 (auditing)	(n.) エラーやセキュリティ違反などの重大なイベントが発生した場合に、それをあとから調べることができるようにイベントを記録するメソッド。
AUTH	(n.) SMTP コマンドの1つ。SMTP クライアントがサーバーに対して認証方法を指定したり、認証プロトコル交換を実行したり、必要に応じて次に続くプロトコルの相互対話で使用するセキュリティ層のネゴシエーションを行なったりできるようにします。
認証 Directory Server (authenticating Directory Server)	(n.) PTA の場合、認証 Java Enterprise System Directory Server には、要求元クライアントの認証資格があります。PTA 対応のユーザーディレクトリは、バインド要求を認証ディレクトリに渡し、認証ディレクトリは、要求元クライアントのバインド資格を検証します。
認証 (authentication)	<p>(1) (n.) ユーザー、デバイス、またはコンピュータシステムのその他のエンティティのアイデンティティを確認するプロセス。通常は、システムのリソースへのアクセスを許可するための前提条件になります。Java サブレット仕様では、3種類の認証(基本、フォームベース、相互)が要求され、ダイジェスト認証をサポートします。インターネットなど、プライベートおよびパブリックコンピュータネットワークでは、ログインパスワードを使用して認証を行なっているのが現状です。これは、パスワードを知っていれば、ユーザー本人であるという前提に基づいています。「基本認証 (basic authentication)」、「フォームベース認証 (form-based authentication)」、「相互認証 (mutual authentication)」、および「ダイジェスト認証 (digest authentication)」も参照してください。</p> <p>(2) (n.) Java Enterprise System Directory Server に対し、クライアントユーザーであることを立証するプロセス。ユーザーは、ディレクトリへのアクセスを許可してもらうために、バインド DN と対応するパスワードまたは証明書を示す必要があります。Java Enterprise System Directory Server により、ユーザーは、ディレクトリ管理者から与えられたアクセス権に基づいて機能を実行したり、ファイルおよびディレクトリにアクセスしたりすることができます。「サーバー認証 (server authentication)」も参照してください。</p>
認証証明書 (authentication certificate)	(n.) 相手を検証し認証するために、サーバーからクライアント、またはクライアントからサーバーに送信されるデジタルファイル。証明書は、その所有者であるクライアントまたはサーバーの信頼性を保証します。証明書は譲渡できません。
認証ドメイン (authentication domain)	(n.) Liberty Alliance Project (LAP) を使用したユーザー認証情報の交換に同意する、1つ以上の認証情報プロバイダを持つサービスプロバイダのグループ。「 トラストサークル (circle of trust) 」が確立されると、シングルサインオン認証がすべてのプロバイダの間で有効になります。トラストサークルとも呼ばれます。
承認 (authorization)	(n.) 主体がサービスを使用できるかどうか、主体がどのオブジェクトにアクセスできるか、および、各オブジェクトにどのようなアクセスの種類が許可されているかを決定するプロセス。承認は、認証を介した要求に関連付けられている主体が指定されたセキュリティロールにあるかどうかの判断に基づきます。セキュリティ

ロールとは、アプリケーションをアセンブルする担当者によって定義される、ユーザーの論理グループです。配備担当者は、セキュリティーロールをセキュリティーアイデンティティーにマッピングします。セキュリティーアイデンティティーは、運用環境の主体またはグループであることがあります。

承認制約
**(authorization
constraint)**

(n.) Web リソースコレクションへのアクセスが許可される対象を決定する承認ルール。

自動返信オプションファイル
**(autoreply option
file)**

(n.) 不在返信通知などの電子メール自動返信オプションを設定するために使用するファイル。

自動返信ユーティリティ
**(AutoReply
utility)**

(n.) 自動返信機能が有効になっているアカウント宛てに送信されたメッセージに対し、自動的に返信するためのユーティリティ。Java Enterprise System Messaging Server 内のすべてのアカウントは、受信メッセージに対して自動的に返信するように設定できます。

可用性サービス
**(availability
service)**

(n.) Application Server の機能。サーバーインスタンス、Web コンテナ、EJB コンテナおよび RMI/IIOP 要求での高可用性を実現します。

B

B2B	(adj.)企業間 (business-to-business)。
バックボーン (backbone)	(n.)分散システムの主要な接続メカニズム。バックボーン上の中間システムに接続するすべてのシステムは、相互に接続されます。バックボーンがある場合でも、コスト、パフォーマンス、またはセキュリティなどの理由から、バックボーンを迂回するようにシステムを設定することができます。
バックエンド サーバー (back-end server)	(n.) Java Enterprise System Messaging Server において、電子メールメッセージの保存と取得機能のみを持つ電子メールサーバー。メッセージストアサーバーとも呼ばれます。
バックキング Bean (backing bean)	(n.) JavaBeans コンポーネント。JavaServer Faces コンポーネントを含む JSP ページに対応します。バックキング Bean は、ページ上のコンポーネントのプロパティとコンポーネントの処理を実行するメソッドを定義します。この処理には、ナビゲーションに関連付けられたイベント処理、検証、および処理が含まれます。
バックアウト (backout)	(n.)パッチなどのソフトウェアの変更を取り除くことで、システムを以前の状態に戻すこと。
バックアップする (back up)	(v.)フォルダの内容をメッセージストアからバックアップデバイスにコピーすること。「復元する (restore)」も参照してください。
バックアップストア (backup store)	(n.)データのリポジトリ。一般的にはファイルシステムやデータベースを指します。バックアップストアは、バックグラウンドスレッド (スイーパーレッド) で監視して、不要なエントリを削除することができます。
バナー (banner)	(n.)クライアントがはじめて接続したときに IMAP などのサービスによって表示されるテキスト文字列。

ベース DN (ベース識別名)	(base DN (base distinguished namebase)) (n.) DIT のエントリ。検索操作は、ベース DN によって識別されるエントリ、ベース DN 直下の従属エントリ、または DIT エントリおよび「 DIT 」のベース DN の下にあるすべてのエントリに対して実行可能です。
基本認証 (basic authentication)	(n.) 認証メカニズム。Web アプリケーションの組み込み認証メカニズムを使用して取得されたユーザー名とパスワードによって Web サーバーがエンティティを認証します。
Bean 管理による持続性 (bean-managed persistence)	(n.) エンティティ Bean の変数とリソースマネージャーの間で行なわれるデータ転送が「 エンティティ Bean (entity bean) 」によって管理されるメカニズム。通常、データアクセスロジックは、JDBC™(Java™Data Connectivity) ソフトウェアまたはそれ以外のデータアクセステクノロジーを使って、開発者が決定します。「 コンテナ管理による持続性 (container-managed persistence) 」も参照してください。
Bean 管理によるトランザクション (bean-managed transaction)	(n.) プログラムを使用して開発者により制御される、エンタープライズ Bean のトランザクション境界設定。「 コンテナ管理によるトランザクション (container-managed transaction) 」も参照してください。
Berkeley DB (Berkeley データベース)	(Berkeley DB (Berkeley database)) (n.) トランザクション用のデータベースストア。読み取りと書き込みの同時実行の負荷が大きく、さらにトランザクショントリカバリ可能性が要求されるアプリケーションで使用します。Java Enterprise System Messaging Server では、Berkeley データベースをさまざまな目的で使用します。
バイナリエンティティ (binary entity)	(n.) 「 解析対象外エンティティ (unparsed entity) 」を参照してください。
バインド DN (バインド識別名)	(bind DN (bind distinguished name)) (n.) バインド要求において、Java Enterprise System Directory Server に対する認証で使用される識別名。
バインド (binding)	(1) (v.) 「 XML (拡張可能マークアップ言語) 」ファイルに関して、XML データの厳密に定義された部分を処理するために必要なコードを生成すること。 (2) (v.) JavaServer Faces テクノロジーに関して、UI コンポーネントをバックエンドのデータソース (バックング Bean のプロパティなど) に結びつけること。
バインドルール (bind rule)	(n.) バインドルールは、アクセス制御において、ディレクトリ情報へのアクセス権を得るために、特定のユーザーやクライアントが満たす必要のある資格や条件を規定するものです。
BLOB (バイナリ large オブジェクト)	(BLOB (binary large object)) (n.) 複合オブジェクトフィールドの格納と取り出しに使うデータ型。BLOB は、画像などのバイナリまたは直列化可能なオブジェクトで、大きなバイト配列に変換された後、コンテナ管理による持続性フィールドに直列化されます。

BMP	「 Bean 管理による持続性 (bean-managed persistence) 」を参照してください。
BMT	「 Bean 管理によるトランザクション (bean-managed transaction) 」を参照してください。
本体 (body)	(n.) 電子メールメッセージの一部分。ヘッダーおよびエンベロープは標準形式に従う必要がありますが、メッセージの本体には送信者が決める内容が含まれています。本体には、テキスト、グラフィックス、またはマルチメディアを入れることができます。構造化された本体は MIME 標準に従う必要があります。
BPEL	(ビジネスプロセス実行言語) (n.) プロセス間で相互作用させるために「 Web サービス (web service) 」を拡張するビジネスプロセス言語。BPEL プロセスは、「 XML (拡張可能マークアップ言語) 」表記法で表現されます。
ブローカ (broker)	(n.) JMS (Java Message Service) API メッセージのルーティング、配信、持続性、セキュリティ、およびロギングを管理する Message Queue エンティティ。管理者がパフォーマンスおよびリソースの使用状況を監視およびチューニングするためのインタフェースを提供します。
ブラウズ (browsing)	(n.) Java Enterprise System Portal Server においては、検索データベースにあるカテゴリ分けされたリソースを閲覧すること。
ブラウズインデックス (browsing index)	「 仮想リスト表示インデックス (virtual list view index) 」を参照してください。
構築ファイル (build file)	(n.) 1 つ以上の「 asant 」ターゲットが含まれる XML ファイル。ターゲットは、実行する一連の Apache Ant タスクです。asant を起動すると、実行するターゲットを選択できます。ターゲットが指定されないと、プロジェクトのデフォルトのターゲットが実行されます。「 asant 」も参照してください。
構築モジュール (building module)	(n.) 共有サービスへの依存度が限られた、またはまったく依存しないハードウェアまたはソフトウェアの構成体。最適のパフォーマンスと水平方向のスケラビリティを備えた特定の設定です。
ビジネスロジック (business logic)	(n.) データ統合ロジックやプレゼンテーションロジックではなく、不可欠な機能を含むアプリケーションコード。 EJB テクノロジ (Enterprise JavaBeans テクノロジ) では、このロジックはエンタープライズ Bean のメソッドを使用して実装されます。
ビジネスメソッド (business method)	(n.) ビジネスロジックやアプリケーションのルールを実装するエンタープライズ Bean のメソッド。
ビジネスサービス (business service)	複数のクライアントのためにビジネスロジックを実行する アプリケーションコンポーネント (application component) またはコンポーネントアセンブリ。そのため、マルチスレッドプロセスになります。ビジネスサービスは、 Web サービス (web

service)としてカプセル化された分散コンポーネントのアセンブリまたはスタンドアロンのサーバーになります。

C

- CA** (1) (認証局) (n.) 「[認証局 \(CA\)](#)」を参照してください。
(2) (コネクタアーキテクチャー) (n.) 「[コネクタアーキテクチャー \(connector architecture\)](#)」を参照してください。
- キャッシュ (cache)** (n.) ローカルに保管される元データのコピー。データがキャッシュされていれば、要求があったときに再びリモートサーバーから取得する必要がなくなります。
- キャッシュ制御指令 (Cache Control Directives)** (n.) プロキシサーバーにどの情報をキャッシュさせるかを制御する Java Enterprise System Application Server の機能。キャッシュ制御指令を使うことで、プロキシによるデフォルトのキャッシングがオーバーライドされ、機密情報をキャッシュせずに後から検索することができます。この指令を利用するには、プロキシサーバーが HTTP 1.1 に準拠している必要があります。
- キャッシュされた行セット (cached rowset)** (n.) データソースからデータを取り込み、その後そのデータを確認したり変更したりしながらデータソースから切り離しできるオブジェクト。キャッシュされた行セットには、取得した元のデータ、およびアプリケーションによるデータの変更点の両方が記録されます。アプリケーションが元のデータソースを更新しようとする、行セットがデータソースに再び接続され、変更された行だけがデータベースにマージされます。
- カレンダーアクセスプロトコル (calendar access protocol)** 「[CAP \(カレンダーアクセスプロトコル\)](#)」を参照してください。
- Calendar Express** (n.) Calendar Server にエンドユーザーがアクセスできるようにする、Web ベースのカレンダークライアントプログラム。
- カレンダーグループ (calendar group)** (n.) 複数のカレンダーの管理を容易にする数個のカレンダーの集合。

カレンダー ID (calendar ID)	(n.) Java Enterprise System Calendar Server データベース内のカレンダーに関連付けられた一意の識別子。calidとも呼ばれます。
カレンダー検索 データベース (calendar lookup database)	「CLD (カレンダー検索データベース)」を参照してください。
Calendar Server アプリケー ションプログラ ミングインタ フェース (Calendar Server application programming interface)	「CSAPI (Calendar Server アプリケーションプログラミングインタフェース)」を参照してください。
カレンダーユー ザーエージェント (calendar user agent)	「CUA (カレンダーユーザーエージェント)」を参照してください。
呼び出し可能な 文 (callable statement)	(n.) ストアドプロシージャから結果セットを返すデータベースのデータベースプロシージャまたは関数呼び出しをカプセル化するクラス。
コールバックメ ソッド (callback method)	(n.) コンテナによって呼び出されるコンポーネントメソッド。ライフサイクルで重要なイベントのコンポーネントを通知します。
呼び出し元 (caller)	(n.) 「呼び出し側主体」と同じ。
呼び出し側主体 (caller principal)	(n.) エンタープライズ Bean メソッドの呼び出し側を識別する主体。
CAP (カレンダーア クセスプロトコ ル)	(CAP (calendar access protocol)) (n.) カレンダー操作のための標準インターネットプロトコル。Internet Engineering Task Force (IETF) 規定の要件に基づいています。
capability	(n.) クライアントに提供され、特定の IMAP サービスで利用可能な機能を定義する文字列。
カスケード型削 除 (cascading deletion)	(n.) 別の削除がトリガーとなる削除。カスケード型削除は、「コンテナ管理による持続性 (container-managed persistence)」のある「エンティティ Bean (entity bean)」に指定できます。
カスケード型レ プリケーション (cascading replication)	(n.) カスケード型レプリケーションのシナリオでは、通常ハブサプライヤと呼ばれる1台のサーバーが、特定のレプリカのコンシューマおよびサプライヤの両方として動作します。サーバーは、読み取り専用レプリカを保持し、変更ログを保守します。サーバーは、データのマスターコピーを持つサプライヤサーバーから更新情報を受け取り、今度はその更新情報をコンシューマに供給します。
カタログ (catalog)	「インデックス (index)」を参照してください。

カタログ化 (cataloging)	「インデックス化 (indexing)」を参照してください。
カテゴリ (category)	(n.) 検索データベースにあるリソースの論理グループ。総称して、一連のカテゴリを分類と呼ぶこともあります。
CCPP (複合機能 と設定プロ ファイル)	(CCPP (composite capability and preference profiles)) (n.) Portal Server Mobile Access ソフトウェアの場合に、クライアント検出用の User Agent Profile および事前設定データで使用される仕様。CCPP 仕様には、デバイスの機能とユーザー設定が記述されています。
CDATA	(n.) 「これらの文字を解釈しない」という意味を表す文字データの定義済み「XML (拡張可能マークアップ言語)」タグ。XML 構文の通常のルールが適用される解析対象文字データ (PCDATA) とは逆です。一般に、CDATA セクションは、XML 構文の例を示すために使用されます。「PCDATA」も参照してください。
セントラルロ ガー (central logger)	(n.) すべてのセントラルログを管理するコアコンポーネント。セントラルログは、各コネクタの監査ログとエラーログを集積したものです。管理者は、これらのログを監視することで、Identity Synchronization for Windows インストール全体の健全性を監視できます。セントラルログは、直接、または Identity Synchronization for Windows コンソールから表示できます。デフォルトでは、セントラルログはコアがインストールされたマシンの <install-root>/logs/central/ サブディレクトリにあります。
証明書 (certificate)	(1) (n.) Instant Messaging Server を特定するために使用する電子ドキュメント。公開鍵に関連付けられています。Java Enterprise System Instant Messaging Server は、Instant Messaging サーバー間での証明書の交換をサポートしています。証明書の交換は、個々のユーザーにとっては透過的に実行されます。 (2) (n.) 個人や企業などのエンティティの名前を指定するデジタルデータ。証明書に含まれる公開鍵がそのエンティティのものであることを証明します。クライアントとサーバーの両方が証明書を持つことができます。 (3) (n.) ユーザーまたは CA の公開鍵と識別情報を強力に関連付ける証明書。通常は、ユーザーまたは CA の識別名です。証明書は CA によってデジタル署名され、SSL 接続の設定時に検証して接続相手の公開鍵を取得することができます。X.509 証明書は、caCertificate;binary 属性または userCertificate;binary 属性内のディレクトリに保管されます。
認証局 (CA)	(1) (n.) 暗号化されたトランザクションで使用される公開鍵証明書を発行する社内またはサードパーティーの信頼されている組織。ベアラーに ID を提供します。 (2) (n.) メッセージの暗号化と復号化で使用するセキュリティー資格と公開鍵を発行および管理する、ネットワーク内の機関。PKI の一部として、CA は登録機関で確認処理を行い、デジタル証明書の要求側によって提示される情報を検証します。登録機関が要求側の情報の検証すると、CA は証明書を発行することができるようになります。「PKI (公開鍵基盤)」も参照してください。

<p>証明書に基づく 認証 (certificate-based authentication)</p>	<p>(n.) クライアントが提供したデジタル証明書によるユーザーの識別。「パスワード認証 (password authentication)」も参照してください。</p>
<p>証明書データベース (certificate database) 証明書名 (certificate name)</p>	<p>(n.) サーバーの1つまたは複数のデジタル証明書が含まれているファイル。証明書ファイルとも呼ばれます。</p> <p>(n.) 証明書とその所有者を特定する名前。</p>
<p>証明書失効リスト (certificate revocation list)</p>	<p>「CRL (証明書失効リスト)」を参照してください。</p>
<p>CGI</p>	<p>(CGI (common gateway interface)) (n.) 外部プログラムがHTTPサーバーと通信するために使用するインタフェース。CGIを使用するプログラムは、CGIプログラムまたはCGIスクリプトと呼ばれる。通常サーバーでは処理されないフォームや解析されない出力を、CGIプログラムが処理したり解析したりする。</p>
<p>cHTML</p>	<p>(n.) モバイルデバイスに適したHTMLの簡易版。</p>
<p>更新履歴ログ (change log)</p>	<p>(n.) 更新履歴ログは、レプリカに加えられた変更の記録です。サブライヤサーバーは、マルチマスターレプリケーションの場合に、コンシューマサーバーに保存されたレプリカまたはその他のマスターへの変更を再生します。レプリケーションには使用されない旧バージョン形式の更新履歴ログとは異なります。</p>
<p>チャンネル (channel)</p>	<p>(1) (n.) メッセージを処理する基本的なMTAコンポーネント。チャンネルは、別のコンピュータシステムまたはシステムグループとの接続を表します。各チャンネルは、1つまたは複数のチャンネルプログラムと1つの送信メッセージキューから構成されます。送信メッセージキューには、そのチャンネルに関連付けられている1つまたは複数のシステム宛てのメッセージが格納されます。「チャンネルブロック (channel block)」、「チャンネルホストテーブル (channel host table)」、および「チャンネルプログラム (channel program)」も参照してください。</p> <p>(2) (n.) Java Enterprise System Portal Server デスクトップの場合、チャンネルは、プロバイダと設定で構成されます。チャンネルは、マークアップフラグメント、フレームセット、HTMLページなどで構成可能なコンテンツを生成します。チャンネルコンテンツは、ほかのチャンネルコンテンツと一緒に集約されポータルデスクトップを形成することがよくあります。</p>
<p>チャンネルブロック (channel block)</p>	<p>(n.) 単一のチャンネル定義。「チャンネルホストテーブル (channel host table)」も参照してください。</p>
<p>チャンネルホストテーブル (channel host table)</p>	<p>(n.) チャンネル定義のセット。「チャンネルブロック (channel block)」も参照してください。</p>

チャンネルプログラム (channel program)	(n.) メッセージをリモートシステムに送信して、送信後にメッセージをキューから削除し、リモートシステムからメッセージを受信して適切なチャンネルキューに置くチャンネルの構成要素。「 マスターチャンネルプログラム (master channel program) 」、「 スレーブチャンネルプログラム (slave channel program) 」も参照してください。
文字タイプ (character type)	(n.) アルファベット文字を数値またはほかの文字から区別する属性、および大文字と小文字のマッピング。
チャット (chat)	(n.) Instant Messaging を使ったインスタントメッセージ。チャットはリアルタイムの会話機能です。チャットセッションは、必要に応じて、またはあらかじめ設定された会議室に作成されるチャットルームで開かれます。
チェックポイント (checkpoint)	(n.) ステートフルセッション Bean のライフサイクルで定義済みのポイント。Application Server インスタンスが失敗すると、このポイントで Bean の状態が持続ストアに保存されます。
子 (child)	(1) (n.) 別のカテゴリの下位カテゴリとなるカテゴリ。「 カテゴリ (category) 」も参照してください。 (2) (n.) XML ファイル内の要素。親と呼ばれる別の要素内に含まれます。「 親 (parent) 」も参照してください。
chroot (chroot)	(n.) サーバーを特定のディレクトリに制限するために作成可能な追加ルートディレクトリ。この機能を使用して保護されていないサーバーを守ることができます。
暗号化方式 (cipher)	(n.) 暗号化方式は、暗号化と復号化に使用される暗号化アルゴリズム (数学関数) です。
ciphertext (暗号化テキスト)	(n.) 暗号化された情報。復号化するための正しい鍵を持っていないければ、誰もその情報を読み取ることができません。
トラストサークル (circle of trust)	(n.) 「 認証ドメイン (authentication domain) 」を参照してください。
CKL (危殆化鍵リスト)	(CKL (compromised key list)) (n.) 危殆化鍵を持つユーザーに関する鍵情報のリスト。このリストも CA が提供します。[用語「 危殆化鍵リスト (compromised key list) 」の追加定義。「 CRL (証明書失効リスト) 」も参照してください。
クラシック CoS (classic CoS)	(n.) DN およびターゲットエントリの属性値の 1 つによりテンプレートエントリを識別します。
分類規則 (classification rules)	(n.) 1 つまたは複数のカテゴリにリソースを割り当てるための一連の規則。
クラスローダー (class loader)	(n.) 特定の規則に従って Java クラスを読み込む機能を果たす Java™ テクノロジベースのコンポーネント。

サービスクラス (class of service)	「CoS(サービスクラス)」を参照してください。
CLD(カレンダー検索データベース)	(CLD (Calendar Lookup Database)) (n.) 複数のバックエンドサーバーにカレンダーデータベースが分散されているときに、カレンダーの物理的な位置を調べるプラグイン。Calendar Server には、LDAP CLD プラグインおよびアルゴリズム CLD プラグインが用意されています。
平文 (cleartext)	(n.) 暗号化されていないテキスト。
クライアント証明書認証 (client-certificate authentication)	(n.) サーバーと場合によってはクライアントが相互に認証するための、HTTP over SSL を使用した認証メカニズム。X.509 公開鍵インフラストラクチャーで定義されている規格に適合する公開鍵証明書を使用します。「認証 (authentication)」、「認証局 (CA)」も参照してください。
クライアント規約 (client contract)	(n.) クライアントと EJB™ コンテナ間の通信規則を決め、エンタープライズ Bean を使うアプリケーションのために均一な開発モデルを設定し、クライアントとの関係を統一することによって Bean を効率よく再利用できるように保証する規約。
クライアントの条件付きプロパティ (client conditional properties)	(n.) Portal Server Mobile Access クライアントタイプのプロパティ。このプロパティにより管理者は、所定のクライアントのチャンネルまたはコンテナチャンネルのプロパティを指定できます。
クライアントデータベース (client database)	(n.) Portal Server Mobile Access で使用する、内部と外部のライブラリで構成されるデータベース。内部ライブラリには、すべてのデフォルトモバイルデバイスのデータ定義が収められています。外部ライブラリには、内部ライブラリの定義をオーバーライドする、カスタマイズされたクライアントデータ定義が収められています。
クライアント検出 (client detection)	(n.) ポータルにアクセスする各モバイルデバイスの機能と特性を判断する、Access Manager のプロセス。
Client Editor	(n.) クライアントタイプを作成し、クライアントのプロパティを管理する作業を可能にする、Access Manager のインタフェース。Client Editor インタフェースには、Access Manager コンソールからアクセスできます。
クライアント識別子 (client identifier)	(n.) 接続および接続のオブジェクトを、クライアントに代わって Java Enterprise System メッセージサーバーが管理する状態と関連付ける識別子。
Client Manager	(n.) コンソールからアクセス可能な Access Manager のインタフェース。このインタフェースを使用して、クライアントタイプとプロパティを管理できます。
クライアントプロフィール (client profile)	(n.) 各クライアントを識別する Access Manager のプロファイル。

*クライアントランタイム (*client runtime)	「Java Enterprise System クライアントランタイム」を参照してください。
クライアントサーバーモデル (client-server model)	(n.) ネットワーク接続されたコンピュータがほかのクライアントコンピュータに特定のサービスを提供するコンピュータ処理モデル。例として、DNS のネームサーバーとネームリゾルバのパラダイム、NFS やディスクレスホストなどのファイルサーバーとファイルクライアントの関係などがあります。
*クライアントタイプ (*client type)	(n.) Access Manager クライアントデータベースのエントリ。
clientType	(n.) Access Manager クライアントデータの一意のインデックスとなる名前を参照するプロパティ。
クラスタ	(1)(n.) 高速ネットワークに接続され、全体として1つのサーバー、ブローカ、またはノードであるかのように動作するサーバー、ブローカ、またはノードのグループ。クラスタのサーバー、ブローカ、またはノードの1つで障害が発生した場合、そのサービスを稼働中のサーバー、ブローカ、またはノードにフェイルオーバーできます。「ブローカ (broker)」、「フェイルオーバー (failover)」、「ノード」、「サーバー」を参照してください。
CMP	「コンテナ管理による持続性 (container-managed persistence)」を参照してください。
CMR	「コンテナ管理による関係 (container-managed relationship)」を参照してください。
CMT	「コンテナ管理によるトランザクション (container-managed transaction)」を参照してください。
cn	「共通名属性 (common name attribute)」を参照してください。
CNAME レコード (CNAME record)	(n.) ドメイン名のエイリアスをドメイン名にマップする DNS レコードの一種。
照合順序 (collation order)	(n.) 所定言語の文字のソート方法に関する、言語と文化に固有の情報。この情報には、アルファベットの文字の順序や、アクセント記号付きの文字とアクセント記号のない文字の比較方法などが含まれます。
コレクション (collection)	(n.) ワードリストやファイルのプロパティなど、ドキュメントについての情報を保管するデータベース。検索機能では、コレクションを使用して、指定された検索条件に一致するドキュメントを検索します。
コロケーション (colocation)	(n.) 同じノードに存在するプロパティ。この概念は、パフォーマンスを向上させるために、クラスタ構成時に使用されます。

共存する (colocate)	(v.) リモートプロシージャコールを避けてパフォーマンスを向上させるために、関連するコンポーネントと同じメモリー空間にコンポーネントを配備すること。
列(column)	(n.) データベーステーブル内のフィールド。
comm_dssetup.pl	(n.) 既存の Directory Server を Messaging Server で使用できるように準備する Directory Server 準備ツール。
コメント (comment)	(n.) 「XML (拡張可能マークアップ言語)」ドキュメントで、パーサーに識別するように具体的に指示されない限り、無視されるテキスト。
コメント文字 (comment character)	(n.) 行の先頭に配置して、その行を実行されないコメントに変換する文字。
コミットする (commit)	(1) (v.) 必要なコマンドをデータベースなどのリソースに送信することによって、トランザクションを実行すること。「ロールバック (rollback)」、「トランザクション (transaction)」を参照してください。 (2) (n.) トランザクションに関係したあらゆるリソースに対して、すべての更新が固定化された段階のトランザクションのポイント。
共通ドメイン (common domain)	(n.) 複数のアイデンティティプロバイダで構成される「トラストサークル (circle of trust)」の場合、サービスプロバイダは、主体が使用するアイデンティティプロバイダを決定する方法を必要とします。この機能は、任意の数のドメインネームシステム (DNS) ドメイン全体で作動させる必要があるため、Liberty では、トラストサークル内のアイデンティティプロバイダとサービスプロバイダのすべてに共通なドメインを作成する手法を使用しています。このあらかじめ決定されるドメインを、共通ドメインと呼びます。共通ドメイン内で主体がサービスプロバイダに対して認証されている場合、アイデンティティプロバイダは主体のアイデンティティプロバイダを保存する共通ドメイン cookie を書き込みます。このとき、主体が同じトラストサークル内の別のサービスプロバイダへのアクセスを試みると、サービスプロバイダは共通ドメイン cookie を読み取るので、要求を正しいアイデンティティプロバイダに転送できます。
共通ログファイル形式 (common log file format)	(n.) アクセスログに情報を入力する場合にサーバーが使用する記録形式。この形式は、Web Server を含むすべての主要なサーバー間で同じです。
共通名属性 (common name attribute)	(n.) LDAP ディレクトリ内のエントリによって定義されるユーザーまたはオブジェクトを識別する cn 属性。
Communication Services	(n.) 電子メール、カレンダー、インスタントメッセージング、および存在情報で成る統合情報をエンタープライズカスタマに配信することができる、総合的なメッセージングソリューション。Communication Services のコアソリューションは、Messaging Server、Calendar Server、および Instant Messaging Server で構成されています。

Communications Express	(n.) Web ベースの通信と連携動作を統合したクライアントとして機能するソフトウェア。電子メール、カレンダー、およびアドレスブックの情報にアクセスするエンタープライズユーザーのニーズを満たします。
Compass	(n.) Portal Server 3.0 の検索機能を利用した検索エンジンサービス。この検索エンジンは、Portal Server 6.0 のコアに組み込まれています。「 検索エンジン (Search Engine) 」を参照してください。
Compass Server	(n.) Portal Server 3.0 で一般に使用される、ネットワークリソースへのユーザーアクセスを促進するサーバーテクノロジー。Portal Server 6.0 には、緊密に統合された検索エンジンが搭載されており、Compass Server が Portal Server 3.0 に提供したサービスを提供します。
構成要素(コンポーネント)	<p>(1) (n.) Java Enterprise System に含まれる「システムコンポーネント (system component)」の1つ。</p> <p>(2) (n.) 分散型アプリケーションを構築するときの基本となるソフトウェアロジックの単位。アプリケーションコンポーネントはカスタマイズされて開発され、通常、CORBA や J2EE プラットフォームなどの分散コンポーネントモデルに適合しており、特定のコンピュータ処理機能を実行します。これらのコンポーネントは、単独または組み合わせて「ビジネスサービス (business service)」を提供し、また、「Web サービス (web service)」としてカプセル化できます。</p> <p>(3) (n.) 「J2EE コンポーネント (J2EE component)」を参照してください。</p>
コンポーネント規約 (component contract)	(n.) J2EE コンポーネントとそのコンテナ間の規約。規約には、コンポーネントのライフサイクル管理、さまざまな情報やサービスをコンテナから取得するためにインスタンスが使用するコンテキストインタフェース、各コンテナがコンポーネントのために提供する必要のあるサービスのリストなどが含まれます。
コンポーネント管理によるサインオン (component-managed sign-on)	(n.) リソースへのサインオンに必要なセキュリティ情報がアプリケーションコンポーネントによって提供されるメカニズム。
コンポーネント製品の記述子ファイル (component product descriptor file) の状態 (component state)	<p>(n.) 所定のコンポーネント製品のメタデータを、通常は XML 形式で記述したファイル。</p> <p>(n.) 会議などのカレンダーイベントを記述する属性の集合。WCAP で compstate パラメータを使用すると、fetch コマンドはコンポーネントの状態別にイベントを返します。たとえば、REPLY-DECLINED (出席予定者が会議への出席を拒否した)、REQUEST_NEEDS-ACTION (出席予定者が会議への出欠をまだ決定していない) などの compstate が考えられます。</p>

危殆化鍵リスト
(**compromised
key list**)

「[CKL \(危殆化鍵リスト\)](#)」を参照してください。

算出属性
(**computed
attribute**)

(n.) エントリ自体を保存した属性ではなく、演算結果の通常の属性と一緒にクライアントアプリケーションに返される属性。

会議室
(**conference
room**)

(n.) 管理者または `sysRoomsAdd` 権限を持つほかのユーザーによって設定される、あらかじめ設置されたチャットルーム。管理者または `sysRoomsAdd` 権限を持つほかのユーザーは、会議室の内容の閲覧やアクセスを許可するユーザーを決定できます。

設定
(**configuration**)

(n.) サーバーを調整する、またはアプリケーションのメタデータを提供する設定の集合。通常、特定のアプリケーションの設定は、そのアプリケーションの配備記述子ファイルに保存されています。「[管理コンソール \(admin console\)](#)」、「[配備記述子 \(deployment descriptor\)](#)」も参照してください。

構成管理者
(**configuration
administrator**)

(n.) サーバーソフトウェアトポロジ全体のサーバーおよび構成ディレクトリデータを管理する管理権限を持つユーザー。構成管理者は、サーバーソフトウェアトポロジ全体のすべてのリソースに無制限にアクセスできます。ほかの管理者にサーバーアクセス権を割り当てることができる唯一の管理者です。構成管理者は、管理者のグループとそのメンバーが配置されるまで初期の管理構成を管理します。

設定 **Directory
Server
(Configuration
Directory Server)**

(n.) 単一サーバーまたは一連のサーバーの構成情報を保持する Java Enterprise System Directory Server。

設定ファイル
(**configuration
file**)

(n.) サーバー、アプリケーション、またはソフトウェアコンポーネントの設定パラメータを保持するファイル。

衝突 (**conflict**)

(n.) レプリケーションによってサーバー間でデータを同期する前に、異なるディレクトリサーバーの同じディレクトリデータに変更が加えられたときに生じる状況。サーバーは、同期処理を試みたときにコピーの不整合を検出すると、その衝突を解決するか、エラーをログに記録します。

衝突解決
(**conflict
resolution**)

(n.) 変更情報を解決するために使用する確定的な手順。詳細については、『Java Enterprise System Directory Server 管理ガイド』を参照してください。

輻輳しきい値
(**congestion
thresholds**)

(n.) システム管理者が設定するディスク容量の上限。システムリソースが不足しているときに新しい操作を制限することによって、データベースへの過重負荷を防ぐことができます。

接続
(**connection**)

(1)(n.) 「[リソースマネージャー \(resource manager\)](#)」の場合、リソースマネージャーとの「[セッション](#)」を表すオブジェクト。

	(2) (n.) Java Enterprise System メッセージサーバーへのアクティブな接続。キュー接続またはトピック接続のいずれかです。
接続ファクトリ (connection factory)	(1) (n.) 「リソースマネージャー (resource manager)」の場合、リソースマネージャー接続を作成するために使用されるオブジェクト。 (2) (n.) 提供された JMS 実装をアプリケーションコードが使用できるようにする Java Message Service (JMS) 接続 (TopicConnection または QueueConnection) を作成するために使用するオブジェクト。アプリケーションコードは、Java Naming and Directory Interface™ (JNDI) サービスにより、JNDI サービスを使って接続ファクトリオブジェクトを特定します。
接続ハンドラ (connection handler)	(n.) Directory Proxy Server で、着信クライアント要求をデータビューに分散させるために使用します。接続は、着信 IP アドレスやドメイン名などの条件に従って接続ハンドラに割り当てられます。接続の処理時に、接続ハンドラは接続ポリシーを参照します。
接続ポリシー (connection policy)	(n.) Directory Proxy Server 接続ハンドラによって経路指定された操作を処理する方法を決定するためのポリシールール。リソース制限ポリシーは、接続、要求、リフェラルに割り当てられるリソースを制限します。要求フィルタリングポリシーは、データのアクセス制御を提供します。
接続プール (connection pool)	(n.) 接続の集まり。物理的な接続をキャッシュおよび再利用することで、データベースへのアクセスを効率的にする方法。接続によるオーバーヘッドを回避し、多数のスレッド間で共有する接続を少数に抑えることができます。「JDBC 接続プール (JDBC connection pool)」も参照してください。
コネクタ (connector)	(n.) 「EIS (エンタープライズ情報システム)」への接続を提供するコンテナ用の標準拡張メカニズム。コネクタは、EIS に固有のもので、EIS 接続用のリソースアダプタおよびアプリケーション開発ツールから構成されています。リソースアダプタは、コネクタアーキテクチャーに定義されたシステムレベル規約を使ってコンテナへ接続されます。「リソースアダプタ (resource adapter)」も参照してください。
コネクタアーキ テクチャー (connector architecture)	(n.) J2EE™ アプリケーションと「EIS (エンタープライズ情報システム)」を統合するためのアーキテクチャー。このアーキテクチャーは2つの部分で構成されます。EIS ベンダー提供のリソースアダプタと、このリソースアダプタの接続を許可する J2EE サーバーの2つの部分に分かれています。このアーキテクチャーは、トランザクション、セキュリティ、リソース管理など、リソースアダプタが J2EE サーバーに接続するために必要な規約を定義しています。
Connector for Microsoft Outlook	(n.) Microsoft Outlook を Sun Java Enterprise System でデスクトップクライアントとして使用可能にするプラグイン。
コンソール (console)	「管理コンソール (admin console)」を参照してください。

コンシュームする (consume)	(v.) 送信先から取得するメッセージをメッセージコンシューマが受信すること。
コンシューマ	<p>(1) (n.) サプライヤサーバーからレプリケートされたディレクトリツリーまたはサブツリーを持つサーバー。</p> <p>(2) (n.) 送信先からのメッセージを受信するために使用するセッションによって作成されるオブジェクト (MessageConsumer)。ポイントツーポイント配信モデルでのコンシューマは、受信側ブラウザ (QueueReceiver または QueueBrowser) のいずれかです。パブリッシュ/サブスクライブ配信モデルでのコンシューマは、サブスクライバ (TopicSubscriber) です。</p>
コンシューマディレクトリサーバー (consumer directory server)	<p>(1) (n.) マスターディレクトリサーバーに対するすべての追加、修正、削除操作を参照する読み取り専用のディレクトリサーバー。</p> <p>(2) (n.) 別のディレクトリサーバーから変更を受け取る任意のディレクトリサーバー。「サプライヤディレクトリサーバー (supplier directory server)」を参照してください。</p>
連絡先 (contact)	(n.) インスタントメッセージを送受信する相手側ユーザーまたは LDAP グループの userID (名前)。個人用の連絡先グループに連絡先を追加し、オンラインステータスを監視できます。ほかのインスタントメッセージング環境では、友だちとも呼ばれます。
連絡先グループ (contact group)	(n.) ユーザーが保持する連絡先のリスト。実際のリストは Instant Messaging Server に保管されます。連絡先グループを作成して、論理的な方法で人々を追跡できます。
連絡先リスト (contact list)	(n.) Java Enterprise System Instant Messaging では、連絡先グループすべてのリストです。
コンテナ (container)	<p>(1) (n.) 特定のタイプの J2EE コンポーネントにライフサイクル管理、セキュリティ、配備、実行時サービスを提供します。Application Server には、すべてのタイプの J2EE コンポーネントがあります。「構成要素 (コンポーネント)」も参照してください。</p> <p>(2) (n.) Java Enterprise System Portal Server 6.0 におけるコンテナは、主に子チャネルのコンテンツを集約して自身のコンテンツを生成するチャネルです。Java Enterprise System Directory Server Access Management Edition でのコンテナは、ほかの Directory Server Management Edition のオブジェクトを収容できる組織オブジェクトのタイプとして定義されます。</p>
コンテナエントリ (container entry)	(n.) ディレクトリのサブツリーの最上位を表すエントリ。

コンテナ管理による持続性 (container-managed persistence)	(n.) 「 エンティティ Bean (entity bean) 」 変数とリソースマネージャー間のデータ転送がエンティティ Bean のコンテナで管理されるメカニズム。「 Bean 管理による持続性 (bean-managed persistence) 」も参照してください。
コンテナ管理による関係 (container-managed relationship)	(n.) クラスペアで表される、一方の動作が他方の動作に影響を与えるようなフィールドの関係。
コンテナ管理によるサインオン (container-managed sign-on)	(n.) リソースへのサインオンに必要なセキュリティー情報がコンテナによって提供されるメカニズム。
コンテナ管理によるトランザクション (container-managed transaction)	(n.) エンタープライズ Bean のトランザクション境界設定が、EJB コンテナによる宣言の形で指定され、自動的に制御されるメカニズム。「 エンティティ Bean (entity bean) 」では、コンテナ管理によるトランザクションを使用する必要があります。「 Bean 管理によるトランザクション (bean-managed transaction) 」も参照してください。
コンテンツ (content)	(n.) 「 XML (拡張可能マークアップ言語) 」ドキュメントで、prolog のあとにある部分 (root 要素と、root 要素内に含まれるすべて)。
コンテキスト属性 (context attribute)	(n.) 「 サーブレット (servlet) 」と関連付けられたコンテキストにバインドされたオブジェクト。
コンテキストルート (context root)	(n.) Web アプリケーションの ドキュメントルート (document root) にマッピングされる名前。
コントロール記述子 (control descriptor)	(n.) エンタープライズ Bean トランザクションおよびセキュリティープロパティーだけでなく、Bean メソッドの個々のプロパティーオーバーライド (オプション) を指定できるようにする一連のエンタープライズ Bean 設定エントリ。
コントローラ (controller)	(n.) Identity Synchronization for Windows コネクタコンポーネントは、エージェントおよびアクセサコンポーネントとのインタフェースです。コントローラは、同期ユーザーリストにおけるユーザーのメンバーシップの判断、等価なユーザーエンティティの検索とリンク、現在のエンティティとオブジェクトキャッシュに格納された以前のバージョンとを比較することによるユーザーに対する変更の検出など、重要な同期関連タスクを実行します。コントローラは、動作についてログメッセージで多く参照されます。
対話状態 (conversational state)	(1) (n.) 同一のクライアントと何度も対話した結果、オブジェクトの状態が変更される状態。「 持続状態 (persistent state) 」も参照してください。

(2)(n) 「セッション Bean (session bean)」のフィールド値に、Bean のフィールドから到達可能なオブジェクトの推移閉包を加えたもの。Bean の推移閉包は、Java プログラミング言語の直列化プロトコル、つまり Bean インスタンスを直列化して格納されるフィールドによって定義されます。

cookie

(n.) 呼び出し側である Web ブラウザに対して送信され、その後、そのブラウザから呼び出すたびに取り出される情報の小さなコレクション。サーバーは、cookie によって、同じクライアントからの呼び出しであるかどうかを認識できます。cookie はドメイン特有の要素で、アプリケーションとサーバー間のほかのデータ交換と同じ Web サーバーセキュリティー機能の利点を生かすことができます。cookie を受け入れれば Web ページを高速に読み込むことができ、ユーザーマシンのセキュリティーを脅かすことはありません。

**連携サーバー
(cooperating
server)**

(n.) ユーザーのサーバーと通信することを望むサーバーおよびユーザーのサーバーが通信することを望むサーバー。coserver と呼ばれます。それぞれの連携サーバーには、文字と数字で構成された文字列であるシンボリック名が与えられます。たとえば、coservern などがあり、この場合の *n* は番号を表します。

**CORBA (共通オブ
ジェクトリクエ
ストブローカ
アーキテク
チャー)**

(CORBA (common object request broker architecture))(n.) 「OMG」で指定された、オブジェクト指向分散コンピューティングの言語に依存しない標準アーキテクチャー定義。

**コアサービス
(core service)**

(n.) サポートサービスや補助サービスに対し、Java Enterprise System サーバーによって提供される基本的な機能を定義する、1 つ以上の主要なサービス。

**CoS (サービスク
ラス)**

(CoS (class of service))(n.) エントリ間で属性を共有する手法。

**CoS 定義エン
トリー (CoS definition
entry)**

(n.) ユーザーが使用している CoS のタイプを識別するエントリー。CoS 定義エントリーは、LDAP サブエントリーとして、影響を与える分岐の下に保存されます。

coserver

「[連携サーバー \(cooperating server\)](#)」を参照してください。

**CosNaming プロ
バイダ
(CosNaming
provider)**

(n.) IIOP アプリケーションクライアントにアクセスすることができるグローバルな JNDI 名前空間をサポートするために、Java Enterprise System Application Server には、CORBA 参照 (リモート EJB 参照) のバインドをサポートする J2EE ベースの CosNaming プロバイダが組み込まれています。

**COSNaming サー
ビス
(COSNaming
Service)**

(n.) IIOP ベースのネームサービス。

**CoS テンプレ
ートエン
トリー (CoS
template entry)**

(n.) 共有属性値のリストが記述されたエントリー。

CRAM-MD5	(n.) RFC 2195 に記述されている標準化過程にある軽量な認証方式。ネットワークへの不正な侵入からユーザーのログインパスワードだけを保護する場合に、TLS (SSL) の代わりに使用できます。TLS より高速ですが、やや強度が落ちます。
crawler	「 ロボット (robot) 」を参照してください。
生成メソッド (create method)	(n.) 「 ホームインタフェース (home interface) 」で定義されたメソッド。「 エンタープライズ Bean (enterprise bean) 」を作成するために、クライアントによって呼び出されます。
CRL (証明書失効リスト)	(CRL (certificate revocation list)) (n.) 証明書発行局が発行するリスト。クライアントユーザーまたはサーバーユーザーのどちらも信頼することができない証明書を示します。この場合、証明書は無効になっています。「 CKL (危殆化鍵リスト) 」も参照してください。
cron job	(n.) UNIX のみ。指定した時間に cron デーモンによって自動的に実行されるタスクです。
CSAPI (Calendar Server アプリケーションプログラミングインタフェース)	(CSAPI (Calendar Server application programming interface)) (n.) Calendar Server の機能セットの変更および拡張機能を提供するプログラミングインタフェース。CSAPI モジュールは、Calendar Server の起動時に cal/bin/plugins ディレクトリから読み込まれるプラグインです。
CSS	(1) (Cascading style sheet) (n.) HTML ドキュメントと XML ドキュメントで使用されるスタイルシート。ブラウザやその他の表示メカニズムに指示を行う特定のタグでマーク付けされたすべての要素に対してスタイルを追加します。
CTS (互換性テストスイート)	(Compatibility test suite) (n.) J2EE 製品が J2EE プラットフォーム仕様に準拠していることを検証する互換性テストのスイート。
CUA (カレンダーユーザーエージェント)	(CUA (Calendar user agent)) (n.) カレンダークライアントが Calendar Server へのアクセスに使用するアプリケーション。

D

DAP(ディレクトリアクセスプロトコル)	(DAP (directory access protocol)) (n.) LDAP の基礎となった ISO/ITU-T X.500 プロトコル。
データ (data)	(n.) 「XML(拡張可能マークアップ言語)」ストリームの要素のコンテンツ。一般的に要素にサブ要素が含まれないときに使用されます。サブ要素が含まれる場合には、通常「コンテンツ」という語が使用されます。XML 構造内のテキストが単純な要素に含まれている場合、およびサブ要素を持つ要素にデータがほとんどまたはまったく含まれない場合、通常は XML ドキュメントではなく、XML データであると見なされます。
データアクセスロジック (data access logic)	(n.) データソースとの対話を伴うビジネスロジック。
データベース (database)	(n.) リレーショナルデータベース管理システム (RDBMS) の一般名。関連する組織化された大量のデータの作成および操作が可能なソフトウェアパッケージです。「スキーマ」も参照してください。
データベース接続 (database connection)	(n.) データベースまたはほかのデータソースとの通信リンク。コンポーネントは、複数のデータベース接続を同時に作成および操作して、データにアクセスできます。
データベースワイヤプロトコル (database wire protocol)	「データ冗長ユニット (DRU)」を参照してください。
データ冗長ユニット (DRU)	(data redundancy unit (DRU)) (n.) アクティブノードとスペアノードの半分、およびデータの完全なコピー1つで構成される HADB ノードのセット。HADB は、相互にミラー化された2つの DRU で構成されます。障害耐性を確保するには、1つの DRU をサポートする各コンピュータがそれぞれ独立して電源、処理装置、およびストレージを完全にサポートしている必要があります。「HADB ノード (HADB node)」

「アクティブノード (active node)」、「スペアノード (spare node)」、および「ミラーノード (mirror node)」も参照してください。

データサービス (data service)	(n.) エンドユーザーに関するデータのクエリーおよび修正をサポートする Web サービス。データサービスの例として、ユーザーのプロファイル情報 (名前、住所、電話番号など) をホストし、公開する Web サービスがあります。
データソース (data source)	(1) (n.) データベースなどの、データのソースへのハンドル。データソースは、Application Server に登録された後、接続を確立してデータソースと対話できるようにするために、プログラムによって取得されます。データソース定義により、データソースへの接続方法を指定します。 (2) (n.) Directory Proxy Server がアクセスするリポジトリ。リポジトリには、LDAP ディレクトリ、JDBC 準拠データベース、LDIF フラットファイルなどがあります。
データソースオブジェクト (data source object)	(n.) そのデータソースが実際には何を表すかを識別し、説明する一連のプロパティを持ったオブジェクト。
データソースプール (data source pool)	(n.) 同等のデータを保持するデータソースのセット。データソースプールでは、Directory Proxy Server の負荷分散およびフェイルオーバー管理を提供します。
データストア (data store)	(1) (n.) ディレクトリ情報を保管するストア。通常は「DIT」全体の情報を保管します。 (2) (n.) Message Queue ブローカに必要な情報 (持続性サブスクリプション、送信先のデータ、持続メッセージ、および監査データ) が恒久的に格納されるデータベース。
データビュー (data view)	(n.) DN ベースのルーティングを使用して、Directory Proxy Server 接続ハンドラからデータソースプールへの接続を経路指定します。
DC ツリー (ドメインコンポーネントツリー)	(DC tree (domain component tree)) (n.) DNS ネットワーク構文を反映する「DIT」。DC ツリー内の識別名には、cn=billbob,dc=bridge,dc=net,o=internet などがあります。
DDP	(Document-driven programming) (n.) アプリケーションを定義する「XML (拡張可能マークアップ言語)」の使用方法。
宣言 (declaration)	(n.) XML であることを宣言する、「XML (拡張可能マークアップ言語)」ドキュメントの最初の部分。最小の宣言は、<?xml version="1.0"?> です。宣言はドキュメントの「prolog」の一部です。
宣言によるセキュリティ (declarative security)	(n.) セキュリティープロパティをコンポーネントの配備記述子内で宣言し、コンポーネントのコンテナ (例: Bean のコンテナやサーブレットエンジン) にセキュリティを暗黙的に管理させること。このタイプのセキュリティでは、プログラム

	の制御が不要です。「プログラムセキュリティー (programmatic security)」とは反対の概念です。「コンテナ管理による持続性 (container-managed persistence)」も参照してください。
宣言によるトランザクション (declarative transaction)	「コンテナ管理によるトランザクション (container-managed transaction)」を参照してください。
復号化 (decryption)	(n.) 暗号化された情報を判読可能にするプロセス。「暗号化 (encryption)」も参照してください。
デフォルトカレンダー (default calendar)	(n.) Calendar Express にログインしたときに最初に表示されるカレンダー。通常、デフォルトカレンダーのカレンダー ID は、ユーザーのユーザー ID と同じです。たとえば、jdoe@example.com のデフォルトカレンダー名は jdoe です。
デフォルトインデックス (default index)	(n.) Directory Server のインストール時に、データベースインスタンスごとに作成される 1 組のインデックス。Java Enterprise System Directory Server がインストールされると、データベースインスタンスごとに 1 組のデフォルトインデックスが作成されません。詳細については、『Java Enterprise System Directory Server 管理ガイド』を参照してください。
連携の解除 (defederation)	(n.) 「 連携の終了 (federation termination) 」を参照してください。
定義エントリ (definition entry)	「 CoS 定義エントリ (CoS definition entry) 」を参照してください。
デフラグ (defragmentation)	(n.) 小さなメッセージつまり断片の集まりに分割されたサイズの大きなメッセージを、元の大きさのメッセージに組み立て直す MIME の機能。各断片の Message Partial Content-Type ヘッダーフィールドには、断片を 1 つのメッセージに再組立するために使用する情報が含まれています。「 断片化 (fragmentation) 」も参照してください。
Delegated Administrator	(n.) ドメイン管理者が、ホストドメインのディレクトリにユーザーを追加したり、ホストドメインのディレクトリのユーザーやグループを変更したりできるようにする一連の GUI および CLI のインターフェース。
Delegated Administrator コンソール (Delegated Administrator Console)	(n.) Web ブラウザベースのソフトウェアコンソール。ドメイン管理者はこれを使用して、ホストドメインに対してユーザーやグループの追加または変更を行うことができます。また、エンドユーザーは、これを使用して、自分のパスワードの変更、メッセージ転送規則の設定、不在返信規則の設定、メールリスト登録の一覧表示なども行えます。
メッセージングおよびコラボレーション用 Delegated Administrator (delegated administrator for messaging and collaboration)	(n.) ドメイン管理者がホストドメインに対してユーザーやグループの追加または変更を行うために使用する一連のインターフェース (GUI とユーティリティー)。

Delegated Administrator Server	(n.) ホストドメインによるディレクトリへのアクセス制御を処理するデーモンプログラム。
委譲 (delegation)	(1) (n.) オブジェクトの構成を実装方法として使うオブジェクト指向技術の1つ。ある処理の結果に責任を持つオブジェクトが、別のオブジェクトに実装を委譲します。たとえば、クラスローダーは一部のクラスのロードを親に委譲することが多くあります。「 クラスローダー (class loader) 」も参照してください。
メッセージを削除する (delete message)	(v.) 削除するメッセージにマークを付けること。削除したメッセージは、別の処理で消去またはパージするまで、メッセージストアからは削除されません。「 メッセージをパージする (purge message) 」、「 メッセージを破棄する (expunge message) 」も参照してください。
配信 (delivery)	「 メッセージ配信 (message delivery) 」を参照してください。
配信モード (delivery mode)	(n.) メッセージングの信頼性を示すモード。持続配信モードでは、メッセージが1度かぎりで配信され確実にコンシュームされることを保証し、非持続配信モードでは、メッセージが、多くて1度、配信されることを保証します。
配信モデル (delivery model)	(n.) メッセージの配信方法を示すモデル。このモデルは、ポイントツーポイントまたはパブリック/サブスクライブのいずれかです。Java™ Message Service (JMS) には、それぞれ特定のクライアントランタイムオブジェクトと特定の送信先タイプ(キューやトピック)を使用する個別のプログラミングドメイン、およびユニファイドプログラミングドメインがあります。
配信ポリシー (delivery policy)	(n.) 複数のメッセージコンシューマを登録した場合に、キューがメッセージをルートする方法を細かく定めた仕様。ポリシーには、シングル、ファイルオーバー、ラウンドロビンがあります。
配信ステータス通知 (delivery status notification)	(n.) 受信側に向かっているメッセージについてのステータス情報を示すメッセージ。たとえば、ネットワークが停止しているために配信が遅れていることを示すメッセージがあります。
サービス拒否攻撃 (denial of service attack)	(n.) 個人が意図的にまたは誤ってメッセージを大量に送信したために、メールサーバーが処理不能になる状態。サーバーのスループットに著しい悪影響を与えたり、サーバー自体が過負荷状態になって機能しなくなったりすることがあります。
拒否フィルタ (Deny filter)	(n.) POP、IMAP、またはHTTP サービスの中で、1つ以上のサービスへのアクセスが拒否されるクライアントを識別する、Java Enterprise System Messaging Server のアクセス制御規則。「 許可フィルタ (Allow filter) 」も参照してください。
デプロイヤ (deployer)	(n.) J2EE モジュールとアプリケーションを運用環境にインストールする担当者。
配備 (deployment)	(1) (n.) 運用環境にソフトウェアをインストールする処理。

(2) (n.) Java Enterprise System ソリューションのライフサイクルプロセスの1段階。ここでは、配備シナリオが配備設計に変換され、実装、プロトタイプ作成、および本稼働環境での展開が行われます。このプロセスの最終結果は配備または配備ソリューションとも呼ばれます。

配備アーキテクチャ
(**deployment architecture**)

(n.) 「**論理アーキテクチャ (logical architecture)**」から物理的なコンピューティング環境への対応を描いた全体的な構図。物理的な環境は、イントラネットまたはインターネット環境にあるコンピュータ、それらのコンピュータ間のネットワークリンク、ソフトウェアをサポートするために必要なほかの物理的デバイスなどで構成されます。

配備記述子
(**deployment descriptor**)

(n.) 各モジュールおよびアプリケーションに備わっている、それらの配備方法を記述した「**XML (拡張可能マークアップ言語)**」ファイル。配備記述子は、配備ツールに、特定のコンテナオプションでモジュールまたはアプリケーションの配備を指示し、デプロイヤーが解決する必要のある特定の設定要件を示しています。「**メタデータ**」を参照してください。

配備シナリオ
(**deployment scenario**)

(n.) Java Enterprise System ソリューションの「**論理アーキテクチャ (logical architecture)**」、およびそのソリューションがビジネスニーズを満たすために満たす必要のあるサービス品質要件。サービス品質要件には、パフォーマンス、可用性、セキュリティ、保守性、スケーラビリティ/潜在容量などに関する要件があります。配備シナリオは、配備設計の開始点です。

深さ (**depth**)

(n.) サイトの検索エンジンの開始点から続くリンク数。サイトを定義する場合は、ロボットがその開始点からたどることのできるリンク数を定義し、検索の深さを制限します。

エイリアスを参照解除する
(**dereferencing an alias**)

(v.) バインドまたは検索で、ディレクトリサービスがエイリアス識別名をエントリの実際の識別名に変換するように指定すること。

デスクトップ
(**Desktop**)

「**Portal Server Desktop**」を参照してください。

送信先
(**destination**)

(n.) 作成されたメッセージを送信する先のメッセージサーバーの物理的な位置。メッセージはそこでルーティング処理されたあと、コンシューマに配信されます。この物理的な送信先(「**キュー (queue)**」または「**トピック (topic)**」)は、「**管理対象オブジェクト (administered object)**」によって特定され、カプセル化されます。クライアントは、管理対象オブジェクトを使用して、クライアントがメッセージを作成する先の、またはクライアントがメッセージを消費する元となる送信先を指定します。「**ポイントツーポイント配信モデル (point-to-point delivery model)**」、「**パブリッシュ/サブスクライブ配信モデル (publish/subscribe delivery model)**」も参照してください。

送信先リソース (destination resource)	(n.) トピック送信先またはキュー送信先を表すオブジェクト。キューの読み出しと書き込み、トピックのパブリッシュとサブスクライブを行うときにアプリケーションが使用します。アプリケーションコードでは、Java Naming and Directory Interface™ (JNDI) サービスを使用し、JNDI Name によって Java Message Service (JMS) のリソースオブジェクトを特定します。
開発 (development)	(n.) Java Enterprise System ソリューション配備プロセスに含まれるタスク。このタスクにより、「 配備アーキテクチャー (deployment architecture) 」のカスタムコンポーネントをプログラムしてテストします。
デバイス検出 (device detection)	「 クライアント検出 (client detection) 」を参照してください。
デバイス情報 (device information) DHCP	(n.) Portal Server Mobile Access で使用するデバイス固有のクライアントデータ。 (DHCP (dynamic host configuration protocol)) (n.) インターネット提案標準プロトコル。システムが、ネットワーク内の個々のコンピュータに IP アドレスを動的に割り当てることができるようにします。「 IP アドレス (IP address) 」も参照してください。
ダイジェスト認証 (digest authentication)	(n.) ユーザー名とパスワードを平文で送信しなくてもユーザーを認証することができる認証タイプ。Web アプリケーションでは、HTTP 要求メッセージとともにメッセージダイジェストをサーバーに送信することで、Web サーバーに対して自分自身を認証します。HTTP 要求メッセージとクライアントのパスワードを連結したもののに対して一方向のハッシュアルゴリズム (MD5 と呼ばれる) を使用して、ダイジェストを計算します。ダイジェストは、通常は HTTP 要求と比べて非常に小さく、パスワードを含みません。サーバーはダイジェスト認証プラグインを使用して、クライアントから送られてきたダイジェスト値を比較します。
DIGEST-MD5	(n.) CRAM-MD5 より安全性の高い、標準化過程にある軽量認証メカニズム。RFC 2831 に記述されています。RFC 2831 には、TLS (SSL) のような設定の手間をかけずに接続全体を保護するオプションも記述されています。
デジタル署名 (digital signatur)	(n.) メッセージと署名者の両方の認証に使用される電子的なセキュリティーメカニズム。
指示 (directive)	(n.) 検索エンジンの文。固有の書式を使用して、ロボットアプリケーション関数などの関数を呼び出し、パラメータブロックの関数にパラメータを渡します。たとえば、次の指示は <code>enumerate-urls</code> 関数を呼び出して、 <code>max</code> および <code>type</code> パラメータを渡します。 <code>Enumerate fn=enumerate-urls max=1024 type=text/html</code>
ディレクトリ (directory)	(n.) データの書き込みよりもデータの読み込みに対して最適化された特殊なデータベース。ほとんどのディレクトリ、業界標準プロトコルの LDAP (Lightweight Directory Access Protocol) をベースにしています。

ディレクトリアクセスプロトコル (directory access protocol)	「DAP (ディレクトリアクセスプロトコル)」を参照してください。
ディレクトリコンテキスト (directory context)	(n.) メッセージストアへのアクセスに対して、ユーザーとパスワードの認証に使用するエントリの検索を開始するディレクトリツリー情報内のポイント。「ベース DN (ベース識別名)」も参照してください。
ディレクトリ配備 (directory deployment)	(n.) Application Server において、アーカイブファイルではなく展開されたディレクトリ形式で、パッケージ化されていない J2EE アプリケーションやモジュールを配備すること。
ディレクトリエントリ (directory entry)	(n.) 識別名で特定されるディレクトリ属性とその値のセット。各エントリには、エントリが記述するオブジェクトの種類を指定し、エントリに含まれる属性のセットを定義するオブジェクトクラス属性が含まれています。
ディレクトリ情報ツリー (directory information tree)	「DIT」を参照してください。
ディレクトリ検索 (directory lookup)	(n.) ユーザーやリソースの名前またはその他の特性に基づき、ディレクトリ内で特定のユーザーやリソースに関する情報を検索するプロセス。
ディレクトリマネージャー (Directory Manager)	(1) (n.) ディレクトリサーバーデータベースの管理権限を持つユーザー。アクセス制御は、このユーザーには適用されません。ディレクトリマネージャーはディレクトリのスーパーユーザーと考えることができます。 (2) (n.) UNIX システムの root ユーザーに対応する、権限を持ったデータベース管理者。アクセス制御はディレクトリマネージャーには適用されません。
ディレクトリスキーマ (directory schema)	(n.) ディレクトリに保存できるデータを定義する一連の規則。
Directory Server	(n.) 「ライトウェイトディレクトリアクセスプロトコル (Lightweight Directory Access Protocol)」の Java Enterprise System バージョン。Application Server のすべてのインスタンスは、Directory Server を使用してユーザーおよびグループに関する情報などの共有サーバー情報を保存します。
Directory Server Access Management Edition	(n.) ユーザーおよびサービスの管理、認証サービスとシングルサインオンサービス、ポリシー管理、ロギングサービス、デバッグユーティリティ、およびクライアントによる Portal Server のサポートを実現する一連のインタフェース。

ディレクトリ サーバー (directory server)	(1)(n.) 論理的に一元管理されたりポジトリから、組織内部のユーザーおよびリソースに関する情報を提供するサーバー。「 LDAP (Lightweight Directory Access Protocol) 」、「 Directory Server Access Management Edition 」も参照してください。 (2)(n.) LDAP を基にした Java Enterprise System ディレクトリサービス。
ディレクトリ サービス (directory service) Directory Service Control Center	(n.) 組織内部のユーザーおよびリソースについて説明した属性ベースの情報を管理するためのデータベースアプリケーション。 (n.) Directory Server と Directory Proxy Server を管理するための、ブラウザベースの GUI。
ディレクトリ サービス管理者 (Directory Service Manager)	(n.) Directory Service Control Center を使用して複数の Directory Server と Directory Proxy Server 上のサーバー設定およびデータを管理する LDAP スーパーユーザー。
ディレクトリ同 期 (directory synchronization)	(n.) MTA ディレクトリキャッシュを、ディレクトリサービスに保存された現在のディレクトリ情報で同期するプロセス。「 MTA ディレクトリキャッシュ (MTA directory cache) 」も参照してください。
切断状態 (disconnected state)	(n.) メールクライアントが、サーバーに接続して選択したメッセージのキャッシュコピーを作成し、その後サーバーとの接続を切断する状態。
ディス パッチャー (Dispatcher)	(n.) 定義済み TCP ポートへの接続要求を処理する MTA コンポーネント。ディスパッチャーは、指定のサービスに対する責任を複数のマルチスレッドサーバーで共有すること許可する、マルチスレッド接続ディスパッチエージェントです。ディスパッチャーを使用すると、複数のマルチスレッド SMTP サーバープロセスを同時実行できるようになります。
表示プロファイル (display profile)	(n.) Java Enterprise System Portal Server のプロバイダとチャンネルを定義および設定するために使用する一連の XML ドキュメント。
識別名 (distinguished name)	「 DN 」を参照してください。
分散可能 セッション (distributable session)	(n.) クラスタ内のすべてのサーバー間で分散できるユーザーセッション。
分散認証 UI サー バー (Distributed Authentication UI Server)	(n.) Access Manager のサブコンポーネント。Access Manager 配備内の 2 つのファイアウォール間でセキュリティ保護された分散認証が可能になります。分散認証 UI サブコンポーネントは、Access Manager 配備のセキュリティ保護されていない (DMZ) レイヤー内の 1 つまたは複数のサーバーにインストールします。このサブコンポーネントは、エンドユーザーと 2 番目のファイアウォールの背後にある Access Manager インスタンス間の認証インタフェースとして機能するので、Access Manager サービスの URL がエンドユーザーに公開されなくなります。

分散エンタープライズアプリケーション (distributed enterprise application)	(n.) ロジックがネットワーク環境またはインターネット環境を対象にしている(分散型)、適用範囲とスケールが本稼動環境またはサービスプロバイダのニーズを満たしている(エンタープライズ向け)アプリケーション。アプリケーションのコンポーネントは、通常は異なるプラットフォームにある、別個の実行時環境で実行されます。一般的な分散アプリケーションは、2層(クライアント-サーバー)、3層(クライアント-ミドルウェア-サーバー)、および複数層(クライアント-複数のミドルウェア-複数のサーバー)です。
分散インデックス化 (distributed indexing)	(n.) 検索エンジンの別々のロボットを割り当てて、ネットワークの異なる部分をインデックス化するプロセス。分散インデックス化により、各ロボットの負荷を低減させます。その後、1つの検索エンジンが、それぞれのロボットからリソース記述をインポートすることにより、異なるロボットすべてからすべてのリソース記述を収集できます。
分散トランザクション (distributed transaction)	(n.) 別個のサーバー上に配備されている複数の異種データベースに適用可能な1つのトランザクション。
配布リスト (distribution list)	「メールリスト (mail list)」を参照してください。
配布リスト所有者 (distribution list owner)	「メールリスト所有者 (mail list owner)」を参照してください。
DIT	(DIT (directory information tree)) (n.) ディレクトリに保管された情報の論理的な表現。DITは、ほとんどのファイルシステムで使用されるツリーモデルを反映し、ツリーのルートを経階層の最上位に表示します。
DN	(DN (distinguished name)) (n.) エントリの名前およびディレクトリ内の位置を示す文字列表現。
DN属性 (DN attribute)	(n.) 関連するユーザー、グループ、オブジェクトの識別情報を含むテキスト文字列。
DNS (ドメインネームシステム)	(DNS (domain name system)) (n.) ネットワークのマシンで使用される体系。00.120.000.168などのIPアドレスとwww.example.comなどのホスト名を関連付けます。クライアントは通常、DNSを使用して接続先のサーバーのIPアドレスを見つけます。多くの場合DNSのデータは、NISやUNIXシステムの/etc/hostsファイルなどのローカルテーブル内で増加します。「IPアドレス (IP address)」も参照してください。
DNSエイリアス (DNS alias)	(n.) DNSサーバーが認識し、異なるホストを指すホスト名。DNSエイリアスは、DNS CNAMEレコードとして実装されます。マシンの実際の名前は1つだけですが、1つまたは複数のエイリアスを持つことができます。たとえば、www.example.comというエイリアスが、現在サーバーが置かれているrealthing.example.comという実際のマシンを指すようにすることができます。

DNS データベース (DNS database)

(n.) ドメイン名 (ホスト名) およびそれに対応する IP アドレスのデータベース。

DNS ドメイン (DNS domain)

(n.) 共通のサフィックス (ドメイン名) の付いたホスト名を持つコンピュータのグループ。構文的には、ピリオド (ドット) で区切られた一連の名前 (ラベル) から成るインターネットドメイン名です。たとえば `corp.mktng.example.com` などです。「[ドメイン \(domain\)](#)」も参照してください。

DNS スプーフィング (DNS spoofing)

(n.) DNS サーバーが不正情報を提供するように仕向けるネットワーク攻撃の形態。

ドキュメント (document)

(1) (n.) ネットワーク上の文書で、ほとんどの場合は Web ページまたはワードプロセッシング文書です。テキストファイルやスプレッドシートの場合もあります。検索エンジンによってインデックス化されるリソースの一般名です。

(2) (n.) 1 つ以上の要素にテキストとサブ要素が混在する「[XML \(拡張可能マークアップ言語\)](#)」構造。「[データ \(data\)](#)」も参照してください。

ドキュメントオブジェクトモデル (DOM)

(n.) 「[XML \(拡張可能マークアップ言語\)](#)」ドキュメントにツリー構造としてアクセスして操作するための API。DOM は、プラットフォームや言語に依存しないインタフェースを提供します。このインタフェースを使用すると、プログラムやスクリプトから XML ドキュメントのコンテンツや構造に動的にアクセスして修正できます。

ドキュメントルート (document root)

(1) (n.) Java Enterprise System Web Server にアクセスするユーザーに表示されるファイル、イメージ、データを格納した、サーバーマシン上のディレクトリ。

(2) (n.) サーバーにアクセスするユーザーに対して表示するファイル、イメージ、データを格納した、サーバーマシン上のディレクトリ。

(3) (n.) 主ドキュメントディレクトリ。仮想サーバーの全ファイルを格納してリモートクライアントに提供するための中心的なディレクトリです。

ドキュメントタイプ定義 (document type definition)

「[DTD \(ドキュメントタイプ定義\)](#)」を参照してください。

ドメイン (domain)

(1) (n.) たとえば、`example.com` や `host.example.com` などのドメイン名を所有する企業または組織を特定する、完全修飾ドメイン名の最後の部分。

(2) (n.) 単一コンピュータシステムの管理制御下にあるリソース。

(3) (n.) JMS メッセージング処理をプログラミングするために、Java Message Service (JMS) クライアントが使用するオブジェクトの集まり。ポイントツーポイント配信モデル用とパブリッシュ/サブスクライブ配信モデル用の 2 つのプログラミングドメインがあります。

(4) (n.) Sun Java System Application Server の機能の 1 つ。この機能により、複数の管理ユーザーがそれぞれの専用ドメインを作成および管理できます。ドメインは、インストールされたバイナリの共通セットを使用して、1 つのシステムで作成される一連のインスタンスです。

Domain Administration Server

(n.) Domain Administration Server は、Application Server の管理タスクすべてを処理する、特別設計の Application Server インスタンスです。Application Server 設定情報の中央リポジトリの保守と更新を行います。Domain Application Server が動作していない場合は、管理タスクを実行できません。

ドメイン管理者 (domain administrator)

(n.) Delegated Administrator for Messaging and Collaboration の GUI または CLI を使用して、ホストドメイン内のメールユーザー、メールリスト、およびファミリーアカウントの作成、変更、および削除を行うための管理権限を持つユーザー。デフォルトでは、このユーザーがトポロジ内のすべてのメッセージサーバーに対するメッセージストア管理者となります。

ドメインエイリアス (domain alias)

(n.) 別のドメインを指すドメインエントリ。ホストドメインはエイリアスを使用することにより、複数のドメイン名を持つことができます。

ドメインディレクトリ (domain directory)

(n.) Application Server 「ドメイン (domain)」のディレクトリ。1 つ以上の「インスタンスディレクトリ (instance directory)」を含みます。これは Application Server における「サーバールート (server root)」の呼称です。

ドメインホスティング (domain hosting)

(n.) ドメインをホストするプロセス。共有 Messaging Server 上で 1 つまたは複数のドメインをホストする機能です。たとえば、example.com および example.org というドメインの両方を、example.com メールサーバー上でホストできます。ユーザーは、ホストドメインにメールを送信し、そのドメインからメールを受信します。メールサーバーの名前は、メールアドレスに含まれません。

ドメイン名 (domain name)

(1) (n.) 電子メールアドレス内で使用されるホスト名。
(2) (n.) 管理組織を定義する一意の名前。ドメインにはほかのドメインを含めることができます。ドメイン名は右から左の方向に解釈されます。たとえば、example.com は、Example Company のドメイン名であり、かつトップレベルの com ドメインのサブドメインです。example.com ドメインをさらに corp.example.com などのサブドメインに分割することもできます。「**ホスト名 (host name)**」、「**完全修飾ドメイン名 (fully qualified domain name)**」も参照してください。

ドメインネームシステム (domain name system)

「DNS (ドメインネームシステム)」を参照してください。

ドメイン組織 (domain organization)

(n.) 組織ツリー内のホストドメインの下にあるサブドメイン。ドメイン組織は、企業内でユーザーとグループのエントリを部門別に編成する場合に有用です。

ドメイン部分 (domain part)	(n.) 電子メールアドレスの@記号の右側にある部分。たとえば、example.com は、電子メールアドレス jdoe@example.com のドメイン部分です。
ドメイン制限容量 (domain quota)	(n.) 電子メールメッセージ用にドメインに割り当てられる容量。容量の大きさは、システム管理者が設定します。
ドメインレジストリ (domain registry)	(n.) サーバーのインストールで作成および設定されたすべてのドメインについて、ドメイン固有の情報を登録した1つのデータ構造。ドメインの名前、位置、ポート、ホストなどの情報が登録されています。
ドメイン書き換え規則 (domain rewrite rules)	「書き換え規則 (rewrite rule)」を参照してください。
ドメインテンプレート (domain template)	(n.) アドレスのホスト部分とドメイン部分の書き換え方法を定義する、書き換え規則の一部。テンプレートには、完全に静的なホストアドレスとドメインアドレス、または単一フィールド置換文字列、あるいはその両方を定義できます。
二重障害 (double failure)	(n.) HADB の1つ以上のミラーノードペアで発生する同時障害。「HADB」、「HADB ノード (HADB node)」、「アクティブノード (active node)」、「スペアノード (spare node)」、「ミラーノード (mirror node)」、および「データ冗長ユニット (DRU)」を参照してください。
ドロップワード (drop word)	「ストップワード (stop word)」を参照してください。
DRU	「データ冗長ユニット (DRU)」を参照してください。
DSA	(DSA (directory system agent)) (n.) Directory Server での X.500 用語。
DSCC	「Directory Service Control Center」を参照してください。
DSE (ディレクトリサーバーエントリ)	(DSE (directory server entry)) (n.) サーバー固有の追加情報が関連付けられているエントリ、または DSA 固有エントリ。Root DSE やスキーマ DSE などの DSE には、それぞれのサーバーで異なる属性を持ちます。
DSP (デジタル信号処理)	(DSP (digital signal processing)) (n.) アナログからデジタルへの信号変換。音声アクセス用の電話を使用して Portal Server ソフトウェアにアクセスするには、DSP cvard が必要です。
DSML (ディレクトリサービスマークアップ言語)	(DSML (directory service markup language)) (n.) ディレクトリサービスを XML で表現するのを可能にする、XML マークアップ言語を表現する場合のドキュメント形式ファミリ。Java Enterprise System Directory Server 5.2 は、DSML 標準のバージョン 2 (DSMLv2) に準拠しています。

DSN	(n.) 「 配信ステータス通知 (delivery status notification) 」を参照してください。
dsservd	(n.) ディレクトリ情報が格納されたデータベースにアクセスし、LDAP プロトコルを使用してディレクトリクライアントと通信するデーモン。
dssetup	(n.) 既存の Directory Server を Java Enterprise System Messaging Server で使用できるように準備する、Java Enterprise System Directory Server 準備ツール。
DTD (ドキュメントタイプ定義)	(DTD (document type definition)) (n.) XML 規格で指定された、「 XML (拡張可能マークアップ言語) 」ドキュメントの prolog で省略可能な部分。DTD では、XML ドキュメントに配置する有効なタグおよびタグシーケンスについて制約を指定します。DTD には多くの欠点があるものの、それをきっかけにさまざまなスキーマが提案されるようになりました。たとえば DTD エントリ <code><!ELEMENT username (#PCDATA)></code> は、username という XML 要素に解析対象文字データ (つまりテキスト単独) が含まれ、その他の構造要素がないことを表します。DTD には、現在のファイルで定義されたローカルサブセットと、外部 DTD ファイルに含まれる定義で構成される外部サブセットがあります。外部 DTD ファイルは、パラメータエンティティを使用して、ローカルサブセットで参照されます。
持続性サブスクリプション (durable subscription)	(n.) JMS 「 パブリッシュ/サブスクライブ配信モデル (publish/subscribe delivery model) 」において、現在アクティブなサブスクライバオブジェクトがあるかどうかに関係なく存在し続けるサブスクリプション。アクティブなサブスクライバがない場合、「 JMS プロバイダ (JavaMessage Service プロバイダ) 」は、サブスクリプションによって受信されるか、またはその有効期限が切れるまでサブスクリプションの「 メッセージ (message) 」を保持します。
DWP (データベースワイヤプロトコル)	(DWP (database wire protocol)) (n.) Calendar Server 専用プロトコルの 1 つ。単一の Calendar Server システムに複数のサーバーをリンクすることにより、分散カレンダーストアを形成することができます。Calendar Server は、DWP を使って、カレンダーデータベースに格納されているリモートデータを検索します。
動的配備 (dynamic deployment)	(n.) Application Server で、J2EE アプリケーションやモジュールの配備または再配備は動的です。つまり、サーバーの再起動は必要ありません。「 動的再読み込み (dynamic reloading) 」も参照してください。
ダイナミックグループ (dynamic group)	(n.) LDAP 検索 URL によって定義されるメールグループ。通常、ユーザーはディレクトリエントリ内で LDAP 属性を設定することによってグループに参加します。
動的再読み込み (dynamic reloading)	(n.) すでに配備されたコンポーネントを修正および再読み込みする際に、完全な配備プロセスやサーバーの再起動を必要としないプロセス。デフォルトでは、サーブレット、JavaServer Pages™ (「 JSP テクノロジー (JSP technology) 」) によって作成されたページ、およびエンタープライズ Bean コンポーネントを動的に再読み込みできます。「 動的配備 (dynamic deployment) 」も参照してください。

ダイナミック
Web アプリケー
ション (**dynamic
web application**)

(n.) サーブレット、JSP™ ページ、コンテンツプロバイダ、またはユーザーのブラウザからアクセスする Java Web コンテナによって処理される必要のあるすべての要素を指します。Java Enterprise System Portal Server の場合、アプリケーションは Web サーバーにインストールされます。

E

- EAR** ファイル(エンタープライズアーカイブファイル) (EAR file (enterprise archive file)) (n.) J2EE アプリケーションを収めたアーカイブファイル。EAR ファイルの拡張子は .ear です。
- ebXML** (Electronic Business XML) (adj.)企業が「XML(拡張可能マークアップ言語)」ベースのメッセージを交換することで、業務を遂行できるように設計された一連の仕様。OASIS と U.N./CEFACT (United Nations Centre for the Facilitation of Procedures and Practices in Administration, Commerce and Transport) の後援を受けています。
- ebXML** レジストリ (ebXML registry) (Electronic Business XML registry) (n.)これは、標準および拡張可能な「メタデータ」で記述されたあらゆる種類の電子コンテンツを管理する連携された「レジストリ(registry)」およびリポジトリの一種です。
- ECC** (elliptic curve cryptography) (n.)モバイル環境やワイヤレス環境における、楕円曲線を用いた公開鍵暗号化の1つ。
- 電子商取引 (e-commerce (electronic commerce)) (n.)インターネットで行うビジネス。
- EHLO** コマンド (EHLO command) (n.)サーバーが拡張 SMTP コマンドをサポートするかどうかをサーバーに照会するための SMTP コマンド。RFC 1869 に定義されています。
- EIS**(エンタープライズ情報システム) (EIS (enterprise information system)) (n.)企業全体の情報を処理する、企業の既存システムの構成要素となるアプリケーション。これらのアプリケーションによって、企業の情報インフラストラクチャーが提供されます。エンタープライズ情報システムでは、詳細に定義されたサービスのセットがクライアントに提供されます。これらのサービスは、ローカルインタフェースやリモートインタフェースとしてクライアントに公開されます。エンタープライズ情報システムの例には、エンタープライズリ

ソースプランニングシステム、メインフレームトランザクション処理システム、レガシーデータベースシステムなどがあります。具体的には、R/3、PeopleSoft、Tuxedo、CICSなどです。

- EIS リソース (EIS resource)** (n.) エンタープライズ情報システム固有の機能をクライアントに提供するリソース。データベースシステムのレコードまたはレコードセット、エンタープライズリソースプランニングシステムのビジネスオブジェクト、トランザクション処理システムのトランザクションプログラムなどがあります。
- EJB コンテナ (EJB container)** (n.) J2EE アーキテクチャーの EJB コンポーネント規約を実装するコンテナです。この規約は、セキュリティ、並行性、ライフサイクル管理、トランザクション、配備、ネーミング、およびその他のサービスなど、「[エンタープライズ Bean \(enterprise bean\)](#)」の実行時環境を指定します。EJB コンテナは、EJB または J2EE サーバーによって提供されます。「[コンテナ \(container\)](#)」も参照してください。
- EJB コンテナプロバイダ (EJB container provider)** (n.) EJB コンテナを提供するベンダー。
- EJB コンテキスト (EJB context)** (n.) 「[エンタープライズ Bean \(enterprise bean\)](#)」がコンテナが提供するサービスを呼び出したり、クライアントが呼び出したメソッドの呼び出し側に関する情報を取得することを可能にするオブジェクト。
- EJB ホームオブジェクト (EJB home object)** (n.) 「[エンタープライズ Bean \(enterprise bean\)](#)」のライフサイクル操作 (作成、削除、検索) を提供するオブジェクト。EJB ホームオブジェクトのクラスは、コンテナの配備ツールで生成されます。EJB ホームオブジェクトは、エンタープライズ Bean の「[ホームインタフェース \(home interface\)](#)」を実装します。クライアントは、EJB ホームオブジェクトを参照して、EJB オブジェクト上のライフサイクル操作を実行します。クライアントでは、EJB ホームオブジェクトを検索するために「[JNDI 名 \(Java Naming and Directory Interface 名\)](#)」を使用します。
- EJB JAR ファイル (EJB JAR file)** (n.) 「[EJB モジュール \(EJB module\)](#)」を収めたアーカイブファイル。EJB JAR ファイルの拡張子は .jar です。
- EJB モジュール (EJB module)** (n.) 1 つまたは複数の「[エンタープライズ Bean \(enterprise bean\)](#)」および EJB 配備記述子で構成される配備可能なユニット。「[モジュール](#)」も参照してください。
- EJB オブジェクト (EJB object)** (n.) 「[エンタープライズ Bean \(enterprise bean\)](#)」の「[リモートインタフェース \(remote interface\)](#)」を実装するクラスのオブジェクト。クライアントがエンタープライズ Bean を直接参照することはなく、常に EJB オブジェクトを参照します。EJB オブジェクトのクラスは、コンテナの配備ツールで生成されます。
- EJB サーバー (EJB server)** (n.) 「[EJB コンテナ \(EJB container\)](#)」にサービスを提供するソフトウェア。たとえば、EJB コンテナは通常 EJB サーバーの一部であるトランザクションマネージャーを利用して、関連するすべてのリソースマネージャーに対する 2 段階のコミットを実行し

ます。J2EE アーキテクチャーでは、EJB コンテナは同じベンダーの EJB サーバーによって処理されることを前提としているので、この2つのエンティティー間の規約は指定されていません。1つの EJB サーバーで1つまたは複数の EJB コンテナを処理することができます。

**EJB サーバ
プロバイダ (EJB
server provider)**

(n.) EJB サーバーを提供するベンダー。

EJB™ QL

(EJB Query Language) (n.) 「**コンテナ管理による持続性 (container-managed persistence)**」を持つ「**エンティティー Bean (entity bean)**」の検索および選択メソッドのクエリーを定義します。SQL92 の部分集合である EJB QLは、エンティティー Bean の「**抽象スキーマ (abstract schema)**」に定義されている関係に対する操作を可能とする拡張機能を備えています。

**EJB テクノロジ
(Enterprise
JavaBeans™™ テク
ノロジ)**

(EJB technology (Enterprise JavaBeans technology)) (n.) エンタープライズレベルでオブジェクト指向の分散アプリケーションを開発および配備するためのコンポーネントアーキテクチャー。エンタープライズ JavaBeans アーキテクチャーで記述されたアプリケーションは、スケーラブルかつトランザクション可能であり、セキュリティ保護されています。「**エンタープライズ Bean (enterprise bean)**」も参照してください。

**ejbc ユーティリ
ティー (ejbc
utility)**

(n.) エンタープライズ Bean のコンパイラ。このユーティリティーは、すべての EJB クラスとインタフェースが EJB 仕様に合っているかどうかをチェックして、スタブとスケルトンを作成します。

要素 (element)

(n.) より大きな集合体の一部分。たとえば、配列内のデータ単位や論理要素など。「**XML (拡張可能マークアップ言語)**」ファイルでは、要素が基本的な構造単位であり、タグで区切られます。XML 要素の中には、サブ要素または「**データ (data)**」があり、「**属性 (attribute)**」を持つこともあります。

**楕円曲線暗号方
式 (elliptic curve
cryptography)**

「**ECC**」を参照してください。

**空要素タグ
(empty tag)**

(n.) コンテンツを何も囲んでいない「**XML (拡張可能マークアップ言語)**」タグ。

**暗号化
(encryption)**

(n.) 情報を判読不能にすることにより、不正な使用から情報を保護するプロセス。暗号化方式の中には、鍵と呼ばれるコードを使用して情報を暗号化するものもあります。「**復号化 (decryption)**」も参照してください。

端点 (endpoint)

(1) (n.) 負荷分散されたクラスタ内のマシンの IP アドレスまたはホスト名。

(2) (n.) Java Message Service のメッセージコンシューマ。「**メッセージ駆動型 Bean (message-driven bean)**」を参照してください。

(3) (n.) `javax.jws.WebService` 注釈で注釈が付けられた Java クラス。通常はサーブレットやステートレスセッション Bean などです。この注釈は、クラスを「**Web サービス (web service)**」端点として定義します。この端点では、Web サービスクライアントからのメッセージを受け取ります。

エンドユーザー (end user) (n.) インターネットブラウザやモバイルデバイス GUI などのグラフィカルユーザーインターフェイスを通して、分散型アプリケーションを使用するユーザーのこと。アプリケーションによってサポートされる同時使用エンドユーザー数は、アプリケーションの「**配備アーキテクチャー (deployment architecture)**」の重要な決定要因です。

ENS 「**イベント通知サービス (event notification service)**」を参照してください。

エンタープライズ Bean (enterprise bean) (n.) ビジネスタスクやビジネスエンティティを実装し、EJB コンテナによってホストされる J2EE コンポーネント。「**エンティティ Bean (entity bean)**」、「**メッセージ駆動型 Bean (message-driven bean)**」、「**セッション Bean (session bean)**」のいずれか。「**コンテナ (container)**」も参照してください。

エンタープライズ Bean プロバイダ (enterprise bean provider) (n.) エンタープライズ Bean クラス、リモートインタフェース、ホームインタフェース、および配備記述子ファイルを生成し、それらを EJB JAR ファイルにパッケージ化するアプリケーション開発者。

エンタープライズネットワーク (enterprise network) (n.) 地理的に分散している相互接続されたネットワークの集合で構成されるネットワーク。エンタープライズネットワークは、広範囲に分散している企業のニーズを満たすもので、企業のミッションクリティカルなアプリケーションで使用されません。

エンティティ (entity) (1) (n.) 「**XML (拡張可能マークアップ言語)**」ファイルで、明確に区別された個々の項目。参照することで XML ドキュメントに含めることができます。このようなエンティティ参照では、1 文字程度の小さいエンティティに名前を付けることがあります。たとえば `<` は小なり記号 (左角かっこ `<`) を参照します。エンティティ参照では、ドキュメント全体や外部エンティティを参照したり、DTD 定義のコレクションを参照したりすることもあります。

エンティティ Bean (entity bean) (n.) データベースに保持される持続データを表す EJB 1.x または 2.x 「**エンタープライズ Bean (enterprise bean)**」。エンティティ Bean では、自身の持続性を管理したり、この機能をコンテナに委譲したりすることができます。エンティティ Bean は、プライマリキーで識別されます。エンティティ Bean がホストされているコンテナがクラッシュしても、エンティティ Bean、そのプライマリキー、およびすべてのリモート参照は保護されます。また、常にトランザクションおよびマルチユーザーを認識します。「**持続性 (persistence)**」、「**メッセージ駆動型 Bean (message-driven bean)**」、「**読み取り専用 Bean (read-only bean)**」、「**セッション Bean (session bean)**」も参照してください。

エンティティ参照 (entity reference)	(n.) 「XML (拡張可能マークアップ言語)」ドキュメントの解析時に挿入される参照エンティティへの参照。< のような事前定義されたエンティティを参照したり、DTD で定義されたエンティティを参照したりすることができます。XML データでは、DTD のローカルサブセットに定義されたエンティティを参照したり、外部 XML ファイル (外部エンティティ) を参照したりすることもあります。さらに、DTD 仕様の一部のセグメントに名前を付けて、パラメータエンティティを定義することで DTD の複数ポイントで再使用できるように (含められるように) することもできます。
エントロピ (entropy)	(n.) 閉ざされたシステムでの無作為性を示す尺度。特に SSL においては、乱数の生成にエントロピを導入する (無作為性を確保する) ために複数のシードが使用されます。
エントリ (entry)	(n.) 属性のグループと一意の識別名。
エントリの分散 (entry distribution)	(n.) 拡張して多くのエントリをサポートするために、複数のサーバー全部にディレクトリのエントリを分散する手法。
エントリ ID リスト (entry ID list)	(n.) エントリ ID のリスト。ディレクトリが使用するそれぞれのインデックスは、インデックスキーのテーブルと一致するエントリ ID のリストで構成されます。エントリ ID リストは、クライアントアプリケーションの検索要求に一致する可能性のあるエントリ候補のリストを作成するために、ディレクトリが使用します。
列挙 (enumeration)	(n.) ロボットのオペレーションのフェーズ。ロボットは、ハイパーテキストリンクを抽出したりたどったりして、リソースを探索します。
エンベロープ (envelope)	(n.) 電子メールメッセージの送信側と受取人に関する情報を転送するためのコンテナ。これらの情報は、メッセージヘッダーには含まれません。エンベロープは、さまざまな電子メールプログラムで、メッセージを別の場所に移動するとき 사용됩니다。ユーザーには、メッセージのヘッダーと本体だけが表示されます。
エンベロープフィールド (envelope field)	(n.) メッセージエンベロープ内の名前付きの情報項目。RCPT TO などがあります。
等価インデックス (equality index) ERP	(n.) 特定の属性値を持つエントリを効率よく検索できるようにするインデックス。 (ERP (enterprise resource planning)) (n.) マルチモジュールのソフトウェアシステム。通常、購買、在庫、人事、顧客サービス、出荷、資金計画、ならびにその他のビジネスの重要な局面を管理するリレーショナルデータベースとアプリケーションが組み込まれています。
エラーハンドラ (error handler)	(n.) エラーを処理するプログラム。Messaging Server でのエラーハンドラは、エラーメッセージを発行し、ポストマスターが入力したエラーハンドラアクションフォームを処理します。

エラーハンドラ アクション フォーム (Error-Handler Action form)	(n.) Messaging Server が処理できない受信メッセージとともにポストマスターアカウントに送信されるフォーム。ポストマスターは、フォームに入力して、メッセージの処理方法をサーバーに指示します。
ESMTP	「 Extended Simple Mail Transfer Protocol 」を参照してください。
ESP	(n.) エンタープライズサービスプロバイダ。
ETRN コマンド (ETRN command)	(n.) クライアントからサーバーに対して、サーバー上でクライアントマシンを待機しているメッセージのメールキューの処理を開始するように要求する SMTP コマンド。RFC 1985 に定義されています。
イベント (event)	<p>(1) (n.) 日時が関連付けられている、カレンダー内のエントリ。イベントの例としては、カレンダーに新たに追加された会議やアポイントメントが挙げられます。</p> <p>(2) (n.) モジュールまたは外部の Java Naming and Directory Interface™ (JNDI) リソースから応答をトリガーする名前付きアクション。</p> <p>(3) (n.) 管理されるオブジェクトの状態、マスター、重大度の説明における変更。</p> <p>(4) (n.) Application Server で、サーバーの自己管理ルールに関連付けられているアクションをトリガーする出来事。「管理ルール (management rule)」も参照してください。</p>
イベント通知 サービス (event notification service)	(n.) サーバーレベルの分類可能なイベントのレポートを受け取り、特定のカテゴリのイベントについて関心があるサーバーとして登録されているサーバーに通知する汎用サービス。これにより Java Naming and Directory Interface (JNDI) Service は、リモート JNDI サーバーへの橋渡しとして動作できます。
エキスパンダ (expander)	(n.) メッセージをアドレスのリストに配信できるようにする、電子メール配信システムの一部。メールエキスパンダは、メーリングリストを実装するために使用されます。ユーザーが、users@example.com などの 1 つのアドレスにメッセージを送信すると、メールエキスパンダがリスト内のメールボックスへの配信を行います。メールエクスプローダとも呼ばれます。「 EXPN コマンド (EXPN command) 」も参照してください。
展開 (expansion)	(n.) メールリスト宛てのメッセージを、各メールリストのメンバーに必要な数のコピーに変換すること。MTA によるメールリストの処理に適用されます。
有効期限 ヘッダー (expires header)	(n.) リモートサーバーによって指定される、返されたドキュメントの有効期限時刻。
EXPN コマンド (EXPN command)	(n.) メールリストを展開するための SMTP コマンド。RFC 821 に定義されています。

メッセージを破棄する (**expunge message**)

(v.) INBOX から削除されたメッセージを永久に削除すること。「[メッセージを削除する \(delete message\)](#)」、「[メッセージをパージする \(purge message\)](#)」も参照してください。

Extended Simple Mail Transfer Protocol

(n.) インターネットメッセージ転送プロトコルの一種。ESMTP では、SMTP コマンドセットにオプションのコマンドを追加することで、その機能が拡張されています。ESMTP サーバーが、リモートサイトで実装されているコマンドを検出する機能などが含まれます。

拡張可能マークアップ言語 (**Extensible Markup Language**)

「[XML \(拡張可能マークアップ言語\)](#)」を参照してください。

拡張可能スタイル言語 (**extensible style language**)
拡張可能スタイル言語変換 (**extensible style language transformation**)

「[XSL \(拡張可能スタイル言語\)](#)」を参照してください。

「[XSLT \(拡張可能スタイル言語変換\)](#)」を参照してください。

外部エンティティ (**external entity**)

(n.) 外部「[XML \(拡張可能マークアップ言語\)](#)」ファイルとして存在するエンティティ。「[エンティティ参照 \(entity reference\)](#)」を使用して、XML ドキュメントに含まれます。

外部サブセット (**external subset**)

(n.) 外部 DTD ファイルへの参照によって定義される「[DTD \(ドキュメントタイプ定義\)](#)」の部分。

抽出 (**extracting**)

(n.) ドキュメント内のハイパーテキストリンクを特定するプロセス。抽出されたそれぞれのリンクは、URL プールに追加されてさらに処理されます。

エクストラネット (**extranet**)

(n.) 企業のイントラネットをインターネット領域にまで延長したネットワーク。顧客、サプライヤ、およびリモート作業者がデータにアクセスできます。

F

- 外見 (**facade**) (n.) アプリケーション固有のステートフルセッション Bean を使用して、さまざまな Enterprise JavaBeans™ コンポーネントを管理する状態。
- facility (機能) (n.) Messaging Server ログファイルエントリでの、ログエントリを生成したソフトウェアサブシステム(ネットワークやアカウントなど)の指定。
- ファクトリクラス (**factory class**) (n.) 持続マネージャーを作成するクラス。「[接続ファクトリ \(connection factory\)](#)」も参照してください。
- フェイルオーバー (**failover**) (1) (n.) サーバーがクラッシュしてもセッション、サーブレット、またはステートフルセッション Bean の状態が透過的に維持される回復プロセス。「[持続性 \(persistence\)](#)」、「[セッションフェイルオーバー \(session failover\)](#)」も参照してください。
(2) (n.) 冗長バックアップを提供するための、あるシステムから別のシステムへのコンピュータサービスの自動転送。
- ファミリーグループ管理者 (**family group administrator**) (n.) ファミリーグループ内のファミリーメンバーの追加と削除を行うための管理権限を持つユーザー。このユーザーは、グループのほかのメンバーにファミリーグループ管理アクセス権を与えることができます。
- ファンシインデックス (**fancy indexing**) (n.) シンプルインデックスよりも多くの情報を提供するインデックス化の手法。ファンシーインデックスは、ファイルサイズ、最後の変更日付、およびファイルタイプを反映するアイコンと一緒に、名前ごとのコンテンツリストを表示します。このため、ファンシーインデックスは、クライアントがロードするときに、シンプルインデックスの場合より時間がかかります。
- 致命的エラー (**fatal error**) (n.) ドキュメントの形式に問題があるか、処理できない場合、SAX パーサーで致命的エラーが発生します。「[警告 \(warning\)](#)」も参照してください。

連携アイデンティティー (federated identity)	(n.) 1人のユーザーがアクセスするすべてのサービスプロバイダのアカウント情報の融合。アカウント情報には個人データ、認証情報、購入の習慣と履歴、購買趣向などがあります。情報は、ユーザーによって管理され、ユーザーの同意の下、ユーザーが選択したプロバイダでセキュリティ保護されて共有されます。
連携 cookie (federation cookie)	(n.) 連携 cookie は、fedCookie という名前で Access Manager によって実装される cookie です。主体の連携ステータスに基づいて、「yes」または「no」のどちらかの値を保管できます。LAP 仕様によって定義されている部分ではありません。
連携の終了 (federation termination)	(n.) ユーザーのアイデンティティープロバイダアカウントと連携サービスプロバイダアカウントの間に確立された連携関係を取り消すプロセス。連携の解除とも呼ばれます。
ファイル キャッシュ (file cache)	(n.) ファイルキャッシュには、ファイルに関する情報と静的なファイルコンテンツが保存されます。ファイルキャッシュはデフォルトで有効になっています。
ファイル拡張子 (file extension)	(n.) ファイル名の最後の部分で、通常はファイルタイプを定義します。たとえば、ファイル名が index.html の場合、ファイル拡張子は html です。
ファイル転送プロトコル (file transfer protocol)	「 FTP (ファイル転送プロトコル) 」を参照してください。
ファイルタイプ (file type)	(n.) 所定のファイルの形式。たとえば、グラフィックファイルのファイルタイプは、テキストファイルのファイルタイプとは異なります。ファイルタイプは、通常そのファイル拡張子によって識別されます。「 ファイル拡張子 (file extension) 」も参照してください。
フィルタ (filter)	<p>(1) (n.) 検索要求において、検索範囲内にあるエントリが、検索応答で返されるために一致する必要があるパターン。フィルタは、ルール定義とアクセス制御定義を作成する場合にも使用されます。</p> <p>(2) (n.) 特定タイプのリソースを定義する一連の規則。これらのフィルタは、ロボットが受け入れまたは無視するリソースのタイプを定義するために、サイト定義で使用されます。</p> <p>(3) (n.) 要求や応答のヘッダーかコンテンツ、またはその両方を変換できるオブジェクト。Web コンポーネントとは異なり、フィルタは、通常は応答を作成しますが、リソースに対する要求を修正したりリソースからの応答を修正したりします。フィルタは、複数のタイプの Web リソースから構成できるように、フィルタの対象となる Web リソースに対して依存関係がないようにしてください。</p>
フィルタチェーン (filter chain)	(n.) 「 XSLT (拡張可能スタイル言語変換) 」変換を連結したもの。ある変換の出力が次の変換の入力になります。

フィルタを適用したロール (filtered role)	(n.) ロールをエントリに割り当てるための手法。各エントリが持っている属性に応じて、エントリをロールに割り当てることができます。これには、LDAP フィルタを指定します。フィルタに一致するエントリは、ロールを所有するように指示されます。
フィルタの実行 (filtering)	(n.) ドキュメントが、インデックスに加えるべきサイトの一部であるかどうかを判断するプロセス。
検索メソッド (finder method)	(n.) グローバルに利用可能なディレクトリから、クライアントが「エンティティー Bean (entity bean)」または Bean のコレクションを調べることができるようにする、「ホームインタフェース (home interface)」に定義されたメソッド。
ファイアウォール	(n.) 組織内部のネットワーク接続されたコンピュータを、通常はハードウェアとソフトウェアの両方により、外部からのアクセスから保護するネットワーク構成。一般に、ファイアウォールは物理的な建物または組織のサイト内にある、ネットワークの電子メールやデータファイルなどの情報を保護するために使用されます。
フレキシブルログ形式 (flexible log format)	(n.) アクセスログに情報を入力する場合にサーバーが使用する記録形式。
フォルダ (folder)	(n.) メッセージの名前付きのコレクション。フォルダにはほかのフォルダを含めることができます。メールボックスとも呼ばれます。「個人用フォルダ (personal folder)」、「公開フォルダ (public folder)」、「共有フォルダ (shared folder)」、および「INBOX」も参照してください。
フォームアクションハンドラ (form action handler)	(n.) サーブレットまたはアプリケーションロジック内で特別に定義されているメソッド。フォーム上の名前付きのボタンに基づいてアクションを実行します。
フォームベース認証 (form-based authentication)	(n.) Web コンテナがアプリケーション固有のログイン用フォームを提供する認証メカニズム。この認証フォームでは Base 64 エンコーディングを使用するため、SSL を使用していない接続がある場合は、ユーザー名とパスワードが公開されてしまう可能性があります。
FORTEZZA	(n.) 重要ではあっても機密扱いされない情報を管理するために、米国政府機関によって使用される暗号化システム。
転送 (forwarding)	「メッセージ転送 (message forwarding)」を参照してください。
Foundation Profile	(n.) CDC を一緒にした API セット。次世代アプリケーション、家電製品、および組み込み型デバイスをターゲットとした J2ME™ アプリケーション実行環境を提供します。

断片化 (fragmentation)	(n.) 大きなメッセージを複数の小さなメッセージに分割する MIME の機能。「 デフラグ (defragmentation) 」も参照してください。
フレッシュスタート (fresh start)	(n.) 開始点からロボットを起動すること。フレッシュスタートによってロボットの状態情報を削除し、ロボットが初期状態から次の実行を開始するようにします。再起動とは反対の概念です。
FSMO ロール (FSMO role)	(Flexible Single-Master Operation role) (n.) マルチマスター配備で更新の競合が発生しないように Active Directory で使用されるメカニズム。マルチマスター配備であっても、一部のオブジェクトはシングルマスターモードで更新されます。これは、Windows NT ドメインにおけるプライマリドメインコントローラ (PDC) の従来の概念とよく似ています。Active Directory 配備には 5 つの FSMO ロールがありますが、PDC エミュレータロールだけが Identity Synchronization for Windows に影響します。パスワードの更新がすぐにレプリケートされるのは PDC エミュレータロールを持つ Active Directory ドメインコントローラだけであるため、Identity Synchronization for Windows ではこのドメインコントローラを使用して同期します。そうでないと、Directory Server との同期に数分の遅延が発生する場合があります。
FTP (ファイル転送プロトコル)	(FTP (file transfer protocol)) (n.) ネットワーク上で 1 つのコンピュータから別のコンピュータへのファイル転送を実現するインターネットプロトコルの 1 つ。
完全修飾ドメイン名 (fully qualified domain name)	(n.) ホスト名とドメイン名の両方を含んだシステムのフルネーム。次に例を示します。example.sun.com の場合は、example がサーバーのホスト名、sun.com がドメイン名です。

G

- ゲートウェイ (gateway)** (n.) 1つのネイティブ形式から別の形式に変換するシステム。たとえば、X.400 と RFC 822 間の送受信を行う電子メールゲートウェイなどがあります。複数の電子メールシステム (特に、2つの異なるネットワーク上の類似性のないメールシステム) を接続し、その間でメッセージを転送するマシンです。マッピングと変換は複雑になることがあります。一般的に、あるシステムからメッセージを完全に受け取ってから適切な変換を行なって次のシステムに送信するようなストアアンドフォワードのしくみが必要です。
- 汎用アクセス (general access)** (n.) ユーザーアクセスのレベル。このレベルのアクセスが付与されている場合は、すべての認証済みユーザーがディレクトリ情報にアクセスできることを示します。
- 汎用 ACL (general ACL)** (n.) ユーザーまたはグループと1つまたは複数のアクセス権を関連させた、Java Enterprise System Directory Server の名前付きアクセス制御リスト。このリストは、自由に定義およびアクセスして、任意のアクセス権のセットを記録できます。
- 汎用エンティティ (general entity)** (n.) 「XML (拡張可能マークアップ言語)」ドキュメントのコンテンツの一部として参照されるエンティティ。「DTD (ドキュメントタイプ定義)」内で参照されるパラメータエンティティとは異なります。汎用エンティティは、解析対象エンティティの場合も解析対象外エンティティの場合もあります。
- 生成 (generation)** (n.) ロボットのオペレーションのフェーズ。ロボットは、列挙フェーズで検出した各リソースのリソース記述を生成します。
- 汎用サーブレット (generic servlet)** (n.) javax.servlet.GenericServlet を拡張するサーブレット。汎用サーブレットはプロトコルに依存しません。よって、HTTP やその他の転送プロトコルのサポートを初めから備えていません。「HTTP サーブレット (HTTP servlet)」も参照してください。
- GIF (グラフィックス交換形式)** (GIF (graphics interchange format)) (n.) CompuServe によって開発されたクロスプラットフォームのイメージ形式。GIF ファイルは通常、TIFF などのほかのグラフィックファイルタイプよりもサイズがかなり小さくなっています。GIF は、広く利用される

一般的な交換形式です。GIF イメージは、UNIX、Microsoft Windows、および Apple Macintosh の各システムでそのまま表示できます。

- グローバルデータベース接続 (global database connection)** (n.) 複数のコンポーネントに対して利用可能なデータベース接続。接続にはリソースマネージャーが必要です。
- グローバルトランザクション (global transaction)** (n.) トランザクションマネージャーによって管理および調整され、1つのデータベースおよびプロセスに制限されないトランザクション。トランザクションマネージャーは通常、XA プロトコルを使ってデータベースのバックエンドと対話します。「ローカルトランザクション (local transaction)」も参照してください。
- GMT (グリニッジ標準時)** (GMT (Greenwich Mean Time)) (n.) 英国のグリニッジ子午線の平均太陽時であり、世界中のほかのタイムゾーンの基準となる時間標準。GMT は、夏時間による影響を受けません。
- 細分レベル (granularity level)** (n.) アプリケーションを細分化するアプローチ。細分度が高いとは、アプリケーションが細かく定義された多数の Enterprise JavaBeans™ コンポーネントに分割されていることを示します。細分度が低いとは、アプリケーションの分割数が少なく、大きなプログラムが生成されていることを示します。
- グリーティングフォーム (greeting form)** (n.) ユーザーのアカウントが作成されたときにユーザーに送信されるメッセージ。このフォームは、新しいアカウントを確認し、その内容を検証するために使用されます。
- グループ (group)** (1) (n.) 役職や顧客構成などの一般的な性質によって分類された、ユーザーの認証済みセット。グループは、ロールのセットとも関連付けられ、グループのメンバーである各ユーザーは、そのグループに割り当てられたすべてのロールを継承します。デフォルトユーザーグループと標準ユーザーグループの2つがあります。通常、グループのメンバーシップはローカルシステム管理者が管理します。「ユーザー」、[「ロール \(role\)」](#)も参照してください。
(2) (n.) [「識別名 \(distinguished name\)」](#)によって管理される LDAP メールエントリの集合。通常は、メールリストとして使用されますが、グループのメンバーに特定の管理権限を付与するために使用される場合もあります。[「ダイナミックグループ \(dynamic group\)」](#)、[「スタティックグループ \(static group\)」](#)も参照してください。
- グループフォルダ (group folders)** (n.) 共有フォルダとグループフォルダを格納しているフォルダ。「[公開フォルダ \(public folder\)」](#)、「[共有フォルダ \(shared folder\)」](#)も参照してください。
- グループ ID (group ID)** (n.) カウンタやログなどの Calendar Server ファイルのグループ ID。グループ ID は、ics.conf ファイル内の local.servergid パラメータに格納されます。GID とも呼ばれます。

グループスケ
ジューリングエ
ンジン (**group
scheduling
engine**)

(n.) グループスケジューリングを処理する Calendar Server プロセス。このエンジンにより、ユーザーは同一サーバーまたは別のサーバーのほかのカレンダーユーザーとの間で、イベントのスケジューリングを行うことができます。これに対し、ほかのユーザーは、イベントの修正や取り消し、またはイベントへの応答を行うことができます。

GUI

(n.) グラフィカルユーザーインターフェース。

H

HA	「 高可用性 (high availability) 」を参照してください。
HA データサービス (HA data service)	「 データサービス (data service) 」を参照してください。
HADB	「 高可用性データベース (HADB) 」を参照してください。
HADB ノード (HADB node)	(n.) セッションデータの保存と更新で使用される、一連の HADB プロセス、共有メモリーの専用領域、および1つ以上の二次ストレージデバイス。各アクティブ (データストレージ) ノードにはミラーノードが1つ必要なので、ノードはペアとして存在します。また、2つ以上のスペアノードを組み込んで、可用性を最大限に高めることができます。アクティブノードで障害が発生してタイムアウト期限内にリカバリできない場合は、スペアノードがミラーノードからデータをコピーしてアクティブになります。「 高可用性データベース (HADB) 」も参照してください。
ハンドル (handle)	(n.) 「 エンタープライズ Bean (enterprise bean) 」を識別するオブジェクト。クライアントはハンドルを直列化した後で直列化を解除し、Bean への参照を取得できます。
ハード再起動 (hard restart)	(n.) プロセスまたはサービスの終了と、その後の再起動。「 ソフト再起動 (soft restart) 」も参照してください。
hashdir	(n.) 特定ユーザーのメッセージストアが含まれるディレクトリを調べるためのコマンド行ユーティリティ。
HDML (ハンドヘルドデバイスマークアップ言語)	(HDML (Handheld Device Markup Language)) (n.) Openwave ブラウザを使用するモバイルデバイスをプログラムするための Openwave の独自の言語。
ヘッダー (header)	(n.) 電子メールメッセージでメッセージ本体の前にある部分。ヘッダー内では、フィールド名のあとにコロンと値が続きます。ヘッダーには、電子メールプログラムとユーザーにとって、メッセージが意味をなすようにするために有用な情報が含

まれています。たとえば、ヘッダーには、配信情報、内容の概要、トレース、および MIME 情報が記述されます。また、メッセージの受取人、送信側、送信日時、件名も記述されます。ヘッダーは、電子メールプログラムが読み取れるように RFC 822 に従って記述されている必要があります。

ヘッダーフィールド (header field)

(n.) メッセージヘッダーにある「From:」や「To:」などの情報項目。ヘッダー行とも呼ばれます。

ハートビート (heartbeat)

(n.) Application Server で、クラスタ内の使用可能なすべてのサーバーに送信される定期メッセージ。特定の間隔が経過し、何回か試行してもハートビートがない場合は、「フェイルオーバー (failover)」がトリガーされる可能性があります。

ヒューリスティックな判断 (heuristic decision)

(n.) 特定のトランザクションが使用するトランザクションモード。トランザクションは、コミットまたはロールバックする必要があります。

高可用性 (high availability)

(n.) サービスの中断を検出できるようにし、システム障害やプロセス失敗時のリカバリメカニズムを提供します。また高可用性により、一次システムに障害が発生した場合には、バックアップシステムがサービスを引き継ぐことができるようにします。HA とも呼ばれます。

高可用性データベース (HADB)

(high availability database (HADB)) (n.) スケーラビリティと可用性に優れたセッション状態持続インフラストラクチャー。Application Server は、HADB を使用して、HTTP セッション状態とステートフルセッション Bean 状態を保存します。「HADB ノード (HADB node)」、「アクティブノード (active node)」も参照してください。

ホームハンドル (home handle)

(n.) ホームインタフェースへの参照を取得するために使用できるオブジェクト。ホームハンドルは、直列化されて安定化記憶領域に書き込まれ、直列化復元されて参照が取得されます。

ホームインタフェース (home interface)

(n.) クライアントによる EJB 1.x または 2.x 「エンタープライズ Bean (enterprise bean)」の作成や削除を可能にするメソッドを定義するインタフェース。セッション Bean のホームインタフェースでは、create と remove の各メソッドを定義しますが、エンティティ Bean のホームインタフェースでは、create、finder、remove の各メソッドを定義します。「リモートインタフェース (remote interface)」も参照してください。

ホームページ (home page)

(n.) サーバーに置かれ、サーバーのコンテンツのカタログまたはエントリポイントとして動作するドキュメント。このドキュメントの位置は、サーバーの設定ファイル内で定義されます。

ホップ (hop)

(n.) 2 台のコンピュータ間での送信。

<p>水平方向のスケラビリティ (horizontal scalability)</p>	<p>(n.) 単一のサーバー上で実行できる Calendar Server 機能。可能な構成オプションが多岐にわたる複数のサーバー上に分散されているプロセスを1つのグループとして実行することもできます。</p>
<p>ホスト (host)</p>	<p>(n.) 1つ以上のサーバーが置かれているマシン。</p>
<p>ホストドメイン (hosted domain)</p>	<p>(n.) ISP にアウトソースされた電子メールドメイン。ISP は、企業の電子メールサービスを運営および管理し、その企業の電子メールドメインのホスティングを提供します。ホストドメインは、ほかのホストドメインと同一の Java Enterprise System Messaging Server ホストを共有します。初期の LDAP ベースの電子メールシステムでは、1つのドメインが1つまたは複数の電子メールサーバーホストによってサポートされていました。Messaging Server では、複数のドメインを単一のサーバーでホストできます。各ホストドメインには、そのドメインのユーザーとグループのコンテナを指す LDAP エントリがあります。仮想ホストドメインまたは「仮想ドメイン (virtual domain)」とも呼ばれます。</p>
<p>ホスト-IP 認証 (host-IP authentication)</p>	<p>(n.) 特定のコンピュータを使うクライアントだけにアクセスを限定することによって、Java Enterprise System Administration Server、または Web サイト上のファイルやディレクトリへのアクセスを制限するセキュリティーメカニズム。</p>
<p>ホスト名 (host name)</p>	<p>(n.) ドメイン内の特定マシンの名前。ホスト名は、「簡易形式」ホスト名 (mail など) または完全修飾ホスト名として表記される IP ホスト名です。完全修飾ホスト名は、ホスト名と「ドメイン名 (domain name)」で構成されます。たとえば、mail.example.com は、comexample.com ドメイン内の mail という名前のホストを表します。ホスト名は、ドメイン内で一意にする必要があります。組織は、たとえば、mail.corp.example.com と mail.field.example.com のようにマシンが異なるサブドメイン内にあれば、mail という名前のマシンを複数構成できます。ホスト名は、常に、特定の IP アドレスにマップされます。「完全修飾ドメイン名 (fully qualified domain name)」、「IP アドレス (IP address)」も参照してください。</p>
<p>ホスト名の非表示 (host name hiding)</p>	<p>(n.) 特定の内部ホスト名を含まないドメインベースの電子メールアドレスを使用すること。</p>
<p>HTML (ハイパーテキストマークアップ言語)</p>	<p>(HTML (hypertext markup language)) (n.) インターネット上のハイパーテキストドキュメントに使用されるマークアップ言語。HTML では、URL を使用したイメージ、サウンド、ビデオストリーム、フォームフィールドなどのオブジェクトへの参照の埋め込みと、テキストの基本的な書式設定が可能です。テキストの各ブロックは、テキストの種類を指定したコードで囲まれます。</p>
<p>HTML ページ (HTML page)</p>	<p>(n.) HTML でコード化され、Web ブラウザで表示することを目的としたページ。</p>

HTTP	(hypertext transfer protocol (HTTP)) (n.) リモートホストからハイパーテキストオブジェクトをフェッチする Transmission Control Protocol/Internet Protocol ベースのインターネットプロトコル。HTTP メッセージには、クライアントからサーバーへの要求と、サーバーからクライアントへの応答があります。
HTTPD (ハイパーテキスト転送プロトコルデーモン)	(HTTPD (hypertext transfer protocol daemon)) (n.) HTTP デーモンまたはサービスの略。HTTP プロトコルを使用して情報を提供するプログラムです。
HTTP-NG	(hypertext transfer protocol-next generation) (n.) 次世代 HTTP。
HTTPS	「 Secure Socket Layer 」プロトコルを使用した HTTP 実装のセキュリティー保護バージョン。
HTTP サブレット (HTTP servlet)	(n.) <code>javax.servlet.HttpServlet</code> を拡張するサブレット。HTTP サブレットには、HTTP プロトコルのサポートが組み込まれています。「 汎用サブレット (generic servlet) 」も参照してください。
ハブ (hub)	(n.) システムの単一接続ポイントとして機能するホスト。たとえば、2つのネットワークがファイアウォールで分離されている場合は、しばしばファイアウォールコンピュータがメールハブとして機能します。
hypertext transfer protocol secure	「 HTTPS 」を参照してください。

**iCalendar
Message-Based
Interoperability
Protocol**

(n.) このプロトコルは、「[iCalendar Transport-Independent Interoperability Protocol](#)」からインターネット電子メールベースのトランスポートへのバインディングを指定します。このプロトコルは、iMIPとも呼ばれ、RFC 2447で定義されています。

**iCalendar
Transport-Independent
Interoperability
Protocol**

(n.) iCalendar オブジェクト仕様を基にしたインターネットプロトコル。さまざまなカレンダーシステム間でスケジュール機能を相互に利用できるようにします。このプロトコルは、iTIPとも呼ばれ、RFC 2446で定義されています。

IDE (統合開発環境)

(IDE (integrated development environment)) (n.) 単一のグラフィカルユーザーインターフェイスを使用して、コードを作成、アセンブル、配備、およびデバッグするためのソフトウェア。

IDENT

「[Identification Protocol](#)」を参照してください。

**Identification
Protocol**

(n.) 特定のTCP接続のリモート端末を制御するリモートプロセスを識別できるようにするプロトコル。このプロトコルはIDENTとも呼ばれます。RFC 1413に定義されています。

**アイデンティティー
(identity)**

(n.) 特定のエンドユーザーを明確に区別するための一連の情報。ユーザー識別子とパスワード、電子メールアドレス、個人の志向 (音楽の種類、オプトインマーケティングやオプトアウトマーケティングの決定など)、特定業務に固有の情報 (銀行口座番号、出荷先住所など) を定義することで、エンドユーザーは同様にサービスを使用する他のユーザーから自分自身を区別します。

**アイデンティティー連携
(identity
federation)**

(n.) 明確なサービスプロバイダアカウントとアイデンティティープロバイダアカウントを統合する場合に発生するプロセス。ユーザーは、個別のアカウント情報を各プロバイダに保持すると同時に、リンクを確立して、プロバイダのアカウント間で認証情報を交換できます。アカウント連携とも呼ばれます。

アイデンティティプロバイダ (identity provider)	(n.) 認証サービスの提供に特化したサービスプロバイダ。アイデンティティプロバイダは、認証の管理サービスとして、識別情報を保守管理します。アイデンティティプロバイダによる認証は、アイデンティティプロバイダが連携するすべてのサービスプロバイダで有効です。
アイデンティティサービス (identity service)	(n.) アイデンティティサービスはリソースを対象とする Web サービスで、主体(アイデンティティ)に関連するデータ属性に対して、情報の取得や更新を行ったり、何らかのアクションを実行したりします。アイデンティティサービスには、企業の電話帳やカレンダーサービスなどがあります。
IDL (インタフェース定義言語)	(IDL (interface definition language)) (n.) 「CORBA (共通オブジェクトリクエストブローカーアーキテクチャー)」オブジェクトとのインタフェースを定義するために使用する言語。このインタフェースは、オペレーティングシステムやプログラミング言語に依存しません。遠隔手続き呼び出し(「RPC (遠隔手続き呼び出し)」)の機能インタフェースを記述して、コンパイラがプロキシを生成し、マシン間でパラメータを序列化するコードをスタブできるようにします。
アイドル状態 (idle state)	(n.) ロボットが動作していても、URL プール内のすべての URL の処理が済んでいる状態。この状態にある間、ロボットは状態要求に応答できます。
iHTML (i-mode ハイパーテキストマークアップ言語)	(iHTML (i-mode hypertext markup language)) (n.) NTT DoCoMo の日本語 i-mode サービスで使用する言語。
IIOP	(Internet Inter-ORB Protocol) (n.) IIOP 経由のリモートメソッド呼び出し(「RMI (リモートメソッド呼び出し)」)および共通オブジェクトリクエストブローカーアーキテクチャー(「CORBA (共通オブジェクトリクエストブローカーアーキテクチャー)」)の両方で使用されるトランスポートレベルのプロトコル。CORBA オブジェクトリクエストブローカ間の通信に使用されます。
IIOP クラスタ (IIOP cluster)	(n.) RMI/IIOP 要求の高可用性を実現するように構成された IIOP クラスタ。
IIOP エンドポイント (IIOP endpoint)	(n.) RMI/IIOP 要求の高可用性を実現する IIOP クラスタに対応するように構成された IIOP リスナー。
IIOP リスナー (IIOP listener)	(n.) 指定されたポートで待機して、CORBA ベースのクライアントアプリケーションから送信される接続を受け入れる待機ソケット。
imagemap	(1) (n.) イメージ領域をアクティブにするプロセス。ユーザーは、イメージの異なる領域をマウスでクリックして情報を参照および取得できます。 (2) (n.) ほかの「HTTPD (ハイパーテキスト転送プロトコルデーモン)」実装のイメージマップ機能を処理するために使用する「CGI」プログラム。

IMAP4 (インター ネットメッセー ジアクセスプロ トコルバー ジョン4)	(IMAP4 (Internet Message Access Protocol Version 4)) (n.) ユーザーがメインのメッセージングシステムから切断された場合でもメールを処理できるようにする標準プロトコル。IMAP仕様により、切断されたユーザーの管理制御が可能になるとともに、メッセージングシステムに再接続したときにユーザーのメッセージストアの同期化が可能になります。
iMIP	「iCalendar Transport-Independent Interoperability Protocol」を参照してください。
直接従属 (immediate subordinate)	(n.) 「DIT」では、自身の「識別名 (distinguished name)」が、親エントリの識別名に自身の「相対識別名 (relative distinguished name)」を添付して形成されている場合、そのエントリは親エントリの直接従属です。
直接上位 (immediate superior)	(n.) 「DIT」では、自身の「識別名 (distinguished name)」に、ほかのエントリの「相対識別名 (relative distinguished name)」が続く形で子エントリの識別名が形成されている場合、そのエントリはその子エントリの直接上位です。
偽装 (impersonation)	(n.) あるオブジェクトが、制限なしで、別のオブジェクトのアイデンティティーと特権を引き受ける動作。委譲が行なわれたことを、偽装者の呼び出しを受ける側が認識できるような徴候はありません。偽装は、簡単な「委譲 (delegation)」の一例です。
インポートエー ジェント (import agent)	(n.) 「インポート (importing)」時に使用されるプロセス。
インポート (importing)	(n.) 新規または更新されたリソース記述を、別のデータベースから検索エンジンに取り込むプロセス。
imsadmin コマン ド (imsadmin commands)	(n.) ドメイン管理者、ユーザー、およびグループを管理する一連のコマンド行ユーティリティー。
imsimta コマン ド (imsimta commands)	(n.) 「MTA (メッセージ転送エージェント)」の各種の保守、テスト、管理のタスクを行うためのコマンド行ユーティリティーのセット。
非アクティブ ブート環境 (inactive boot environment)	(n.) 現在ブートしていないか、次のリポート時にアクティブになるように割り当てられていない環境。「アクティブブート環境」も参照してください。
INBOX	(n.) ユーザーのデフォルトメールボックス用に予約されている名前。メールの配信で使用されます。INBOXは、大文字と小文字を区別しない唯一のフォルダなので、INBOX、Inbox、およびinboxはすべてユーザーのデフォルトメールボックスとして有効な名前です。
インデックス (index)	(n.) リソースまたはドキュメントの集中管理された検索可能データベース。カタログとも呼ばれます。

インデックス化 (indexing)	(n.) 集中管理された検索可能なリソースのデータベースを提供するプロセス。カタログ化とも呼ばれます。
インデックス キー (index key)	(n.) ディレクトリが使用するそれぞれのインデックスは、インデックスキーのテーブルと一致するエントリ ID のリストで構成されます。
間接 CoS (indirect CoS)	(n.) ターゲットエントリの属性値の 1 つによりテンプレートエントリを識別します。
初期化パラメータ (initialization parameter)	(n.) 「サーブレット (servlet)」に関連するコンテキストを初期化する「パラメータ (parameter)」。
inittab ファイル (inittab file)	(n.) UNIX のみ。何らかの理由で停止した場合に再起動する必要があるプログラムの一覧を記述したファイル。このファイルにより、プログラムの連続動作が保証されます。このファイルは、格納されている位置に基づいて、 <code>/etc/inittab</code> とも呼ばれます。UNIX システムでも、このファイルを使用できない場合があります。
インストール ディレクトリ (installation directory)	(n.) サーバーのバイナリ (実行可能) ファイルがインストールされるディレクトリ。Messaging Server の場合、インストールディレクトリは「サーバールート (server root)」のサブディレクトリ <code>server-root/bin/msg/</code> です。「インスタンスディレクトリ (instance directory)」も参照してください。
インストールパス (installation path)	(n.) Directory Server Enterprise Edition ソフトウェアがインストールされたフルパス。インストールパスは、ソフトウェアを初めてインストールするときに選択できません。
インスタンス ディレクトリ (instance directory)	(n.) サーバーの特定インスタンスを定義するファイルを格納するディレクトリ。Messaging Server の場合、インストールディレクトリは「サーバールート (server root)」のサブディレクトリ <code>server-root/msg-instance/</code> です。 <i>instance</i> は、インストール時に指定されるサーバー名です。Application Server の場合、インスタンスディレクトリは「ドメインディレクトリ (domain directory)」のサブディレクトリです。「インストールディレクトリ (installation directory)」、「サーバーインスタンス (server instance)」も参照してください。
インスタンスパス (instance path)	(n.) Directory Server または Directory Proxy Server サーバーインスタンスのデータがあるフルパス。インスタンスパスは、サーバーインスタンスを作成するときに選択します。
Instant Messaging クラ イアント (Instant Messaging Client)	(n.) ユーザーによるインスタントメッセージとアラートの送受信を可能にするクライアント。
Instant Messaging マル チプレクサ (Instant Messaging multiplexor)	(n.) クライアント接続のマネージャー。大量の同時クライアント接続で必要になるバックエンドの Instant Messaging サーバーへの続数を少なく抑えることにより、

Instant Messaging Server のスケーラビリティを改善します。Instant Messaging クライアントは、Instant Messaging サーバー自体ではなく、マルチプレクサに接続します。ファイアウォールのパブリック側にインストールされる場合、マルチプレクサは、Instant Messaging Server をファイアウォールの背後に隠すことにより、侵入者からユーザーデータベースを保護します。

Instant Messaging Server

(1) (n.) サーバー、マルチプレクサ、Java Enterprise System Instant Messaging Server などのすべてのコンポーネントを含む、Java Enterprise System Messaging Server 製品そのものを指します。

(2) (n.) Instant Messaging Server のマルチプレクサを経由して、Instant Messaging からの送信コマンドを処理する製品内のバックエンドサーバープロセス。Instant Messaging Server は、Instant Messaging ユーザーの認証を行う場合には、LDAP サーバーとも通信します。「[Instant Messaging マルチプレクサ \(Instant Messaging multiplexor\)](#)」も参照してください。

インテリジェントエージェント (intelligent agent)

(n.) ユーザーのために、HTTP、NNTP、SMTP、FTP などのさまざまな要求を実行するサーバー内部のオブジェクト。インテリジェントエージェントは、ある意味でサーバーのクライアントとして動作し、サーバーが実行する要求を発行します。

国際化インデックス (international index)

(n.) 検索インデックスのタイプ。属性に言語タグがある「DIT」内の情報検索速度を高めます。

インターネットメッセージアクセスプロトコルバージョン 4 (Internet Message Access Protocol Version 4)

「[IMAP4 \(インターネットメッセージアクセスプロトコルバージョン 4\)](#)」を参照してください。

インターネットプロトコルネットワーク (Internet Protocol intranet)

「[IP](#)」を参照してください。

(n.) 企業や組織内における複数 Transmission Control Protocol/Internet Protocol ネットワークで構成されるネットワーク。イントラネットでは、World Wide Web で使われているものと同種のサーバーおよびクライアントソフトウェアを、企業 LAN 上で提供される社内アプリケーションとして使用できます。インターネットと通信するイントラネット上の機密情報は、通常はファイアウォールで保護されます。「[ファイアウォール](#)」、「[エクストラネット \(extranet\)](#)」も参照してください。

無効なユーザー (invalid user)

(n.) メッセージ処理中に発生するエラー状態。このエラー状況が発生すると、メッセージストアは、MTA に通信データを送信し、メッセージのコピーを削除します。MTA はメッセージを差出人に戻し、メッセージのコピーを削除します。

IP	(IP (Internet Protocol)) (n.) 世界中のネットワークを接続する Transmission Control Protocol/Internet Protocol 群内のプロトコル。米国国防総省で開発され、インターネットで使用されています。このプロトコル群の主な特長は IP プロトコルです。
IP アドレス (IP address)	(n.) 192.168.255.255 のようにドットで区切られた一連の数値。イントラネットまたはインターネット上でのマシンの実際の位置を示します。Transmission Control Protocol/Internet Protocol を使用するホストには、32 ビットアドレスが割り当てられます。
ISDN	(n.) サービス総合デジタル網。
ISINDEX	(n.) クライアントでの検索を有効にする HTML タグ。ドキュメントは、ネットワークナビゲータの機能を使用して検索文字列を受け入れ、それをサーバーに送信して、フォームを使用せずに検索可能インデックスにアクセスできます。ISINDEX HTML タグを使用するには、クエリーハンドラを作成する必要があります。
ISMAP	(n.) 名前付きイメージが「 imagemap 」であることをサーバーに通知するために HTML ドキュメントで使用される、IMG SRC タグの拡張機能。
ISO 3166	(n.) ISO (国際標準化機構) で管理される国コードの国際規格。
ISO 8601	(n.) 日付と時間の数値表現を規定する ISO (国際標準化機構) 規格。Calendar Server は、ISO8601 規格に規定されている表記を使用して日付、時間、継続時間文字列を表現します。
遮断レベル (isolation level)	「 トランザクション遮断レベル (transaction isolation level) 」を参照してください。
発行された証明書 (issued certificate)	(n.) 「 認証局 (CA) 」によって発行される証明書。「 自己生成証明書 (self-generated certificate) 」も参照してください。
ISV	(n.) 独立系ソフトウェアベンダー。
iTIP	「 iCalendar Transport-Independent Interoperability Protocol 」を参照してください。

J

- J2EE™ アプリケーション (J2EE application)** (n.) 「[J2EE プラットフォーム \(J2EE platform\)](#)」機能の配備可能なユニット。単体の J2EE モジュールの場合も、J2EE アプリケーション配備記述子とともに EAR ファイルにパッケージ化された一連のモジュールの場合もあります。J2EE アプリケーションは、通常、複数のコンピューティング階層に分散させて設計されます。
- J2EE コンポーネント (J2EE component)** (n.) コンテナでサポートされる機能が自己完結したソフトウェアユニット。配備時に設定できます。「[Web アプリケーション \(web application\)](#)」、「[セッション Bean \(session bean\)](#)」、「[エンティティ Bean \(entity bean\)](#)」、「[メッセージ駆動型 Bean \(message-driven bean\)](#)」、「[アプリケーションクライアント \(application client\)](#)」、「[コネクタ \(connector\)](#)」など。これらの J2EE コンポーネントは、Java™ プログラミング言語で記述され、Java 言語によるほかのプログラムと同様の方法でコンパイルされます。「[構成要素 \(コンポーネント\)](#)」も参照してください。
- J2EE モジュール (J2EE module)** (n.) 同じコンテナタイプの 1 つ以上の J2EE コンポーネントと、そのタイプの配備記述子 1 つで構成されるソフトウェアユニット。モジュールは、スタンドアロンユニットとして配備することも、J2EE アプリケーションに組み込むこともできます。「[ライフサイクルモジュール \(life-cycle module\)](#)」、「[モジュール](#)」も参照してください。
- J2EE プラットフォーム (J2EE platform)** (J2EE™ platform (Java 2 Platform, Enterprise Edition)) (n.) 多層 Web ベースエンタープライズアプリケーションを開発および配備するための環境。J2EE プラットフォームは、一連のサービス、「[API](#)」、およびこれらのアプリケーションを開発する機能を提供するプロトコルから構成されています。
- J2EE 製品 (J2EE Product)** (n.) J2EE プラットフォーム仕様に適合した実装。
- J2EE 製品プロバイダ (J2EE product provider)** (n.) J2EE 製品を提供するベンダー。

J2EE サーバー (J2EE server)	(n.) J2EE 製品の実行時部分。J2EE サーバーは、EJB と Web コンテナのどちらかまたは両方を提供します。「 コンテナ (container) 」も参照してください。
J2ME™ プラット フォーム (J2ME platform)	(J2ME platform (Java 2 Platform, Micro Edition)) (n.) 幅広いコンシューマ製品 (ポケットベル、携帯電話、スクリーンフォン、デジタルセットトップボックス、カーナビゲーションシステムなど) 向けに高度に最適化された Java 実行時環境。
J2SE™ プラット フォーム (J2SE platform)	(Java 2 Platform, Standard Edition) (n.) 主要な Java テクノロジプラットフォーム。
JAF	(JavaBeans™ Activation Framework) (n.) MIME データタイプのサポートを Java プラットフォームに統合します。「 MIME データタイプ (MIME Data Type) 」も参照してください。
JAR ファイルの 規約 (JAR file contract)	(n.) エンタープライズ Bean パッケージに含める必要のある情報について規定した、Java アーカイブファイルの規約。
JATO	(n.) Java プログラミング言語と「 XML (拡張可能マークアップ言語) 」で記述されたコード間で、コード変換を行うライブラリ。Sub Java System Web Application Framework および Application Framework とも呼ばれます。JATO は、エンタープライズ Web アプリケーション開発用に装備されています。JATO には、表示フィールド、アプリケーションイベント、コンポーネント階層、ページ中心開発方式などの概念が組み合わされています。
J2EE プラット フォーム (Java 2 Platform, Enterprise Edition) (J2EE platform (Java 2 Platform, Enterprise Edition))	「 J2EE プラットフォーム (J2EE platform) 」を参照してください。
J2ME プラット フォーム (Java Platform, Micro Edition) (J2ME platform (Java 2 Platform, Micro Edition)) J2SE プラット フォーム (Java 2 Platform, Standard Edition) (J2SE Platform (Java 2 Platform, Standard Edition))	「 J2ME プラットフォーム (J2ME platform) 」を参照してください。 「 J2SE プラットフォーム (J2SE platform) 」を参照してください。
JavaBean™ 名前 空間 (JavaBean namespace)	(n.) パッケージで定義される一連の要素名に一意のラベルを指定できるようにする標準。パッケージを使用するドキュメントは、要素名どうしの競合を起さずに、ほ

かのドキュメントに組み込むことができます。パッケージ内で定義されている要素は一意に識別されるので、たとえばパーサーは、別のパッケージではなく、使用するパッケージに従って要素に割り込むタイミングを決定できます。

JavaBeans Activation Framework

「[JAF](#)」を参照してください。

JavaBeans コンポーネント (JavaBeans component)

(n.) ツールによる操作や、アプリケーションの作成が可能な Java クラス。JavaBeans コンポーネントは、特定のプロパティおよびイベントのインタフェース規約に従う必要があります。

JavaBeans コンポーネントアーキテクチャー (JavaBeans component architecture)

(n.) 移植可能でプラットフォームに依存しない、再利用できるコンポーネントモデル。

Java Enterprise System

(n.) 個々の Sun ソフトウェア製品を、分散エンタープライズアプリケーションをサポートする 1 つのソフトウェアシステムにまとめた統合システム。

Java ES

「[Java Enterprise System](#)」を参照してください。

Java™ ES 共有コンポーネント (Java ES shared component)

「[共有コンポーネント \(shared component\)](#)」を参照してください。

JavaMail™ (API, 拡張) (JavaMail (API, extension))

(n.) 電子メールの送受信 API。アプリケーションコードは、Java Naming and Directory Interface™ (JNDI) サービスにより、JNDI 名を使って JavaMail セッションリソースオブジェクトを特定します。

JavaScript™ プログラミング言語 (JavaScript programming language)

(n.) クライアントおよびサーバーのインターネットアプリケーションを開発する際に使用する、コンパクトなオブジェクトベーススクリプト言語。

JavaServer Faces™ 変換モデル (JavaServer Faces conversion model) (JavaServer Faces イベントおよびリスナーモデル (JavaServer Faces event and listener model))

(n.) JavaServer Faces UI コンポーネントが生成する文字列ベースのマークアップとサーバー側 Java オブジェクトの間の変換メカニズム。

(n.) JavaServer Faces UI コンポーネントによってイベントが処理される方法を決定するメカニズム。このモデルは、JavaBeans コンポーネントイベントおよびリスナーモデルに基づいています。

**JavaServer Faces
式言語
(JavaServer Faces
expression
language)**

(n.) JavaServer Faces UI コンポーネントタグ属性で使用される単純な式言語。関連コンポーネントを Bean プロパティにバインドしたり、関連コンポーネントの値をメソッドや外部データソース (Bean プロパティなど) にバインドしたりするために使用します。「JSP 式言語 (JSP expression language)」による式とは異なり、JavaServer Faces EL による式は、Web コンテナではなく JavaServer Faces 実装によって評価されます。

**JavaServer Faces
ナビゲーション
モデル
(JavaServer Faces
navigation
model)**

(n.) JavaServer Faces アプリケーションでページが表示されるシーケンスを定義するメカニズム。

**JavaServer Faces
UI コンポーネン
ト (JavaServer
Faces UI
component)**

(n.) データをクライアントに出力したり、ユーザーが JavaServer Faces アプリケーションにデータを入力したりできるようにするユーザーインタフェースコントロール。

**JavaServer Faces
UI コンポーネン
トクラス
(JavaServer Faces
UI component
class)**

(n.) JavaServer Faces UI コンポーネントの動作やプロパティを定義する JavaServer Faces クラス。

**JavaServer Faces
テクノロジー
(JavaServer Faces
technology)**

(n.) Java プログラミング言語で記述された Web アプリケーションのサーバー側ユーザーインタフェースを構築するためのフレームワーク。

**JavaServer Faces
検証モデル
(JavaServer Faces
validation model)**

(n.) ユーザーが JavaServer Faces UI コンポーネントに入力するデータを検証するメカニズム。

**JavaServer
Pages™ テクノ
ロジー (JavaServer
Pages Web Start ソ
フトウェア)
(Java
Web Start
software)**

「JSP テクノロジー (JSP technology)」を参照してください。

(n.) Web アプリケーションの起動ツール。Java Web Start ソフトウェアを使用すれば、Web リンクをクリックしてアプリケーションを起動できます。アプリケーションがコンピュータにインストールされていない場合、Java Web Start は自動的にアプリケーションをダウンロードし、コンピュータのキャッシュに保存します。アプリケーションがキャッシュにダウンロードされると、デスクトップアイコンまたはブラウザのリンクからアプリケーションを起動できます。アプリケーションの起動方法に関係なく、常にアプリケーションの最新バージョンが起動されます。

JAXM

(Java API for XML Messaging) (n.) SOAP 標準を使用してアプリケーションによるドキュメント指向 XML メッセージの送受信を可能にする Java API。これらのメッセージは、ファイルが添付されていても、されていなくても送受信できます。

JAXP	(Java API for XML Processing) (n.) 「XML (拡張可能マークアップ言語)」ドキュメントを処理するためのAPI。JAXPでは、パーサー規格のSAXやDOMを使用して、イベントのストリームとしてデータを解析したり、ツリー構造表現を作成したりできるようにします。JAXPでは、「XSLT (拡張可能スタイル言語変換)」規格をサポートしているため、データの表現を制御したり、データを別のXMLドキュメントやHTMLなどの別のフォーマットに変換したりすることができます。JAXPには名前空間サポートが用意されているため、スキーマを処理する際に名前の競合を回避することができます。
JAXR	(Java API for XML Registries) (n.) さまざまな種類の「XML (拡張可能マークアップ言語)」レジストリにアクセスするための、統一された標準Java API。ユーザーは、Webサービスを作成、配備、および検出できます。「レジストリ (registry)」も参照してください。
JAXR クライアント (JAXR client)	(n.) JAXR API を使って JAXR プロバイダ経由でビジネスレジストリにアクセスするクライアントプログラム。
JAXR プロバイダ (JAXR provider)	(n.) JAXR API の実装。特定のレジストリプロバイダまたは共通の仕様に基づく一連のレジストリプロバイダへのアクセス機能を提供します。
JAX-RPC	(Java API for XML-based RPC) (n.) 「XML (拡張可能マークアップ言語)」ベースの「RPC (遠隔手続き呼び出し)」プロトコルに基づいて、開発者が相互利用可能なWebアプリケーションとWebサービスを作成するのを可能にするJava API。
JDBC™ 接続プール (JDBC connection pool)	(n.) データベースへの接続を指定するためのJDBCデータソースのプロパティと接続プールのプロパティを組み合わせるプール。
JDBC リソース (JDBC resource)	(n.) アプリケーションサーバー内で稼働しているアプリケーションを、既存のJDBC接続プールを介してデータベースに接続するために使用するリソース。アプリケーションが使用するJava Naming and Directory Interface (JNDI) 名と既存のJDBC接続プールの名前で作成されています。
JDBC テクノロジー (Java DataBase Connectivity ソフトウェア)	(JDBC technology (Java DataBase Connectivity software)) (n.) 開発者がデータ認識コンポーネントを作成するときに使う、標準ベースの一連のクラスおよびインタフェース。JDBC API は、プラットフォームやベンダーに依存せずに、データソースに接続して対話型の操作を実行するための手法を実装します。JDBC テクノロジーでは、SQLベースのデータベースアクセス用に呼び出しレベルのAPIを提供します。
JHTML (J-Sky ハイパーテキストマークアップ言語)	(JTHML (J-Sky hypertext markup language)) 日本語 J-Sky デバイスをプログラムするためのVodafoneの独自言語。
JMS	(Java Message Service) (n.) (n.) メッセージサービス機能へのJavaクライアントのアクセス方法について定義するインタフェースおよびセマンティックの標準セット。これ

らのインタフェースを利用することにより、Java プログラミング言語で記述されたプログラムで、メッセージの作成、送信、受信、および読み取りを行うことができます。

JMS 管理対象オブジェクト (JMS administered object)

(JMS-administered object (Java Message Service administered object)) (n.) 1 つ以上の JMS クライアント用に管理者が作成する、あらかじめ設定された Java Message Service オブジェクト (「[JMS 接続ファクトリ \(Java Message Service 接続ファクトリ\)](#)」または「[JMS 送信先 \(Java Message Service 送信先\)](#)」)。管理オブジェクトを使用することにより、JMS クライアントがプロバイダの独自性から切り離すことができるので、クライアントをプロバイダ依存しないようにすることができます。これらのオブジェクトは、管理者によって Java Naming and Directory Interface (JNDI) 名前空間に配置されます。JMS クライアントは、JNDI 検索を使用してそれらのオブジェクトにアクセスします。

JMS API

(Java Message Service API) (n.) JMS クライアントが JMS メッセージサービスの機能にアクセスする方法を定義するインタフェースとセマンティックの標準セット。これらのインタフェースを利用することにより、Java プログラミング言語で記述されたプログラムで、メッセージの作成、送信、受信、および読み取りを行うことができます。

JMS アプリケーション (Java Message Service アプリケーション)

(JMS application (Java Message Service application)) (n.) メッセージを交換する 1 つ以上の JMS クライアント。

JMS クライアント (Java Message Service クライアント)

(JMS client (Java Message Service client)) (n.) JMS メッセージサービスを使ってメッセージを交換する別の JMS クライアントと通信するアプリケーションまたはソフトウェアコンポーネント。

JMS 接続ファクトリ (Java Message Service 接続ファクトリ)

(JMS connection factory (Java Message Service connection factory)) (n.) JMS クライアントが JMS メッセージサービスへの接続を確立するために使用する Java Message Service によって管理されるオブジェクト。

JMS 送信先 (Java Message Service 送信先)

(JMS destination (Java Message Service destination)) (n.) 作成されたメッセージが配信される JMS メッセージサービスの物理的な送信先。メッセージはそこでルーティング処理されたあと、コンシューマに向けて配信されます。この物理的な送信先は、JMS クライアントが送受信されるメッセージの送信先を指定するために使用する、JMS 管理によるオブジェクトによって識別およびカプセル化されます。

JMS メッセージ (Java Message Service メッセージ)

(JMS messages (Java Message Service messages)) (n.) Java Message Service のクライアントによってコンシュームされる非同期要求、レポート、またはイベント。メッセージは、ヘッダーと本体から構成されており、ヘッダーにはフィールドを追加できません。メッセージのヘッダーでは、標準フィールドとオプションのプロパティを指定します。送信中のデータはメッセージ本体に含まれています。

JMS プロバイダ (Java Message Service プロバイダ)	(JMS provider (Java Message Service provider)) (n.) メッセージングシステムの JMS インタフェースを実装し、製品全体に必要な管理および制御機能を追加する製品。
JMS サービス (Java Message Service サービス)	(JMS service (Java Message Service service)) (n.) JMS クライアントへの接続、メッセージのルーティングおよび配信、持続性、セキュリティ、ロギングなど、Java Message Service のメッセージングシステム用の配信サービスを提供するソフトウェア。 メッセージサービスは、JMS クライアントのメッセージ送信先、およびメッセージを消費するクライアントに配信されるメッセージの送信元である物理的送信先を維持します。
JMS セッション (Java Message Service セッション)	(JMS session (Java Message Service session)) (n.) JMS メッセージを送受信するためのシングルスレッドのコンテキスト。JMS セッションは、非トランザクションやローカルトランザクションである場合や、分散トランザクションに参加する場合があります。
JNDI 拡張 (Java Naming and Directory Interface 拡張)	(JNDI extension (Java Naming and Directory Interface extension)) (n.) 企業の複数の名前サービスやディレクトリサービスに対する統一インタフェースを Java テクノロジー対応のアプリケーションに提供する、Java プラットフォームの標準拡張。Java Enterprise API セットの一部として、JNDI は、企業の異種名前サービスおよび異種ディレクトリサービスへの接続を可能にします。
JNDI 名 (Java Naming and Directory Interface 名)	(JNDI name (Java Naming and Directory Interface name)) (n.) JNDI 名前サービスに登録されているリソースへのアクセスに使用する名前。
ジョブコントローラ (job controller)	(n.) ほかのさまざまな MTA コンポーネントの要求に応じてタスクをスケジュールおよび実行する「 MTA (メッセージ転送エージェント) 」コンポーネント。
ジョインルール (join rule)	(n.) Directory Proxy Server 二次データビューのエントリと一次データビューのエントリをリンクする方法や、SQL テーブルのエントリと別の SQL テーブルのエントリをリンクする方法を指定するルール。
jspx ユティリティ (jspx utility)	(n.) 「 JSP テクノロジー (JSP technology) 」によって作成されたページのコンパイラ。このユティリティで、JSP 仕様に準拠しているかどうかについて、すべての JSP ページをチェックします。
JSP™ アクション (JSP action)	(n.) 暗黙オブジェクトやその他のサーバー側オブジェクトで動作可能な JSP 要素。または、新しいスクリプト変数を定義可能な JSP 要素。アクションは、要素の XML 構文に従い、開始タグ、本体、終了タグがあります。本体が空の場合は、空白タグの構文を使用することもできます。タグでは、接頭辞を使用する必要があります。標準アクションとカスタムアクションがあります。

JSP コンテナ (JSP container)	(n.) サブレットコンテナと同じサービスと、JSP ページを解釈して「サブレット (servlet)」へと処理するエンジンを提供する「コンテナ (container)」。
JSP コンテナ、分散 (JSP container, distributed)	(n.) 分散可能であるとタグ付けされ、異なるホストで実行している可能性のある複数の Java 仮想マシンに分散している Web アプリケーションを実行できる JSP コンテナ。
JSP カスタムアクション (JSP custom action)	(n.) タグライブラリ記述子を使用して移植可能な方法で記述された、ユーザー定義のアクション。taglib 指令によって JSP ページにインポートされます。カスタムアクションは、記述している JSP ページで繰り返し実行されるタスクをカプセル化するために使用されます。
JSP カスタムタグ (JSP custom tag)	(n.) JSP カスタムアクションを参照する「タグ (tag)」。
JSP 宣言 (JSP declaration)	(n.) メソッド、変数、またはその両方を JSP ページで宣言する JSP スクリプト要素。
JSP 指示子 (JSP directive)	(n.) JSP コンテナに対して指示を与える JSP 要素。変換時に解釈されます。
JSP ドキュメント (JSP document)	(n.) XML 構文で記述され、XML ドキュメントの制約に従う JSP ページ。
JSP 要素 (JSP element)	(n.) JSP トランスレータで認識される JSP ページの部分。要素は、指令、アクション、またはスクリプト要素です。
JSP 式 (JSP expression)	(n.) 有効なスクリプト言語式を含むスクリプト要素。スクリプト言語式は、評価されて String に変換され、暗黙的な out オブジェクトに配置されます。
JSP 式言語 (JSP expression language)	(n.) JavaBeans コンポーネントのプロパティにアクセスする式を記述するために使用する言語。EL 式は、静的テキスト、または式が使用可能なあらゆる標準タグ属性やカスタムタグ属性でも使用できます。
JSP ページ (JSP page)	(n.) 静的テキストや、要求を処理して応答を作成する方法を記述する JSP 要素を含む、テキストベースのドキュメント。JSP ページは「サブレット (servlet)」に変換されて要求が処理されます。
JSP スクリプト要素 (scripting element)	(n.) 構文が JSP 仕様で定義され、JSP ページで使用されているスクリプト言語に従ってコンテンツが記述されている JSP 宣言、スクリプトレット、または式。JSP 仕様では、言語ページ属性が "java" である場合の構文やセマンティクスが説明されています。
JSP スクリプトレット (JSP scriptlet)	(n.) JSP ページで使用しているスクリプト言語で有効なコードの断片を含む JSP スクリプト要素。JSP 仕様では、言語ページ属性が "java" である場合の有効なスクリプトレットが説明されています。

JSP 標準アクション (JSP standard action)	(n.) JSP 仕様で定義されていて、常に JSP ページで使用できるアクション。
JSP タグファイル (JSP tag file)	(n.) JSP コードの再使用可能な断片を含むソースファイル。この断片は、JSP ページがサブレットに変換されるときに、タグハンドラに変換されます。
JSP タグハンドラ (JSP tag handler)	(n.) カスタムタグの動作を実装する Java プログラミング言語オブジェクト。
JSP タグライブラリ (JSP tag library)	(n.) タグライブラリ記述子と Java クラスを使用して記述されたカスタムタグのコレクション。「 JSTL 」も参照してください。
JSP™ テクノロジ (JSP technology)	<p>(1) (n.) 静的データ、JSP 要素、サーバー側 Java オブジェクトを使用してクライアントの動的コンテンツを生成する、拡張可能な Web テクノロジ。通常、静的データは HTML 要素または XML 要素で、多くの場合、クライアントは Web ブラウザです。JSP テクノロジによって作成されたページは、標準ブラウザページのレイアウト機能とプログラミング言語の能力をあわせ持っています。</p> <p>(2) (n.) インスタンス化、初期化、破棄、ほかのコンポーネントからのアクセス、および設定管理を含む、すべての JSP テクノロジのメタ関数を可能にする拡張機能。Web ブラウザ内ではなく、Web サーバー上で実行される再利用可能な Java アプリケーションです。</p>
JSS	「 Network Security Services for Java (JSS) 」を参照してください。
JSEE	(Java Secure Socket Extension) (n.) セキュリティー保護されたインターネット通信を可能にするパッケージのセット。
JSTL	(JavaServer Pages Standard Tag Library) (n.) 多くの JSP アプリケーションに共通するコア機能をカプセル化したタグライブラリ。JSTL は、一般的な構造化タスク (繰り返しや条件付き)、XML ドキュメントの操作タグ、国際化およびロケール固有のフォーマットタグ、SQL タグ、および関数をサポートします。
JTA (Java トランザクション API)	(JTA (Java transaction API)) (n.) アプリケーションおよび J2EE サーバーがトランザクションにアクセスするための API。
JTS (Java トランザクションサービス)	(JTS (Java transaction service)) (n.) JTA をサポートし、API より下のレベルで Object Management Group Object Transaction Service 1.1 仕様の Java マッピングを実装するトランザクションマネージャーの実装を指定します。

K

- 鍵データベース (key database)** (n.) サーバーの証明書用の鍵ペアを記述したファイル。鍵ファイルとも呼ばれます。
- 鍵ペアファイル (key-pair file)** 「[信頼データベース \(trust database\)](#)」を参照してください。
- キーストア (keystore)** (n.) 認証に使用する鍵および証明書を記述したファイル。
- 知識情報 (knowledge information)** (n.) ディレクトリサービスインフラストラクチャー情報の一部。Directory Server は、知識情報を使用して、情報要求をほかのサーバーに渡します。
- 知識参照 (knowledge reference)** (n.) 異なるデータベースに保存されるディレクトリ情報のポインタ。

L

**最終変更
ヘッダー
(last-modified
header)**

(n.) HTTP 応答でサーバーから返されたドキュメントファイルの最終変更時刻。

**LDAP
(Lightweight
Directory Access
Protocol)**

(LDAP (Lightweight Directory Access Protocol)) (n.) TCP/IP を介して複数のプラットフォーム間で動作するように設計されたディレクトリサービスプロトコル。X.500 Directory Access Protocol (DAP) を簡素化したもので、ユーザープロファイル、配布リスト、Sun Java System サーバー群全体の設定データなどの情報の格納、検索、および配布を単一の場所で管理できるようにします。Directory Server は、LDAP プロトコルを使用します。

**LDAP データ
ベース (LDAP
database)**

(n.) 認証で使用するユーザーとグループのリストを保管するデータベース。

**LDAP データ交
換形式 (LDAP
Data Interchange
Format)**

「[LDIF \(LDAP データ交換形式\)](#)」を参照してください。

**LDAP フィルタ
(LDAP filter)**

(n.) 特定の属性または属性値の有無に基づいて一連のエントリを指定する方法。

**LDAP リフェラ
ル (LDAP
referrals)**

(n.) 別の LDAP エントリへのシンボリックリンク (リフェラル) から成る LDAP エントリ。LDAP リフェラルは、LDAP ホスト名と識別名で構成されます。通常、LDAP リフェラルは、データをレプリケートせずに、既存の LDAP データを参照するために使用されます。また LDAP リフェラルは、移動された可能性のある特定のエンタリに依存するプログラムの互換性を維持するためにも使用されます。

**LDAP 検索文字
列 (LDAP search
string)**

(n.) ディレクトリ検索に使用される属性を定義する、置換可能なパラメータを含む文字列。たとえば、「uid=%s」という LDAP 検索文字列は、ユーザー ID 属性に基づく検索であることを意味します。

LDAP サーバー (LDAP server)	(n.) LDAP ディレクトリを管理し、そのディレクトリに対するクエリー (問い合わせ) サービスを提供するソフトウェアサーバー。Sun Directory Services および Netscape Directory Services は、LDAP Server の実装です。
LDAP サーバー フェイルオーバー (LDAP server failover)	(n.) LDAP サーバーのバックアップ機能。1つのLDAPサーバーに障害が発生した場合、システムは、別のLDAPサーバーに切り替えることができます。
LDAP URL	(n.) 「DNS (ドメインネームシステム)」を使用してディレクトリサーバーの位置を特定し、LDAP によってクエリーを完了する手段となるURL。たとえばLDAP URLは、 <code>ldap://ldap.example.com</code> のようなURLになります。
LDAPv3	(n.) 「LDAPv3」プロトコルのバージョン3。
LDBM	(n.) LDAP データベースマネージャー。
LDBM データベース (LDBM database)	(n.) Directory Server のすべてのデータを記述したサイズの大きいファイルセットで構成される、高性能ディスクベースデータベース。
LDIF (LDAP データ交換形式)	(LDIF (LDAP Data Interchange Format)) (n.) <i>type:value</i> ペアを使用して、Directory Server のエントリをテキスト形式で表現するために使用する形式。
リーフエントリ (leaf entry)	(n.) それより下位にほかのエントリがないエントリ。リーフエントリは、ディレクトリツリーの分岐点にはなりません。
Legato NetWorker® ソフトウェア (Legato NetWorker software) レベル (level)	(n.) Legato Systems, Inc. が提供するサードパーティー製バックアップユーティリティ。 (n.) ログの詳細度の指定。ログファイルに記録するイベントの種類の相対的な数を意味します。たとえば、緊急レベル (SEVERE) では、ログに記録されるイベントはほとんどありません。情報レベル (INFO) では、多くのイベントがログに記録されます。
Liberty 対応クライアント (Liberty-enabled client)	(n.) Liberty 対応クライアントは、主体がサービスプロバイダに対して認証を行う場合に使用するアイデンティティプロバイダに関する情報を所有している、または入手方法を認識しているクライアントです。
Liberty 対応プロキシ (Liberty-enabled proxy) ライフサイクル (life cycle)	(n.) Liberty 対応プロキシは、Liberty 対応クライアントをエミュレートする HTTP プロキシです。 (1) (n.) J2EE コンポーネントの存在に伴って随時発生する一連のイベント。各種コンポーネントは随時特定のイベントを保持し、それによってコンポーネントの可用性の状態の推移を表します。たとえば、サーブレットは作成時にサーブレットのコン

テナによって呼び出される `init` メソッドを保持し、その後サーブレットが機能するために必要なクライアントや他のサーブレットによって `service` メソッドが呼び出されます。`init` メソッドが呼び出された後、サーブレットは所定の機能を実行するためのデータを保持して準備が整います。サーブレットが終了する前には、終了に関連した処理を実行してリソースを解放できるように `destroy` メソッドがコンテナによって呼び出されます。この例の `init` メソッドと `destroy` メソッドは「[コールバックメソッド \(callback method\)](#)」です。すべての J2EE コンポーネントタイプ(エンタープライズ Bean、Web コンポーネント (サーブレットまたは JSP ページ)、アプリレット、およびアプリケーションクライアント)のライフサイクルについても同様です。

(2) (n.) JavaServer Faces ページの要求を受け取り、このページを表す UI コンポーネントツリーが処理され、応答が生成されるまでの一連のフェーズ。

(3) (n.) 起動からシャットダウンまでのサーバーのランタイムに伴って随時発生する一連のイベント。

ライフサイクル
イベント
(life-cycle event)

(n.) 起動や停止など、サーバーのライフサイクルの各段階。

ライフサイクル
モジュール
(life-cycle
module)
ライトウェイト
ディレクトリア
クセスプロトコ
ル(Lightweight
Directory Access
Protocol)

(n.) サーバーライフサイクルのイベントに応じて、タスクを待機し、実行するモジュール。

「[LDAP \(Lightweight Directory Access Protocol\)](#)」を参照してください。

リスナー
(listener)

(n.) ポストするオブジェクトに登録され、イベント発生時の処理を指示するクラス。

待機ポート
(listen port)

(n.) サーバーがクライアントやその他のサーバーと通信するために使用するポート。

待機ソケット
(listen socket)

(n.) 「[ポート \(port\)](#)」番号と「[IP アドレス \(IP address\)](#)」の組み合わせ。サーバーとクライアントの間の接続は、待機ソケット上で確立されます。

LMTP (ローカル
メール転送プロ
トコル)

(LMTP (Local Mail Transfer Protocol)) (n.) 「[SMTP](#)」に似ていますが、メール配信キューを管理する必要がありません。また、SMTP ではメッセージに対してステータスコードを1つしか提供しませんが、LMTP ではメッセージの各受取人のステータスコードを提供します。RFC 2033 に定義されています。

ロードバランサ
(load balancer)

(n.) 複数のゲートウェイマシンへの接続を制御して、稼働システムそれぞれにかかる負荷の均等化を計るソフトウェア。

負荷分散 (load balancing)	(n.) アプリケーション負荷をクラスタ内のノード全体に分配して、クライアント要求が適宜サービスされるようにするプロセス。スケラブルサービスにのみ適用される。
ローカルデータベース接続 (local database connection)	(n.) ローカル接続のトランザクションコンテキストは現在のプロセスおよびデータソースに対してローカルであり、複数のプロセスまたはデータソース全体に分散されません。
ロケール (locale)	(n.) 特定の地域、文化圏、または習慣を持つユーザーにデータを表示するために使用する、照合順序、文字タイプ、通貨形式、および日時形式の設定。ロケールには、所定の言語のデータを解釈、保存、または照合する方法も設定します。またロケールにより、所定の言語を表現するために使用するコードページも指定します。
ローカルインタフェース (local interface)	(n.) 同じ Java™ 仮想マシン (JVM™ マシン) にあるクライアントのメカニズムに、Bean にアクセスするためのセッションやエンティティ Bean を提供するインタフェース。
ローカルメール転送プロトコル (Local Mail Transfer Protocol)	「 LMTP (ローカルメール転送プロトコル) 」を参照してください。
ローカル部分 (local part)	(n.) 電子メールアドレスの受取人を識別する部分。「 ドメイン部分 (domain part) 」も参照してください。
ローカルセッション (local session)	(n.) 1つのサーバーだけに見えるユーザーセッション。
ローカルサブセット (local subset)	(n.) DTD (ドキュメントタイプ定義) のうち、現在の XML (拡張可能マークアップ言語) ファイル内で定義されている部分。
ローカルトランザクション (local transaction)	(n.) 1つのデータベースに固有で、1つのプロセス内に制限されるトランザクション。ローカルトランザクションは、1つのバックエンドに対してのみ動作します。ローカルトランザクションは通常、JDBC™ API を使用して区別されます。「 グローバルトランザクション (global transaction) 」も参照してください。
ログディレクトリ (log directory)	(n.) サービスのすべてのログファイルが保存されているディレクトリ。
ログ有効期限 (log expiration)	(n.) 有効期間が過ぎたログファイルは、ログディレクトリから削除されます。
論理アーキテクチャ (logical architecture)	(n.) 分散型アプリケーションの論理的な構築ブロック、およびこれらの構築ブロック間の関係(またはインタフェース)を示す設計。論理アーキテクチャには、分散「 アプリケーションコンポーネント (application component) 」およびそれらをサポートするために必要なインフラストラクチャーサービスのコンポーネントの両方が含まれます。

論理ホスト
(logical host)

(n.) Messaging Server 2.0 (最小) の概念。アプリケーション、アプリケーションデータが存在するディスクセットまたはディスクグループ、およびクラスタにアクセスするために使用するネットワークアドレスなどが含まれています。この概念は、SunPlex™ システムには適用されなくなっています。

ログローテーション (log rotation)

(n.) 現在のログファイルとして新しいログファイルを作成すること。以後のログイベントは、新しいログファイルに書き込まれます。以前のログファイルはログディレクトリ内に残りますが、ログが書き込まれることはありません。

ルックアップ (lookup)

(n.) 検索の同義語。特定のパラメータを使ってデータをソートします。

M

- メールボックス (mailbox) (n.) メッセージの格納と表示を行う場所。「フォルダ (folder)」も参照してください。
- メールクライアント (mail client) (n.) ユーザーが電子メールを送受信する際に利用するプログラム。メールクライアントは、さまざまなネットワークやメールプログラムの一部で、ユーザーがもっとも頻繁に使用します。メールクライアントは、配信するメッセージを作成して送信し、新たに受信したメールを確認し、受信メールを受理して整理します。
- メール交換レコード (mail exchange record) 「MX レコード (メール交換レコード)」を参照してください。
- メーリングリスト (mailing list) 「メーリングリスト (mail list)」を参照してください。
- メーリングリスト所有者 (mailing list owner) 「メーリングリスト所有者 (mail list owner)」を参照してください。
- メールリスト (mail list) (n.) 電子メールアドレスのリスト。メールリストのアドレスを指定することによってリストの電子メールアドレス宛てにメッセージを送信できます。グループとも呼ばれます。
- メールリスト所有者 (mail list owner) (n.) メールリストのメンバーの追加と削除を行う管理権限を持つユーザー。
- メールリレー (mail relay) (n.) 「ユーザーアカウント (user account)」または「MTA (メッセージ転送エージェント)」からのメールを受け取り、そのメールを受取人のメッセージストアや別のルーターに中継するメールサーバー。
- メールルーター (mail router) 「メールリレー (mail relay)」を参照してください。

管理対象 Bean
作成機能
(managed bean
creation facility)

(n.) 「[JavaServer Faces テクノロジー \(JavaServer Faces technology\)](#)」 アプリケーション内で使用される 「[JavaBeans コンポーネント \(JavaBeans component\)](#)」 の特性を定義するメカニズム。

管理対象オブ
ジェクト
(managed object)

(n.) 「[MIB \(管理情報ベース\)](#)」 の一部を構成する 「[SNMP](#)」 データ要素。Directory Server では、管理対象オブジェクトが cn=monitor に保持され、SNMP エージェントはネットワーク管理ステーションにそのオブジェクトを提供します。LDAP 属性の場合同様、各管理対象オブジェクトには、ドット表記法の名前とオブジェクト識別子があります。

管理されている
ロール
(managed role)

(n.) メンバーの明示的な列挙リストを作成することができます。

管理情報ベース
(management
information
base)
管理ルール
(management
rule)

「[MTA \(メッセージ転送エージェント\)](#)」 を参照してください。

(n.) 自己調節、自己設定、または自己回復のカスタムアクションを Application Server のトリガーイベントに関連付けます。「[イベント \(event\)](#)」 も参照してください。

マッピング
(mapping)

(1) (n.) オブジェクト指向モデルを、データのリレーショナルモデル (通常はリレーショナルデータベースのスキーマ) に結びつける機能。スキーマを別の構造に変換するプロセスです。

(2) (n.) 使用ユーザーとセキュリティーロールのマッピング。

マッピングツ
リー (mapping
tree)

(n.) サフィックス (サブツリー) の名前とデータベースを関連付けたデータ構造。

マスターエー
ジェント
(master agen)

「[SNMP マスターエージェント \(SNMP master agent\)](#)」 を参照してください。

マスターチャ
ネルプログラム
(master channel
program)

(n.) リモートシステムへの転送を開始するチャンネルプログラム。「[スレーブチャンネルプログラム \(slave channel program\)](#)」 も参照してください。

マスターディ
レクトリサー
バー
(master directory
server)

(n.) レプリケートされるデータが保持される読み書きディレクトリサーバー。

マッチングカ
テゴリ (matching
category)

(n.) 検索送信の結果として返される、検索クエリーに一致するカテゴリ。

マッチングド
キュメント
(matching
document)

(n.) 検索送信の結果として返される、検索クエリーに一致するドキュメント。

マッチングルール (matching rule)	(n.) 検索操作時に、サーバーが文字列の比較を実行する方法のガイドライン。国際検索では、マッチングルールにより照会順序と使用する演算子をサーバーに指示します。
MD5	(n.) RSA Data Security によって提供されるメッセージダイジェストアルゴリズム。MD5を使用すると、高い確率で一意になる短い形式のダイジェストデータを生成できます。
MD5 シグニチャー (MD5 signature)	(n.) 「 MD5 」アルゴリズムによって生成されるメッセージダイジェスト。
MDB (メッセージ駆動型 Bean)	(MDB (message-driven bean)) (n.) 非同期メッセージコンシューマであるエンタープライズ Bean。メッセージ駆動型 Bean には特定のクライアントの状態はありませんが、Bean のインスタンス変数には EJB™ アーキテクチャーに基づくオブジェクトへのオープンデータベース接続やオブジェクト参照などのクライアントメッセージの処理全体の状態が入っていることがあります。クライアントは、メッセージ駆動型 Bean がメッセージリスナとなっている宛先にメッセージを送信して、メッセージ駆動型 Bean にアクセスします。
メンバー (member)	(n.) メールリスト宛ての電子メールのコピーを受け取るユーザーまたはグループ。「 メールリスト (mail list) 」、「 展開 (expansion) 」、「 モデレータ (moderator) 」も参照してください。
メッセージ (message)	(1) (n.) 電子メールの基本単位。メッセージは、「 ヘッダー (header) 」と「 本体 (body) 」で構成され、多くの場合、差出人から受取人に転送される間は「 エンベロープ (envelope) 」に格納されます。 (2) (n.) Java Message Service で、「 JMS 」クライアントによって利用される非同期の要求、レポート、またはイベント。メッセージは、ヘッダーと本体から構成されており、ヘッダーにはフィールドを追加できます。メッセージのヘッダーでは、標準フィールドとオプションのプロパティを指定します。送信中のデータはメッセージ本体に含まれています。メッセージには、エンタープライズアプリケーションの調整に必要な重要な情報が含まれ、特定のビジネスアクションを記述する明確に書式設定されたデータの形で提供されます。
メッセージアクセスサービス (message access services)	(n.) Messaging Server メッセージストアへのクライアントアクセスをサポートするプロトコルサーバー、ソフトウェアドライバ、およびライブラリ。
メッセージコンシューマ (message consumer)	(n.) 「 JMS セッション (Java Message Service セッション) 」によって作成されるオブジェクトで、「 送信先 (destination) 」に送信された「 メッセージ (message) 」を受信する際に使用されます。
メッセージ配信 (message delivery)	(n.) 「 MTA (メッセージ転送エージェント) 」がメッセージをローカルの受取人(メールフォルダまたはプログラム)に配信する動作。

メッセージ駆動型 Bean (message-driven bean)

「MDB (メッセージ駆動型 Bean)」を参照してください。

メッセージ転送 (message forwarding)

(n.) 「MTA (メッセージ転送エージェント)」が、特定のアカウントに配信されたメッセージを、そのアカウントの属性で指定された1つまたは複数の新しい送信先に送信する動作。転送は、ユーザーが設定することもできます。「メッセージ配信 (message delivery)」、「メッセージルーティング (message routing)」も参照してください。

メッセージ処理システム (message handling system)

「MHS (メッセージ処理システム)」を参照してください。

メッセージプロデューサー (message producer)

(n.) 「JMS セッション (Java Message Service セッション)」によって作成されるオブジェクトで、「メッセージ (message)」を「送信先 (destination)」に送信する際に使用されます。

Message Queue

(n.) Java™ Message Service (JMS) オープン標準を実装するメッセージングシステム。Sun Java System Message Queue は JMS プロバイダです。

メッセージキュー (message queue)

(n.) クライアントやほかのメールサーバーから受け取ったメッセージを即時配信または据え置き配信するために保管するディレクトリ。

Message Queue クライアントランタイム (Message Queue client runtime)

(n.) JMS クライアントに Java Enterprise System メッセージサーバーへのインタフェースを提供するソフトウェア。クライアントランタイムは、クライアントが送信先にメッセージを送信し、送信先からメッセージを受信するために必要なすべての操作をサポートします。

Message Queue メッセージサーバー (Message Queue message server)

(n.) JMS クライアントへの接続、メッセージのルーティングと配信、持続性、セキュリティ、ロギングなどの Message Queue メッセージングシステム用の配信サービスを提供するソフトウェア。メッセージサーバーは、JMS クライアントのメッセージ送信先であり、消費する側のクライアントへのメッセージ配信元である、物理的な送信先を維持します。

メッセージ制限容量 (message quota)

(n.) 特定のフォルダが消費できるディスク容量を定義する制限。

メッセージルーティング (message routing)

(n.) 最初の MTA が、受取人がローカルアカウントではなくほかの場所にいると判断したときに、別の「MTA (メッセージ転送エージェント)」にメッセージを転送する動作。通常、ルーティングを設定できるのはネットワーク管理者だけです。「メッセージ転送 (message forwarding)」も参照してください。

メッセージセレクト (message selector)	(n.) JMS メッセージヘッダーのプロパティ値 (セレクト) に基づいて、メッセージを選択するコンシューマのための手段。メッセージサービスは、メッセージセレクトに配置される条件に基づいて、メッセージのフィルタやルーティングを行います。
メッセージサービス (message service)	「 Message Queue メッセージサーバー (Message Queue message server) 」を参照してください。
メッセージストア (message store)	(n.) Messaging Server インスタンスに対してローカルに配信されたすべてのメッセージのデータベース。メッセージは、単一の物理ディスクに格納することも、複数の物理ディスクに格納することもできます。
メッセージストア管理者 (message store administrator)	(n.) Message Server のメッセージストアを管理する管理権限を持つユーザー。このユーザーは、メールボックスの表示と監視、およびストアへのアクセス制御の指定を行うことができます。プロキシ認証の権限を使用して、ストアを管理するための特定のユーティリティーを実行できます。
メッセージストアパーティション (message store partition)	(n.) 単一の物理ファイルシステムパーティション上に置かれたメッセージストアまたはメッセージストアのサブセット。
メッセージ送信 (message submission)	(n.) クライアントの「 userAgent 」は、メールサーバーにメッセージを転送し、配信を要求します。
メッセージ転送エージェント (message transfer agent)	「 MTA (メッセージ転送エージェント) 」を参照してください。
メッセージング (messaging)	(n.) エンタープライズアプリケーションで使用される非同期要求、レポート、またはイベントのシステム。これにより、緩やかに結合されたアプリケーションが情報を確実かつ安全に送信できます。
Messaging Multiplexor	「 MMP 」を参照してください。
Messaging Server 管理者 (Messaging Server administrator)	(n.) Messaging Server インスタンスのインストールや管理などの権限を持つ管理者。
Messaging Server ベースディレクトリ (messaging server base directory)	(n.) 特定のホスト上にある Administration Server に関連付けられたすべてのサーバーがインストールされているディレクトリ。通常は、 <code>msg_svr_base</code> と指定します。「 インストールディレクトリ (installation directory) 」も参照してください。
Messenger Express	(n.) ユーザーがブラウザベース (HTTP) のインタフェースを介してメールボックスにアクセスできるようにするメールクライアント。メッセージ、フォルダ、その他の

メールボックス情報は、HTML形式でブラウザのウィンドウに表示されます。
「[Web メール \(webmail\)](#)」も参照してください。

Messenger Express Multiplexor

(n.) マルチプレクサとして動作するプロキシメッセージングサーバー。このサーバーにより、Messaging Server (Messenger Express) の HTTP サービスに接続できます。Messenger Express Multiplexor を使用すると、複数のサーバーマシンにユーザーを分散できるようになります。

メタデータ

(n.) コンポーネントの名前やそのコンポーネントの動作仕様などの、コンポーネントに関する情報。「[配備記述子 \(deployment descriptor\)](#)」も参照してください。

メタデバイス状態データベースのレプリカ (metadevice state database replica)

(n.) すべてのメタデバイスの設定と状態、およびエラー状況を記録した、ディスクに保管されているデータベース。この情報は、Solstice DiskSuite™ ソフトウェアディスクセットの適切な処理にとって重要です。

メタ情報 (metainformation)

(n.) 作成者の名前、ドキュメントのタイトル、作成日などの、リソースに関する情報。検索エンジンロボットは、リソース記述を作成するときに、メタ情報とドキュメントの内容を使用します。

メソッドバインド式 (method-binding expression)

(n.) バッキング Bean のメソッドを参照する「[JavaServer Faces 式言語 \(JavaServer Faces expression language\)](#)」の式。このメソッドは、メソッドバインディング式を使用するタグを持つ UI コンポーネントのイベント処理、検証、またはナビゲーション処理のいずれかを実行します。

メソッドアクセス権 (method permission)

(n.) どのユーザーに対して1つまたは複数の「[エンタープライズ Bean \(enterprise bean\)](#)」メソッドの実行が許可されているかを判別する「[承認 \(authorization\)](#)」ルール。

MHS (メッセージ処理システム)

(MHS (message handling system)) (n.) 接続されている「[URL マッピング \(URL mapping\)](#)」、ユーザーエージェント、およびメッセージストアのグループ。

MIB (管理情報ベース)

(MIB (management information base)) (n.) 「[SNMP マスターエージェント \(SNMP master agent\)](#)」がアクセス可能な変数を定義するツリー型構造。MIBにより、HTTPサーバーのネットワーク設定、状態、および統計情報へアクセスできます。SNMPを使用すると、これらの情報を「[NMS \(ネットワーク管理ステーション\)](#)」から確認できます。「[AUTH](#)」も参照してください。

移行

(n.) データ設定やカスタマイズなどのデータファイルを、製品の1つのバージョンから別のバージョンに移すプロセス。

MIME (multipurpose internet mail extensions)

(MIME (multipurpose internet mail extensions)) (n.) マルチメディア電子メールとメッセージの新たな標準。電子メールメッセージ内にマルチメディアファイルを追加できるようにするために使用されるプロトコルです。

MIME データタイプ (MIME Data Type)	(n.) MIME タイプは、システムがサポートするマルチメディアファイルのタイプを制御します。
mime.types ファイル (mime.types file)	(n.) MIME タイプの設定ファイル。このファイルは、ファイル拡張子を MIME タイプにマッピングし、要求されているコンテンツのタイプをサーバーが判別できるようにします。たとえば、拡張子が .html のリソース要求は、クライアントが HTML ファイルを要求していることを示し、拡張子が .gif のリソース要求は、クライアントが GIF 形式のイメージファイルを要求していることを示します。
ミラーノード (mirror node)	(n.) 別のアクティブノードと同じデータを持っていて、ほかのデータ冗長ユニットに常駐しているアクティブ HADB ノード。各アクティブノードには1つのミラーノードが必要であるため、ノードはペアで使用されます。ノードは、ミラーノードで障害が発生したことを検出すると、障害ノードのロールを引き継いでサービスを継続します。「HADB」、「アクティブノード (active node)」、「スペアノード (spare node)」、および「データ冗長ユニット (DRU)」も参照してください。
混合コンテンツモデル (mixed-content model)	(n.) 要素をテキストとその他の1つ以上の要素との混合として定義する「 DTD (ドキュメントタイプ定義) 」仕様。この仕様は #PCDATA で始まり、その次に様々な要素が続き、末尾が「0 またはそれ以上の」アスタリスク記号 (*) である必要があります。
MMP	(Messaging Multiplexor) (n.) 複数のメールサーバーに対する単一接続ポイントとして機能し、複数のメールボックスホストを利用する多数のユーザーへの配信を円滑に行うための専用 Messaging Server。
モバイルアプリケーション設定 (mobile application configuration)	(n.) アドレス帳、カレンダー、およびメールアプリケーションを設定してモバイルデバイスに配信できるようにする、Access Manager のサービス。
モバイルクライアントタイプ (mobile client type)	「 *クライアントタイプ (*client type) 」を参照してください。
モバイルデバイス (mobile device)	(n.) 携帯電話や携帯情報端末などの持ち運び可能なワイヤレスデバイス。
モバイルデバイスリンク (mobile device link)	(n.) ポータルデスクトップに表示されるハイパーテキストリンク。
モバイルデバイスページ (mobile device page)	(n.) ユーザーがモバイルデバイスのオプションを管理できるようにする Web ページ。
モバイルポータルデスクトップ (Mobile Portal Desktop)	(n.) モバイルデバイスに表示されるポータルデスクトップ。

モデレータ (moderator)	(n.) メッセージをメーリングリストに転送してもよいかどうかを判断するために、メーリングリストにアドレス指定されたすべての電子メールを最初に受信するユーザー。モデレータは、メーリングリストにメッセージを転送する前に編集できません。「 メールリスト (mail list) 」、「 展開 (expansion) 」、「 メンバー (member) 」も参照してください。
モジュール	(1)(n.) 「 J2EE モジュール (J2EE module) 」を参照してください。 (2)(n.) 相互に依存する、または1つのユニットとして配備され特定のサービスまたは一連のサービスを提供するために緊密に連携する Java Enterprise System サーバーのグループ。サービスモジュールは、配備アーキテクチャーで利用するためにあらかじめテストが済んでいるマルチサーバーアセンブリです。
modutil	(n.) 外部暗号化またはハードウェアアクセラレータデバイスの PKCS#11 モジュールをインストールする場合に必要なソフトウェアユーティリティ。
MTA (メッセージ転送エージェント)	(MTA (message transfer agent)) (n.) メッセージのルーティングと配信専用のプログラム。複数の MTA が連携してメッセージを転送し、目的の受取人に配信します。MTA は、メッセージをローカルのメッセージストアに配信するのか、リモート配信のために別の MTA にルーティングするのかを決定します。
MTA 設定ファイル (MTA configuration file)	(n.) Messaging Server のすべての「 チャンネル (channel) 」定義を記述した <code>imta.cnf</code> ファイルと、ルーティングのためのアドレス書き換え規則を記述した「 書き換え規則 (rewrite rule) 」。
MTA ディレクトリキャッシュ (MTA directory cache)	(n.) ユーザーおよびグループに関するディレクトリサービス情報のスナップショットで、MTA がメッセージを処理するために必要になります。「 ディレクトリ同期 (directory synchronization) 」も参照してください。
MTA ホップ (MTA hop)	(n.) 「 MTA ホップ (MTA hop) 」間でメッセージをルーティングする処理。
MUA	「 ユーザーエージェント (user agent) 」を参照してください。
多重ホームホスト (multihomed host)	(n.) 複数のパブリックネットワーク上にあるホスト。
多重ホストディスク (multihost dis)	(n.) 物理的に複数のノードに接続されたディスク。
マルチマスターレプリケーション (multimaster replication)	(n.) 書き込みまたは更新を実行する前にほかのマスターレプリカと通信する必要なしに、エントリを数個のマスターレプリカコピーのいずれかに書き込みまたは更新することができるレプリケーションモデル。各サーバーには、レプリカの更新履歴ログが残されます。1つのサーバーへの変更内容は、ほかのサーバーに自動的にレプリケートされます。競合が発生した場合は、タイムスタンプを使用して最新バージョンを保持するサーバーが判断されます。

マルチプレクサ
(multiplexor)

(n.) リモートサーバーと通信するデータベースリンクを保持しているサーバー。

**multipurpose
internet mail
extensions**
相互認証
(**mutual
authentication**)

「[MIME \(multipurpose internet mail extensions\)](#)」を参照してください。

(n.) 二者が互いの識別情報を証明し合うために使用される「[認証 \(authentication\)](#)」メカニズム。

MXレコード
(メール交換レ
コード)

(MX record (mail exchange record)) (n.) ホスト名を別のホスト名にマップする、「[DNS \(ドメインネームシステム\)](#)」レコードの一種。

N

- n+1 ディレクトリ問題 (n+1 directory problem)** (n.) タイプが異なるディレクトリとデータベースで同じ情報の複数インスタンスを管理する場合に、ハードウェアコストと人件費がかさむ問題。
- 名前の衝突 (name collision)** (n.) 複数のエントリが追加または名前変更されたあとで同じ「DN」を使用しようとする場合、レプリケーション時に発生する競合。競合エントリは、DN の一意性を確保するために、ディレクトリサーバーによって自動的に名前変更されます。
- 名前識別子 (name identifier)** (n.) 匿名性を維持するために、ユーザーのアカウント情報を多くのサービスプロバイダおよびアイデンティティプロバイダの組織全体でマッピングする際に使用される仮名。この識別子を使用することにより、アイデンティティプロバイダやサービスプロバイダは、ユーザーの実際のアイデンティティを知ることはありません。
- 名前解決 (name resolution)** (n.) 「IP アドレス (IP address)」を対応する名前にマップするプロセス。「DNS (ドメインネームシステム)」も参照してください。
- ネームスペース** (1)(n.) LDAP ディレクトリのツリー構造。「DIT」も参照してください。
(2)(n.) DTD で定義される一連の要素名に一意のラベルを指定できるようにする標準。DTD を使用するドキュメントは、要素名どうしの競合を起こさずに、ほかのドキュメントに組み込むことができます。DTD 内で定義されている要素は一意に識別されるので、たとえばパーサーは、別の DTD の要素 <name> の定義を使用せずに、使用する DTD に従って要素 <name> を解釈するタイミングを決定できます。
- ネーミング属性 (naming attribute)** (n.) 「DIT」識別名の最終属性。「**相対識別名 (relative distinguished name)**」も参照してください。
- ネーミングコンテキスト (naming context)** (1)(n.) 「DN」によって識別される「DIT」の固有サフィックス。Directory Server では、特定のタイプのディレクトリ情報がネーミングコンテキストに格納されます。

たとえば、Example Corporation のボストンオフィスのマーケティング部門の、社員すべてのエントリを格納するネーミングコンテキストは、ou=marketing,ou=Boston,o=example,c=US のようになります。

(2) (n.) 一意でユーザーにとって分かりやすい名前とリソースとの一連の関連付け。「JNDI 拡張 (Java Naming and Directory Interface 拡張)」、「JNDI 名 (Java Naming and Directory Interface 名)」、および「リソース (resource)」も参照してください。

ネーミング環境 (naming environment)

(n.) コンポーネントのソースコードへのアクセスや変更を行わずに、コンポーネントのカスタマイズを可能にするメカニズム。コンテナにはコンポーネントのネーミング環境が実装されていて、JNDI「ネーミングコンテキスト (naming context)」としてコンポーネントに提供されています。各コンポーネントは、java:comp/env JNDI コンテキストを使用して環境エントリへの命名やアクセスを行います。環境エントリはコンポーネントの配備記述子に宣言の形で指定されています。

ネイティブチャネル (native channel)

(n.) ネイティブコンテンツを表示する Portal Server チャネル。

ネイティブコンテンツ (native content)

(n.) 変換することなくクライアントに送信できる、HTML などのネイティブマークアップ言語で記述されたコンテンツ。

ネイティブデスクトップ (native desktop)

(n.) ネイティブコンテンツを表示する Portal Server デスクトップ。

NDN (非配信通知)

(NDN (nondelivery notification)) (n.) メッセージ送信時に、MTA によって「アドレス (address)」と「書き換え規則 (rewrite rule)」の間に一致が検出されない場合に、「MTA (メッセージ転送エージェント)」が元のメッセージを付けて送信側に送り返す非配信レポート。

入れ子のロール (nested role)

(n.) ほかのロール定義を指名する「ロール (role)」。入れ子のロールに含まれているロールのすべてのメンバーが、入れ子のロールのメンバーとなります。入れ子のロールでは、その適用範囲を拡張して、ほかのサブツリーにあるロールのメンバーを含めることもできます。

NetFile

(n.) Java™ テクノロジベースのファイルサーバーアプリケーション。これによりユーザーはファイルシステムにリモートアクセスできるので、ファイルとディレクトリのリモート操作が可能になります。

Netlet

(n.) Java Enterprise System Portal Server で使用される Java アプレット。Netlet を使用すると、Transmission Control Protocol/Internet Protocol に基づくすべてのアプリケーションは、認証された Portal Server 接続を介して、セキュリティー保護された状態でサーバーに接続できます。

NetMail	(n.) NetMail コンポーネントは、Java テクノロジベースクライアントの NetMail と、NetMail Lite 電子メールクライアントを実装します。これらのクライアントは、標準の IMAP サーバーと SMTP サーバーで動作します。
Netscape™ Console	(n.) Java プログラミング言語で記述されたアプリケーション。サーバー管理者が、グラフィカルインタフェースでエンタープライズネットワーク内の任意の場所から集中的にすべての Netscape サーバーを管理できます。ユーザーは、Netscape Console の任意のインストール済みインスタンスから、アクセス権限が与えられているエンタープライズネットワークに属するすべての Netscape サーバーを表示し、アクセスすることができます。
ネットワーク管理ステーション (network management station)	「 NMS (ネットワーク管理ステーション) 」を参照してください。
ネットワークマネージャー (network manager)	(n.) 「 SNMP 」データの読み取り、フォーマット、および表示を行うプログラム。SNMP クライアントとも呼ばれます。
Network Security Services for Java (JSS)	(n.) Network Security Services SSL ライブラリに Java バインディング機能を提供するクラスライブラリ。Portal Server は、このクラスライブラリを使用してサーブレットから「 Secure Socket Layer 」接続を開始し、Portal Server Secure Remote Access Pack ゲートウェイで SSL 接続を受け入れます。
ニュースチャネル (news channel)	(n.) 情報をポストおよび共有するためのフォーラム。ユーザーはニュースチャンネルに登録して更新情報を閲覧します。ニュースチャンネルの情報は、通常、URL を使用して自動的に発行されるか、適切な権限を持つユーザーによって発行されます。管理者は、ユーザーをそのユーザーが必要とするチャンネルに割り当て、ニュースチャンネルの情報を表示または発行可能なユーザーを決めることにより、ニュースチャンネルへのアクセスを制御できます。
ニュースチャネルリスト (news channel list)	(n.) 現在登録しているすべてのニュースチャンネルを表示するウィンドウ。それぞれのニュースチャンネルは、別々のタブによって表示されます。
次のホップリスト (next-hop list)	(n.) メール経路で、メッセージの転送先を判別するために使用される近接システムのリスト。次のホップリスト内のシステムの順序によって、メール経路内でシステムにメッセージが転送される順序が決まります。
NIS	(NIS (network information service)) (n.) UNIX のみ。コンピュータネットワーク全域のマシン、ユーザー、ファイルシステム、およびネットワークパラメータに関する特定の情報を、収集、照合、および共有するために、UNIX システムで使用されるプログラムとデータファイルのシステム。

NMS (ネットワーク管理ステーション)	(NMS (network management station)) (n.) 1つ以上のネットワーク管理アプリケーションがインストールされた強力なワークステーション。NMSは、ネットワークをリモート管理する場合に使用するマシンです。
NNTP (ネットワークニュース転送プロトコル)	(NNTP (Network News Transfer Protocol)) (n.) ニュースグループ用のプロトコル。ニュースサーバーホストを定義し、サーバーでエージェントサービスを使用する必要があります。
ノード	(1) (n.) コンピューティングノード。ネットワークまたはインターネット環境の数あるコンピュータの中の1つです。分散アプリケーションは、さまざまなコンピューティングノードで稼働する異なる分散コンポーネント、ビジネスサービス、およびサーバーとともに、この環境全体に配備されます。「 クラスタ 」も参照してください。 (2) (n.) 「 HADB ノード (HADB node) 」を参照してください。
ノードエージェント (node agent)	(n.) 少なくとも1つの Application Server 「 サーバーインスタンス (server instance) 」をホストするマシン(「 Domain Administration Server 」をホストするマシンを含む)ごとに必要な軽量エージェントです。ノードエージェントは、Domain Administration Server による命令に従って、Application Server インスタンスの開始、停止、作成、削除などのタスクを実行します。
非配信通知 (nondelivery notification)	「 NDN (非配信通知) 」を参照してください。
NoPassword 認証 (NoPassword authentication)	(n.) パスワードの入力を求められることなくユーザーが Access Manager にログインできるようにする認証の種類。
正規化 (normalization)	(n.) サブルーチンを使用するなどしてモジュール化により冗長性を取り除く処理、および不必要な差違を除外して共通項にまとめる処理。たとえば、システムごとに異なる行末は単一の改行に置き換えられて正規化され、複数の空白文字は1つのスペースに正規化されます。
北米産業分類システム (NAICS (North American Industry Classification System (NAICS)))	(n.) 商品の製造やサービスの提供に使用されるプロセスに基づいて産業を分類する体系。
NOTARY メッセージ (NOTARY messages)	(n.) RFC 1892 の NOTARY 仕様に準拠した非配信通知 (NDN) および配信ステータス通知。

**表記法
(notation)**

(n.) 解析対象外エンティティとして参照される非 XML ドキュメントのデータ形式を定義するメカニズム。これは SGML の名残です。名前の競合を避けるために、MIME のデータ型と名前空間を使用することがより新しい標準になっています。

通知メッセージ

(n.) Messaging Server によって送信されるメッセージの一種であり、メッセージ配信処理のステータスと、配信に関する問題や明白な障害の理由などを知らせます。このメッセージは、情報提供を目的とし、ポストマスターに対してアクションを要求するものではありません。「[配信ステータス通知 \(delivery status notification\)](#)」も参照してください。

**通知サービス
(notification
service)**

(n.) ほかのサーバーから登録や通知を受信し、特定のサブスクリバに通知を渡します。Calendar Server の csnotifyd サービスは、イベントブローカとしてイベント通知サービス (ENS) を使用してイベントと実行予定タスクの通知を送信します。

NSAPI

「[サーバープラグイン API \(server plug-in API\)](#)」を参照してください。

ns-slapd

(n.) UNIX のみ。Directory Server のすべての動作の責任を持つプロセスまたはサービス。Windows システムの「[slapd.exe](#)」に相当します。

ns-slapd.exe

(n.) Windows のみ。Windows システムのプロセスモニター。

O

- OASIS** (Organization for the Advancement of Structured Information Standards) (n.) ビジネス標準の開発、統合、および採用を推進するコンソーシアム。Web サイトは <http://www.oasis-open.org/> です。OASIS が後援して提供される DTD (ドキュメントタイプ定義) リポジトリは <http://www.XML.org> にあります。
- obj.conf ファイル (obj.conf file)** (n.) サーバーのオブジェクト設定ファイル。このファイルには、補足的な初期化情報、サーバーのカスタマイズ設定、およびブラウザなどのクライアントからの要求をサーバーが処理する場合に使用する命令が記述されています。Web Server は、クライアント要求を処理するたびにこのファイルを読み取ります。
- オブジェクトクラス (object class)** (n.) エントリが記述するオブジェクトの種類と、そのエントリに含まれる一連の属性を指定するテンプレート。たとえば、Directory Server では、commonname、mail (電子メールアドレス)、mailHost、mailQuota などの属性を持つ emailPerson オブジェクトクラスが指定されます。
- オブジェクト識別子 (object identifier)** 「OID (オブジェクト識別子)」を参照してください。
- オブジェクトの持続性 (object persistence) 廃止パッチ (obsolete patch)** 「持続性 (persistence)」を参照してください。
(n.) 有効または最新と見なされなくなったパッチ。パッチは、次期バージョンのパッチで同じ問題が修正されたとき、別のパッチに元のパッチから修正プログラムが取り込まれたとき、またはパッチが該当するとみなされなくなったときに、廃止となります。
- オフライン状態 (offline state)** (n.) メールクライアントがサーバーシステムからクライアントシステムにメッセージをダウンロードして、メッセージの表示や返信の作成ができる状態。サーバー上のメッセージは、削除される場合と削除されない場合があります。

OID (オブジェクト識別子)	(OID (object identifier)) (n.) ピリオドで区切られた 10 進数のリストで構成される、オブジェクト識別子の文字列表現。たとえば、1.3.6.1.4.1 のようになります。「 ライトウェイトディレクトリアクセスプロトコル (Lightweight Directory Access Protocol) 」では、オブジェクト識別子を使用して、オブジェクトクラスや属性タイプなどのスキーマ要素を一意に識別します。オブジェクト識別子階層の最上位レベルは、標準団体によって管理され、独自のスキーマ定義を構築することを望む組織に委譲されます。
OMG	(Object Management Group) (n.) 相互運用可能なエンタープライズアプリケーションのためにコンピュータ産業の仕様を作成および維持管理するコンソーシアム。Web サイトは http://www.omg.org/ です。
1 方向性 メッセージング	(n.) 応答を受信するまでブロックする必要のない「 メッセージ (message) 」の伝送方法。
オンライン状態 (online state)	(n.) メッセージをサーバー上に残したまま、メールクライアントがリモートから返信する状態。
オペレーショナル属性 (operational attribute)	(n.) オペレーショナル属性には、変更とサブツリープロパティを追跡するために、ディレクトリ内部で使用される情報が記述されています。オペレーショナル属性は、明示的に要求しなければ、検索に対する応答として返されません。
オプション属性 リスト (optional attribute list)	(n.) 特定のオブジェクトクラスのオプション属性のリスト。オプション属性の先頭には、キーワード MAY が置かれます。
ORB (オブジェクトリクエストブローカ)	(ORB (Object request broker)) (n.) CORBA (共通オブジェクトリクエストブローカーアーキテクチャー) オブジェクトが相互に確認して通信できるようにするライブラリ。
組織 (organization)	(n.) Directory Server Access Management Edition において、企業が部門とリソースを管理するために使用する、階層構造の最上位レベルを表すオブジェクト。Directory Server Access Management Edition は、インストール時に最上位レベルの組織 (default o=isp) を動的に作成し、Directory Server Access Management Edition エンタープライズの構成を管理します。インストール後に組織を追加し、別々のエンタープライズを管理することもできます。作成される組織は、すべて最上位レベル組織の下位になります。「 サブ組織 (suborganization) 」も参照してください。
組織管理者 (organization administrator)	(n.) Delegated Administrator for Messaging and Collaboration の GUI または CLI を使用して、組織またはサブ組織内のメールユーザーとメールリストの作成、変更、および削除を行う管理権限を持つユーザー。
O/R マッピング ツール (object-to-relational database tool)	(O/R mapping tool (object-to-relational database tool)) (n.) エンティティ Bean の XML 配備記述子を作成する、Application Server Administrative インタフェース内部のマッピングツール。

OSI ツリー (開放型システム間相互接続ツリー)	(OSI tree (Open System Interconnecttree)) (n.) 開放型システム間相互接続ネットワーク構文を反映する「 DIT 」。OSI ツリー内の識別名(「 DN 」)は、cn=billt,o=bridge,c=us のようになります。
OS プリンシパル (OS principal)	(n.)J2EE プラットフォームが実行中のオペレーティングシステムにネイティブなプリンシパル。
OTS	(Object Transaction Service) (n.) CORBA (共通オブジェクトリクエストブローカーアーキテクチャー) オブジェクトがトランザクションに参加できるようにするインタフェースの定義。

P

パッケージ (package)	(n.) ファイルとディレクトリのコレクション。パッケージングは、インストール用のソフトウェアを配布する方法です。「アセンブリ (assembly)」、「配備 (deployment)」も参照してください。
パラメータ (parameter)	(1) (n.) フォームフィールドデータや HTTP ヘッダー情報など、Java Enterprise System Application Server クライアントから送信される名前と値のペア。要求オブジェクト内にカプセル化されています。「属性 (attribute)」、「プロパティー (property)」も参照してください。 (2) (n.) Java メソッドまたはデータベースコンパイル済みコマンドに渡される引数。
パラメータエンティティ (parameter entity)	(n.) 一般のエンティティとは区別され、「DTD (ドキュメントタイプ定義)」仕様で構成されるエンティティ。DTD で定義されるパラメータエンティティは他のポイントでも参照することができ、そうすることにより定義を使用する場所でそのつど定義を再コードせずに済みます。
親 (parent)	(n.) XML ファイルで、子として参照される別の要素を内包する要素。「子 (child)」も参照してください。
親アクセス (parent access)	(n.) 「バインド DN (バインド識別名)」がターゲットエントリの親である場合に、権限を付与されると、ユーザーがディレクトリツリー内の自分の位置より下のエントリに対するアクセス権を持つことを示します。
解析対象エンティティ (parsed entity)	(n.) XML (拡張可能マークアップ言語) を含む一般のエンティティで、解析対象外エンティティとは対照的に XML ドキュメントに挿入される時点で解析されます。
パーサー (parser)	(n.) 入力ソースから XML (拡張可能マークアップ言語) データを読み込んでいくつかのまとまりに分割し、プログラムがタグ、属性、または要素データをいつ処理するかを識別できるようにするモジュール。非検証パーサを使用することにより、XML データが必ず整形式になるようにできますが、XML データが有効であることは確認しません。「検証パーサー (validating parser)」も参照してください。

パーティション (partition)	「メッセージストアパーティション (message store partition)」を参照してください。
パッシベーション (passivation)	(n.) 「エンタープライズ Bean (enterprise bean)」をメモリーから2次記憶域に転送する処理。Beanを破棄せずにメモリーからBeanのリソースを解放する手法。この手法により、Beanは持続し、インスタンス化のオーバーヘッドなしに再び呼び出すことができます。「アクティベーション (activation)」も参照してください。
パススルー認証 (pass-through authentication)	「PTA (パススルー認証)」を参照してください。
パススルーサブツリー (pass-through subtree) パスワード認証 (password authentication)	(n.) パススルー認証では、PTA Directory Serverが、「DN」がこのサブツリーにあるすべてのクライアントからのバインド要求を認証 Directory Serverに渡します。 (n.) ユーザー名とパスワードによるユーザーの識別。「証明書に基づく認証 (certificate-based authentication)」も参照してください。
パスワードファイル (password file)	(n.) UNIXのみ。UNIXユーザーのログイン名、パスワード、およびユーザーID番号を保存するファイル。パスワードファイルは、ファイルの格納されている位置に基づいて、/etc/passwdとも呼ばれます。
パスワードポリシー	(n.) 所定のディレクトリでのパスワードの使用法を規定する一連の規則。
パッチバージョン番号 (patch version number)	(n.) パッチ識別子の最後の2桁の数字。たとえば、「nnnnnn-03」のようになります。番号は、新規バージョンのパッチがリリースされるたびに1つ大きくなります。
パターン (pattern)	(n.) 許可フィルタや拒否フィルタなどで、マッチングのために使用される文字列表現。
PCDATA	(n.) 解析対象文字データのための定義済みXML (拡張可能マークアップ言語) タグ。「以下の文字を解釈しない」ことを意味する文字データ (CDATA) とは対照的に、XML構文の通常ルールが適用されます。「CDATA」も参照してください。
PDC (個人デジタル証明書)	(PDC (personal digital certificate)) (n.) ユーザーを認証する、メッセージに添付された電子的証明書。個人デジタル証明書は、ユーザーIDとパスワードを正しく入力することによって作成することも、ユーザーが接続しているサーバーのセキュリティー証明書を代わりに使用する、「SSL 証明書 (SSL certificate)」要求を使用して作成することもできます。
ピア (peer)	(n.) 別のサブカテゴリと同じ親カテゴリを持つサブカテゴリ。

永続的な失敗 (permanent failure)	(n.) メッセージ処理中に発生するエラー状態。永続的な失敗が発生すると、メッセージストアは電子メールメッセージのコピーを削除します。「 MTA (メッセージ転送エージェント) 」はメッセージを差出人に戻し、メッセージのコピーを削除します。
アクセス権 (permissions)	<p>(1) (n.) ユーザーまたはグループに対して付与または拒否する一連の権限。この情報には、ユーザーやグループの名前、1つまたは複数の有効な電子メールアドレス、および電子メールの配信方法と配信先が含まれます。</p> <p>(2) (n.) アクセス制御において、ディレクトリ情報へのアクセスが付与または拒否されていることと、付与または拒否されているアクセスレベルを示すアクセス権の状態。「アクセス権限 (access right)」も参照してください。</p> <p>(3) (n.) カレンダーに対するアクセスを制御する設定。たとえば、Calendar Express におけるアクセス権は、空き時間の表示、出席依頼、読み込み、削除、修正などです。Calendar Server の管理者は、コマンド行ユーティリティーを使って、「ACE (アクセス制御エントリ)」の文字列としてアクセス権を設定します。「ACL (アクセス制御リスト)」も参照してください。</p>
持続性 (persistence)	<p>(1) (n.) コンポーネントの場合、インスタンス変数と配下のデータベースとの間で状態を転送するためのプロトコル。「エンティティ Bean (entity bean)」を参照してください。「トランジエンス (transience)」も参照してください。</p> <p>(2) (n.) セッションでは、セッションのストレージメカニズムを意味します。「セッション」、「フェイルオーバー (failover)」、「セッションフェイルオーバー (session failover)」も参照してください。</p>
持続マネージャー (persistence manager)	(n.) EJB 1.x または 2.x 「 エンティティ Bean (entity bean) 」の「 持続性 (persistence) 」を管理するマネージャー。
持続フィールド (persistent field)	(n.) 「 コンテナ管理による持続性 (container-managed persistence) 」のある EJB 2.1 「 エンティティ Bean (entity bean) 」の仮想フィールド。データベース内に格納されません。
持続状態 (persistent state)	(n.) オブジェクトの状態が持続的記憶領域 (通常はデータベース) に保存されている状態。
個人デジタル証明書 (personal digital certificate)	「 PDC (個人デジタル証明書) 」を参照してください。
個人用フォルダ (personal folder)	(n.) 所有者だけが読み取り可能なフォルダ。「 共有フォルダ (shared folder) 」も参照してください。
pk12util	(n.) 内部マシンから証明書と鍵データベースをエクスポートし、外部 PKCS#11 モジュールにそれらの情報をインポートする場合に必要なソフトウェアユーティリティー。

PKI (公開鍵基盤)	(PKI (public key infrastructure)) (n.) ユーザーのアイデンティティーをブラウザまたはモバイルデバイスにリンクできるようにします。無線 PKI は、ハンドセットで行われる「 証明書に基づく認証 (certificate-based authentication) 」を指します。
プレーンテキスト (plaintext)	(n.) データの送信方法。定義は状況によって異なります。「 Secure Socket Layer 」の場合、プレーンテキストパスワードは暗号化され、平文としては送信されません。「 SASL 」では、プレーンテキストパスワードがハッシュされ、パスワードのハッシュだけがテキストとして送信されます。
プレーンテキスト認証 (plaintext authentication)	「 パスワード認証 (password authentication) 」を参照してください。
プラグイン可能な認証 (pluggable authentication)	(n.) J2EE アプリケーションが J2SE™ プラットフォームから Java™ Authentication and Authorization Service (JAAS) ソフトウェアを利用できるようにするメカニズム。開発者は、独自の認証メカニズムをプラグインできます。
プラグイン (plug-in)	(1) (n.) Web ページ内部のコンテンツを表示または実行するブラウザに対するコード拡張機能。プラグインにより、ブラウザは、プラグインがなければ表示できないページのコンテンツ要素を表示できます。 (2) (n.) 読み込んでからシステムの一部として使用できるアクセサリプログラム。たとえば、Calendar Server は、プラグインを使用することによって LDAP 以外のディレクトリサービスにアクセスできます。
POA (ポータブルオブジェクトアダプタ)	(POA (Portable Object Adapter)) (n.) 異機種システム混在「 ORB (オブジェクトリクエストブローカ) 」間で移植可能なサーバーサイドアプリケーションをビルドするための CORBA (共通オブジェクトリクエストブローカーアーキテクチャー) 標準。
ポインタ CoS (pointer CoS)	(n.) テンプレート DN のみを使用してテンプレートエントリを識別するサービスのポインタクラス。
ポイントツーポイント配信モデル (point-to-point delivery model)	(n.) 「 メッセージプロデューサ (message producer) 」が「 メッセージ (message) 」を特定の「 メッセージキュー (message queue) 」に送り、メッセージを保持するために確立されたキューから「 メッセージコンシューマ (message consumer) 」がメッセージを抽出するモデル。メッセージは、1つのメッセージコンシューマにだけ配信されます。
ポリシー (policy)	(1) (n.) 特定の条件下で特定のリソースにアクセスする権限を与えられるユーザーについて規定する規則。この規則は、組織内のユーザーグループまたはロールに基づいて規定することができます。 (2) (n.) Directory Server Access Management Edition では、規則を定義して組織の Web リソースを保護します。ポリシーは、組織またはロールにのみ割り当てられます。

ポーリング (poll)	(n.) Instant Messaging Server の機能。ユーザーに質問に回答するよう依頼することができます。選択したユーザーに質問と考えられる回答を送信すれば、ユーザーは答えを選択して回答することができます。
プール (pooling)	(n.) 多くの設定済みリソースを提供してパフォーマンスを向上させるプロセス。リソースがプールされていれば、新しくリソースをインスタンス化しなくても、コンポーネントはプールにある既存のインスタンスを使用することができます。Java Enterprise System Application Server では、データベース接続、サーブレットインスタンス、およびエンタープライズ Bean インスタンスをすべてプールできます。
POP3 (ポストオフィスプロトコルバージョン3)	(POP3 (Post Office Protocol Version 3)) (n.) 標準の配信方法を提供するプロトコル。このプロトコルを使用する場合、「MTA (メッセージ転送エージェント)」はユーザーのメールフォルダへのアクセス権を持っている必要はありません。アクセス権が不要なことは、メールクライアントとMTAが別のコンピュータに置かれることが多いネットワーク環境において利点となります。
ポート (port)	(n.) TCP/IP 接続の確立先となる位置すなわちソケット。従来、Web サーバーはポート 80 を、FTP はポート 21 を、telnet はポート 23 を使用しています。Java Enterprise System Portal Server では、特にクライアントシステムの場合に特別なポートを使用して、Portal Server セッションを経由し、サーバーとの間で安全な通信を行います。
ポータル (portal)	(n.) エンタープライズが、ポータルのユーザーに提供する、一連のリソースへのエントリーポイント。コンシューマポータルの中には、一連のリソースとして World Wide Web 全体を含めているものもありますが、ほとんどのエンタープライズにとってのリソースとは、ユーザーとエンタープライズの関係に固有の情報、アプリケーション、およびその他のリソースです。Portal Server デスクトップは、Portal Server でポータルを生成するために使用するアプリケーションです。
ポータルデスクトップ (Portal Desktop)	(n.) Portal Server によって生成されたデスクトップすべて。
Portal Server	(n.) リモートユーザーが、組織のネットワークおよびネットワークのサービスに、インターネットを介してセキュリティ保護された方法でアクセスするのを可能にするソフトウェア製品。セキュリティ保護されたインターネットポータルを作成し、社員、ビジネスパートナー、または不特定多数のユーザーなど、すべての対象閲覧者が、コンテンツ、アプリケーション、およびデータにアクセスできるようにします。Sun Java System 製品ソリューション全体の中核ソフトウェアであり、すべての Portal Server パッケージ製品で共有されます。
Portal Server Desktop	(n.) 主要なエンドユーザーインタフェースと、コンテンツプロバイダインタフェース (PAPI) を使用した拡張可能コンテンツを集約するメカニズムを提供します。多くの場合「Desktop」と呼ばれます。Desktop には、コンテンツ階層や、特定タイプのチャンネルを作成する基本的な設計要素を提供する、さまざまなプロバイダが組み込まれています。Desktop は、Directory Server Access Management Edition サービスの最

上位に表示プロファイルデータメカニズムを実装して、コンテンツプロバイダとチャンネルデータを保管します。Desktop には、表示プロファイルとほかの Desktop サービスデータの編集で使用する管理コンソールモジュールも組み込まれています。

Portal Server Instant Collaboration Pack	(n.) サーバー、「マルチプレクサ (multiplexor)」、および Instant Messaging コンポーネントを組み込んだサーバーインスタントメッセージング製品。Instant Messaging Server とも呼ばれます。
Portal Server Pack	(n.) Portal Server のアドオン製品を指す一般的な用語。
ポータルノード (portal node)	(n.) Portal Server ソフトウェアまたは Portal Server Pack ソフトウェアが動作している物理マシン。「ホスト (host)」とも呼ばれます。
ポート番号 (port number)	(n.) ホストマシンにある個々の Transmission Control Protocol/Internet Protocol アプリケーションを指定する番号。送信データの送信先を示します。
配備後 (post-deployment)	(n.) Java Enterprise System ソリューションのライフサイクルプロセスの1段階。ここでは、分散アプリケーションの起動、監視、チューニングによるパフォーマンスの最適化、および動的なアップグレードによる新規機能の組み込みを行います。
インストール後の設定 (postinstallation configuration)	(n.) Java Enterprise System インストーラを(たいていは「後で設定」オプションを指定して)実行した後に行う Access Manager の設定作業。通常、インストール後の設定の作業には数分かかります。たとえば、製品のインスタンスを追加で配備したり、製品に対してセッションフェイルオーバーの設定を行う場合があります。「設定 (configuration)」も参照してください。
ポストマスターアカウント (postmaster account)	(n.) Messaging Server からのシステム生成メッセージを受信する電子メールグループおよび電子メールアドレスのエイリアス。ポストマスターアカウントには、1つ以上の有効なメールボックスを指定する必要があります。
ポストオフィスプロトコルバージョン3 (Post Office Protocol Version 3)	「POP3 (ポストオフィスプロトコルバージョン3)」を参照してください。
配備前 (pre-deployment)	(n.) Java Enterprise System ソリューションのライフサイクルプロセスの1段階。ここでは、ビジネスニーズが、「配備シナリオ (deployment scenario)」すなわち「論理アーキテクチャー (logical architecture)」に変換されます。
優先ディレクトリサーバー (preferred directory server)	(n.) Identity Synchronization for Windows がユーザーエントリを検出して変更を適用するために使用するディレクトリサーバーマスターインスタンス。このサーバーが使用可能である間、Identity Synchronization for Windows は他のディレクトリサーバーマスターとは通信しません。

準備済みコマンド (prepared command)	(n.) 実行の繰り返しを効率よくするために、「 SQL (構造化クエリー言語) 」で書かれた、あらかじめコンパイルされているデータベースコマンド。準備済みコマンドにはパラメータを入れることができます。「 準備済み文 (prepared statement) 」も参照してください。
準備済み文 (prepared statement)	(n.) データをフェッチするために繰り返し使用される、QUERY、UPDATE、または INSERT 文をカプセル化するクラス。準備済み文には、少なくとも1つの「 準備済みコマンド (prepared command) 」が記述されています。
実在インデックス (presence index)	(n.) エントリ内の属性値に関係なく、指定したタイプの属性を持つエントリを効果的に検索できるようにするフィルタ手法。
プレゼンテーションレイアウト (presentation layout)	(n.) Web ページコンテンツの形式。
プレゼンテーションロジック (presentation logic)	(n.) アプリケーションでページを作成するアクティビティー。要求の処理、応答時のコンテンツの生成、クライアントに返すページのフォーマットなど。通常は、Web アプリケーションによって処理されます。
事前設定メッセージ (preset message)	(n.) Portal Server Mobile Access モバイル設定として書き込みおよび保存し、あとでモバイルメールアプリケーションで使用することができるショートメッセージ。
主データビュー (primary data view)	(n.) 結合データビューを構成する2つの Directory Proxy Server データビューの一方。デフォルトでは、主データビューがエントリの正式なソースです。「 二次データビュー (secondary data view) 」も参照してください。
主ドキュメントディレクトリ (primary document directory)	「 ドキュメントルート (document root) 」を参照してください。
主キー (primary key)	(n.) クライアントがホーム内で特定の EJB 2.1 「 エンティティー Bean (entity bean) 」を検出できるようにする固有 ID。
主キークラス名 (primary key class name)	(n.) Bean の主キーの完全修飾クラス名を指定する変数。Java Naming and Directory Interface™ (JNDI) 検索で使います。
主体 (principal)	(n.) 「 認証 (authentication) 」の結果としてユーザーに割り当てられるアイデンティティー。主体は決定を行える連携アイデンティティーを獲得することができ、自ら認証済みのアクションを実行できます。たとえば主体には、個々のユーザー、個々のユーザーによるグループ、企業、ほかの正当なエンティティー、または Liberty アーキテクチャーのコンポーネントなどがあります。

非公開鍵
(private key)

「公開鍵暗号化 (public-key cryptography)」を参照してください。

特権 (privilege)

(n.) ユーザー、1組のユーザー、またはリソースに与えられるアクセス権限のタイプ。このセキュリティ属性には一意性のプロパティがないため、多数の主体で共有することができます。

プロセス
(process)

(1) (n.) オペレーティングシステムによって設定される、自己完結で完全に機能する実行環境。アプリケーションの各インスタンスは、一般に別々のプロセスで実行されます。

(2) (n.) アクティブなプログラムの実行シーケンス。プロセスは、1つまたは複数のスレッドから構成されます。

処理命令
(processing instruction)

(n.) 特定のアプリケーションによって解釈されることを意図した、「XML (拡張可能マークアップ言語)」構造に囲まれた情報。

プロデュースする
(produce)

(v.) メッセージを送信先に配信するために、クライアントランタイムにメッセージを送信すること。

プロデューサ
(producer)

(n.) 送信先にメッセージを送信するために使用するセッションによって作成されるオブジェクト (MessageProducer)。ポイントツーポイント配信モデルでのプロデューサは送信側 (QueueSender) です。パブリッシュ/サブスクライブ配信モデルでのプロデューサは、発行者 (TopicPublisher) です。

稼動環境
(production environment)

(n.) アプリケーションのライフサイクルプロセスの1段階。ここでは、分散アプリケーションの起動、監視、チューニングによるパフォーマンスの最適化、および動的なアップグレードによる新規機能の組み込みを行います。

プログラムセキュリティ
(programmatic security)

(n.) たとえば、コンポーネントのコンテナ、Beanのコンテナ、またはサーブレットエンジンなどによって処理するのではなく、コードを記述して明示的にセキュリティを制御するプロセス。「宣言によるセキュリティ (declarative security)」とは反対の概念です。プログラムセキュリティは、宣言によるセキュリティだけではアプリケーションのセキュリティモデルを表現するのに不十分であるときに役立ちます。

プログラマによる境界設定トランザクション
(programmer-demarcated transaction)

「Bean管理によるトランザクション (bean-managed transaction)」を参照してください。

prolog

(n.) XML (拡張可能マークアップ言語) ドキュメントの、XMLデータに先行する部分。prologは、宣言と省略可能なDTD (ドキュメントタイプ定義) からなります。

伝播動作 (propagation behavior)	(n.) コンシューマとサプライヤの間の同期プロセス。
プロパティ (property)	<p>(1) (n.) アプリケーションコンポーネントの動作を定義する1つの名前と値の対。「パラメータ (parameter)」も参照してください。</p> <p>(2) (n.) XML ファイルで要素を修飾する名前と値の対。ただし、DTD ファイルで事前定義はされていません。「属性 (attribute)」と対比。</p> <p>(3) (n.) Application Server で、組み込みサーバー設定の一部ではない名前と値の対。「属性 (attribute)」と対比。</p>
プロトコル (protocol)	<p>(1) (n.) ネットワークのデバイスが情報を交換する方法について記述した一連の規則。</p> <p>(2) (n.) 情報を交換する2つ以上のシステムが従う必要がある規則と、交換されるメッセージに関する公式の説明。</p>
プロバイダ (provider)	<p>(n.) チャンネルのプログラムの側面。プロバイダに設定データを追加すると、プロバイダは区別されてチャンネルのインスタンスになります。プロバイダには、Java クラスで、ファイルの内容またはアプリケーションやサービスの出力をチャンネルの正しい形式に変換する役割があります。Portal Server には、ブックマークプロバイダ、アプリケーションプロバイダ、ノートプロバイダなどの数多くのプロバイダが付属しています。デスクトップが表示されると、今度は各プロバイダに対して関連付けられたチャンネルのコンテンツに関して照会されます。プロバイダの中には、その設定に基づいて複数のチャンネルを生成できるものもあります。</p> <p>コンテンツプロバイダの例としては、UserInfoProvider や BookmarkProvider があります。「コンテナ (container)」プロバイダの例としては、TabContainerProvider や SingleContainerProvider があります。リーフプロバイダの例としては、JSPPProvider、XMLProvider、URLScrapperProvider、および SimpleWebServicesProvider があります。</p>
プロバイダ連合 (provider federation)	(n.) Liberty Alliance Project 仕様に基づいたアーキテクチャーを使用し、契約により認証情報を交換することに同意したサービスプロバイダのグループ。「 認証ドメイン (authentication domain) 」も参照してください。
プロビジョニング (provisioning)	(n.) Java Enterprise System Directory Server のエントリを追加、変更、または削除するプロセス。これらのエントリには、ユーザー、グループ、およびドメイン情報が含まれます。
プロキシ (proxy)	(1) (n.) 1つのシステムが別のシステムの代わりにプロトコルの要求に応答するメカニズム。プロキシシステムをネットワーク管理で使用すると、モデムなどの単純なデバイスに完全なプロトコルスタックを実装する必要がなくなります。

(2)(n.) クライアントの代わりに要求を作成し、要求サービスを提供する中間プログラム。プロキシは今度はサーバーまたはクライアントとして動作し、さまざまなネットワークサービスのコンテンツを制御するために使用されます。「[リバースプロキシ \(reverse proxy\)](#)」も参照してください。

プロキシ承認 (proxy authorization)

(n.) クライアントが固有のアイデンティティによってディレクトリにバインドされても、操作単位で別のユーザーのアクセス権限が与えられる、認証の特別な形態。もう一方のユーザーは、プロキシユーザーと呼ばれ、その「DN」はプロキシDNと呼ばれます。

プロキシ DN (proxy DN)

(n.) クライアントアプリケーションが操作の実行を試みているターゲットへのアクセス権を持つエントリの「DN」。「[プロキシ承認 \(proxy authorization\)](#)」で使用されます。

Proxylet

(n.) クライアントマシンで動作して、URL を SRA Gateway にリダイレクトする動的プロキシサーバー。「[Secure Remote Access \(SRA\)](#)」も参照してください。

PTA (パススルー認証)

(PTA (pass-through authentication)) (n.) 1 つの Java EnterpriseSystem Directory Server が別の Directory Server に問い合わせるパインドルールを確認するためのメカニズム。

PTA Directory Server

(n.) 「[パススルー認証 \(pass-through authentication\)](#)」では、PTA Directory Server が、受信するパインド要求を認証側の Directory Server に送信、つまりパススルーします。

PTA LDAP URL

(n.) 「[パススルー認証 \(pass-through authentication\)](#)」において、認証側の Directory Server、1 つまたは複数のパススルーサブツリー、およびオプションのパラメータを定義する URL。

公開フォルダ (public folder)

(n.) アクセス可能な複数のユーザーによって共有される、所有者が複数のフォルダ。フォルダに設定されている「[ACL \(アクセス制御リスト\)](#)」によっては、複数のユーザーがそのフォルダを更新したり管理したりできます。

公開情報ディレクトリ (public information directories)

(n.) UNIX のみ。UNIX ユーザーのホームディレクトリまたはユーザーの制御下にあるドキュメントルート内にはないディレクトリ、またはユーザーの制御下にあるディレクトリ。

公開鍵 (public key)

(n.) 公開鍵暗号方式で使用される暗号化鍵。

公開鍵証明書 (public key certificate)

(n.) ユーザーの公開鍵、および証明書が有効な時間と日付に関する情報を収めたデータ構造。クライアント証明書認証で、サーバーが(場合によってはクライアントも)相互認証できるようにするために使用されます。公開鍵証明書はデジタル版のパスポートのようなものです。証明書発行局と呼ばれる信頼できる組織によって発行され、保持者の識別情報を提供します。

公開鍵暗号化
(**public-key
cryptography**)

暗号化の手法。公開鍵暗号システムでは、すべてのユーザーが、公開鍵と秘密鍵(非公開鍵とも呼ばれる)という関連のある2つの補完鍵を持っています。それぞれの鍵は、他方の鍵で作成されたコードのロックを解除します。公開鍵を知っていても、対応する秘密鍵を推論することはできません。公開鍵は、通信ネットワーク全体に公開し、広く配布することができます。このプロトコルにより、従来の暗号化システムで必要とされるセキュリティー保護されたチャンネルがなくても、プライバシーを守ることができます。非対称鍵暗号方式とも呼ばれます。

公開鍵暗号化
(**public key
encryption**)

(n.) 公開コンポーネントと非公開コンポーネントの2つの部分から成る鍵(コード)を使用する暗号化方式。メッセージの暗号化には、受取人の公開鍵が使われます。メッセージを解読する場合は、受取人が、自分だけが知っている非公開の鍵を使用します。

公開鍵インフラ
ストラクチャー
(**public key
infrastructure**)

「PKI(公開鍵基盤)」を参照してください。

**Public Network
Management**

(n.) 障害モニターとフェイルオーバーを使用し、単一のネットワークアダプタやケーブルの障害が原因でノードの可用性が失われるのを防ぐソフトウェア。Public Network Management のフェイルオーバー機能では、ネットワークアダプタフェイルオーバーグループと呼ばれる一連のネットワークアダプタを使用して、クラスタノードとパブリックネットワークの間で冗長接続を確立します。障害モニターとフェイルオーバー機能は共に動作し、リソースの可用性を確保する。

パブリッシュ/サ
ブスクライブ配
信モデル
(**publish/subscribe
delivery model**)

(n.) 一般にはパブリッシャーとサブスクライバが匿名で、コンテンツ階層の特定のノード(「トピック(topic)」と呼ばれる)に対して動的にパブリッシュまたはサブスクライブできるメッセージ送信システム。システムは、トピックの複数のパブリッシャーから到着する「メッセージ(message)」を複数のサブスクライバに配信します。

メッセージを
パージする
(**purge message**)

(動詞) ユーザーおよびグループフォルダ内で削除され、参照することのなくなったメッセージを永久に削除すること。その領域は、メッセージストアのファイルシステムに返されます。「メッセージを削除する(delete message)」、「メッセージを破棄する(expunge message)」も参照してください。

Q

- QoS (サービス品質)** (QoS (Quality of Service)) (n.) サーバーインスタンスまたは仮想サーバーに対して設定するパフォーマンスの制限。たとえば、ISP は、提供する帯域幅に応じて仮想サーバーの課金額を変えることがあります。この場合は、帯域幅の量と接続数を制限することができます。
- クエリー文字列 (query string)** (n.) 要求の処理に影響を与える一連のパラメータや値を含む HTTP 要求 URL のコンポーネント。
- キュー (queue)** (n.) Java Message Service で、ポイントツーポイント配信モデルを実装するために管理者によって作成されるオブジェクト。メッセージをコンシュームするクライアントがアクティブでない場合でも、メッセージを保持するためにキューは常に使用可能です。キューは、プロデューサとコンシューマの中間段階の待機場所として使用されます。「[JMS](#)」、「[ポイントツーポイント配信モデル \(point-to-point delivery model\)](#)」を参照してください。

R

RAF (ロボットアプリケーション関数)	(RAF (robot application function)) (n.) ロボットのフィルタ設定ファイルで使用可能な関数。ユーザー定義のロボットアプリケーション関数は、プラグイン関数とも呼ばれます。これらの関数は、指令によって呼び出されます。
RAM (ランダムアクセスメモリー)	(RAM (random access memory)) (n.) コンピュータ内の物理的な半導体ベースメモリー。
RAR ファイル(リソースアーカイブファイル)	(PAR file (resource adapter archive)) (n.) リソースアダプタを内蔵した Java™ アーカイブ(JAR) ファイルで、「コネクタ (connector)」モジュールとも呼ばれる。
RC2	(n.) RSA Data Security によって提供される可変鍵サイズによるブロック暗号化方式。
rc.2.d ファイル (rc.2.d file)	(n.) UNIX のみ。マシンの起動時に実行されるプログラムについて記述した、UNIX マシンのファイル。このファイルは、格納されている位置に基づいて、/etc/rc.2.d と呼ばれます。
RC4	(n.) RSA Data Security によって提供されるストリーム暗号化方式。RC2 よりも処理が高速です。
RD	「リソース記述 (resource description)」を参照してください。
RDB	(n.) リレーショナルデータベース。
RDBMS	(n.) リレーショナルデータベース管理システム。
RDF (リソース記述フレームワーク)	(RDF (Resource Description Framework)) (n.) 「XML (拡張可能マークアップ言語)」ファイルに含まれるデータの種類を定義する標準。この情報は、日付が単なるテキストではなく日付として必ず扱われるようにするなど、意味上の整合性を確保するのに役立ちます。

RDF スキーマ (RDF schema)	(n.) RDF に含まれる種々の仕様に適用される整合性規則を指定する際の標準。
RDM	「リソース記述メッセージ (resource description message)」を参照してください。
RDN 相対識別名	(RDN (relative distinguished name)) (n.) 実際のエントリ自体の名前。この文字列にエントリの祖先を付加すると、完全な「DN」になりますほとんどの RDN は、エントリからの 1 つの属性タイプと値で構成されています。
読み取り専用 Bean (read-only bean)	(n.) EJB™ クライアントによって修正されることがないエンティティ Bean。「エンティティ Bean (entity bean)」も参照してください。
レルム (realm)	(n.) 共通セキュリティポリシーが定義され、セキュリティサービスのセキュリティ管理者によって適用されている領域。セキュリティポリシードメインまたはセキュリティドメインとも呼ばれます。J2EE サーバー認証サービスで、レルムは、ロール、ユーザー (または主体)、および Web アプリケーション (複数可) の有効なユーザーを識別するグループの完全なデータベースです。
リダイレクション (redirection)	(n.) 特定の URL にアクセスしているクライアントを、同じサーバーまたは別のサーバーにある異なる位置に送信する際のメカニズム。リダイレクションは、リソースが移動されているときに、クライアントに新しい位置を透過的に使用させたい場合に便利です。またリダイレクションを使用して、後続のスラッシュを付けずにディレクトリへのアクセスがあった場合に、相対リンクの整合性を保持します。
再入可能エンティティ Bean (reentrant entity bean)	(n.) 相互に干渉し合うことのない同時呼び出し、インターリーブ呼び出し、または入れ子の呼び出しを複数処理することが可能な「エンティティ Bean (entity bean)」。
参照 (reference)	(n.) XML ドキュメントが解析されるときに、参照から置き換わるエンティティへの参照。「エンティティ参照 (entity reference)」を参照してください。
リファレンス配備アーキテクチャー (reference deployment architecture)	(n.) パフォーマンスの基準として設計、実装、およびテスト済みの「配備アーキテクチャー (deployment architecture)」。リファレンス配備アーキテクチャーは、カスタムソリューションの配備アーキテクチャーを設計する場合の開始点として使用されます。
参照整合性 (referential integrity)	(n.) 「DN」値の属性によって表現されるエントリ間の関係をディレクトリ内で確実に維持するためのメカニズム。
リフェラル (referral)	(n.) サーバーは、処理不能な検索要求または更新要求をクライアントから受け取ると、その要求を処理できる Java Enterprise System Directory Server へのポインタをクライアントに返します。
リフェラルホップ制限 (referral hop limit)	(n.) クライアントが連続して追跡するリフェラルの最大数。

レジストリ (registry)	(n.) 「 Web サービス (web service) 」の構築、配備、および検索を可能にするインフラストラクチャー。これは中立的なサードパーティーであり、疎結合された動的な B2B 対話を支援します。
レジストリプロバイダ (registry provider)	(n.) ebXML や UDDI などの XML レジストリの仕様に準拠したビジネスレジストリの実装。
正規表現 (regular expression)	(n.) パターンマッチングのために、文字の範囲またはクラスを表す特殊文字を使用した文字列。
相対識別名 (relative distinguished name)	「 RDN 相対識別名 」を参照してください。
関係フィールド (relationship field)	(n.) 「 コンテナ管理による持続性 (container-managed persistence) 」のある「 エンティティ Bean (entity bean) 」の仮想フィールド。関係するエンティティ Bean を識別します。
リレー (relaying)	(n.) メッセージサーバー間でメッセージを渡すプロセス。
リモートインタフェース (remote interface)	(n.) EJB 1.x および 2.x コンポーネント用の 2 つのインタフェースのうちの 1 つ。リモートインタフェースでは、クライアントから呼び出すビジネスメソッドを定義します。「 ホームインタフェース (home interface) 」も参照してください。
remove メソッド (remove method)	(n.) 「 ホームインタフェース (home interface) 」に定義され、EJB 1.x または 2.x エンタープライズ Bean を破棄するときにクライアントによって呼び出されるメソッド。
レンダラ (renderer)	(n.) 一連の JavaServer Faces UI コンポーネント (JavaServer Faces UI component) 」の出力をレンダリングする Java クラス。
レンダリング (rendering)	(1) (n.) 抽象マークアップ言語 (Abstract Markup Language、AML) で記述されたコンテンツを、特定のモバイルデバイス用の適切なデバイス固有マークアップ言語に変換するプロセス。 (2) (n.) クライアントに対して出力を生成する処理。「 レンダラ (renderer) 」を参照してください。
レンダリングチャンネル (rendering channel)	(n.) レンダリングコンテンツを表示する Portal Server Mobile Access のチャンネル。
レンダリングエンジン (rendering engine)	(n.) Portal Server において、AML を所定のモバイルクライアントに適切な言語に変換するエンジン。
レンダリングフィルタ (rendering filter)	(n.) レンダリングエンジンとクライアントの間でコンテンツを渡して変換する場合に使用するフィルタ。

レンダリング キット (render kit)	(n.) 特定のクライアントに対して出力をレンダリングする一群のレンダラ。 「 JavaServer Faces テクノロジ (JavaServer Faces technology) 」実装には、HTML マークアップをレンダリングできる複数のレンダラからなる標準の HTML レンダリングキットが備わっています。
レプリカ (replica)	(n.) レプリケーションアグリーメントによって、1つ以上の他のサフィックスにリンクされているディレクトリサーバーのサフィックス。
レプリカサイクル (replica cycle)	「 レプリケーションサイクル (replication cycle) 」を参照してください。
レプリカディレクトリサーバー (replica directory server)	(n.) データのすべてまたは一部のコピーを受け取るディレクトリ。
レプリカグループ (replica group)	(n.) レプリケーションの特定領域のインスタンスを保持するサーバー。サーバーは、複数のレプリカグループに属することができます。
レプリケーション (replication)	(n.) Directory Server にわたって配信されたデータを同期し、更新の競合を修正するプロセス。
レプリケーションアグリーメント (replication agreement)	(n.) サプライヤサーバーに保存され、レプリケートするサフィックス、データをプッシュする先のコンシューマサーバー、「 レプリケーション (replication) 」が行われる時間、コンシューマをバインドするためにサプライヤが使用する「 DN 」と資格、および接続をセキュリティー保護する方法を識別する一連の設定パラメータ。
レプリケーション基本エントリ (replication base entry)	(n.) レプリケートされた領域のルートの「 DN 」。
レプリケーションサイクル (replication cycle)	(n.) 複数のレプリカ間で更新情報が交換される間隔。レプリケーションサイクルは、別のレプリカまたはレプリカのセットとの間でデータをプッシュまたはプルしようとするときに開始し、データの交換が正常に終了したときか、エラーが発生したときに終了します。
レプリケーションセッション (replication session)	(n.) 更新情報を「 レプリケーションサイクル (replication cycle) 」の一部として渡すために、「 レプリカグループ (replica group) 」内の2台のサーバー間に開始されるセッション。
要求オブジェクト (request object)	(n.) クライアントによって生成され、「 JavaServer Pages テクノロジ (JavaServer Pages technology) 」によって作成されたサーブレットまたはページに入力パラメータとして渡されるページデータおよびセッションデータを持つオブジェクト。
要求応答 メッセージング (request-response messaging)	(n.) 「 メッセージング (messaging) 」の一方式で、応答が受信されるまでブロックされます。

必須の属性リスト (required attribute list)	(n.) 特定のオブジェクトクラスで必須の属性のリスト。必須の属性の先頭には、キーワード MUST が置かれます。
必須の属性 (required attributes)	(n.) 特定のオブジェクトクラスを使用するエントリ内に存在する必要がある属性。「許可された属性 (allowed attributes)」、「属性 (attribute)」も参照してください。
リソース (resource)	(1)(n.) Web ページ、ドキュメント、FTP ディレクトリなど、URL によって識別可能なネットワーク上の任意のアイテム。リソースは、非公式にドキュメントと呼ばれることが多くあります。 (2)(n.) クライアントが要求するとサーバーがアクセスしてクライアントに送信できる、URL、ディレクトリ、またはプログラム。 (3)(n.) データベースサーバーやメッセージングシステムなどのシステムへの接続を提供するプログラムオブジェクト。
リソースアダプタ (resource adapter)	(n.) エンタープライズ情報システム (EIS (エンタープライズ情報システム)) に接続するために EJB コンテナまたはアプリケーションクライアントによって使用される、システムレベルのソフトウェアドライバ。通常、リソースアダプタは EIS 専用です。リソースアダプタはライブラリとして使用可能で、サーバーまたはクライアントのアドレス空間内で使用されます。リソースアダプタはコンテナにプラグインします。コンテナ上に配備されたアプリケーションコンポーネントは、アダプタによって公開されたクライアント API またはツール生成の高レベル抽象化オブジェクトを使用して、配下の EIS にアクセスします。リソースアダプタと EJB コンテナは連携して、EIS への接続に、トランザクション、セキュリティ、接続プーリングといった基礎となるメカニズムを提供します。「コネクタ (connector)」も参照してください。
リソースアダプタモジュール (resource adapter module)	(n.) すべての Java インタフェース、クラス、およびネイティブなライブラリを含む配備可能な単位。リソースアダプタ配備記述子とともにリソースアダプタを実装します。
リソースカレンダー (resource calendar)	(n.) 会議室などのリソース、またはノート型コンピュータや OHP などの機器に関連付けられたカレンダー。
リソース記述 (resource description)	(n.) URL によってリソースに関連付けられた属性値ペアのリスト。リソース記述は、エージェントが自動的に生成したり、ユーザーが手動で記述したりすることができます。リソース記述のリポジトリがアセンブルされると、サーバーはリソース記述メッセージを使用して、Web エージェントがリソース記述を検知して取得する、プログラムによる方法でリポジトリをエクスポートできます。リソース記述は「 SOIF 」形式で保存されます。

リソース記述 メッセージ (resource description message)	(n.) リソース記述として知られる、ネットワーク経由でアクセス可能なリソースについてのメタデータを検知および取得するためのメカニズム。
リソース呼び出し (resource invocation)	(n.) ノードで実行されるリソースタイプのインスタンス。ノードで起動されたリソースを表す抽象的な概念。
リソースマネージャー (resource manager)	(n.) 共有リソースのセットへのアクセスを提供します。リソースマネージャーは、トランザクションマネージャーによって外部から制御および同期されるトランザクションに参加します。通常、リソースマネージャーはアクセスを受けるクライアントとは異なるアドレス空間またはマシンにあります。注意: エンタープライズ情報システム (EIS (エンタープライズ情報システム)) は、リソースやトランザクション管理の文脈ではリソースマネージャーと呼ばれます。
リソースマネージャー接続 (resource manager connection)	(n.) リソースマネージャーとの「セッション」を表すオブジェクト。
リソースマネージャー接続 ファクトリ (resource manager connection factory)	(n.) リソースマネージャー接続の作成に使用されるオブジェクト。
リソースオファリング (resource offering)	(n.) ディスカバリサービスでは、リソースオファリングにより、アイデンティティデータの一部と、リソースオファリングにアクセスできるようにするサービスインスタンスの間の関係を定義します。
リソース参照 (resource reference)	(n.) リソースのコード化されたコンポーネント名を識別する配備記述子の要素。
応答バッファ (response buffer)	(n.) Portal Server Mobile Access サーバーの応答バッファでは、サイズの大きな応答を別個の小さな応答として保存し、制限のあるデバイスバッファに適合させます。
応答オブジェクト (response object)	(n.) 呼び出し側クライアントを参照して、そのクライアントへの出力を生成するメソッドを提供するオブジェクト。
再起動する (restart)	(v.) 状態情報を削除せずに「ロボット (robot)」を起動すること。これによりロボットは、最後に停止したときと同じ状態で動作を開始することができます。「フレッシュスタート (fresh start)」とは反対の概念です。
復元する (restore)	(v.) フォルダの内容をバックアップデバイスからメッセージストアにコピーすること。「バックアップする (back up)」も参照してください。

ResultSet オブジェクト (ResultSet object)	(n.) <code>java.sql.ResultSet</code> インタフェースを実装するオブジェクト。 ResultSet オブジェクトは、データベースまたはほかのソースの表形式データから取得した一連の行のカプセル化に使用されます。
再同期間隔 (resync interval)	(n.) Identity Synchronization for Windows ディレクトリソースに変更がないかをコネクタが確認する頻度。この定期確認は効率的で、最後の確認以降に変更されているユーザーのエントリのみを読み取ります。コンソールはこの値をミリ秒で表します。デフォルトは 1000 (1 秒) です。
旧バージョン形式の更新履歴ログ (retro changelog)	(n.) 変更を、システムに適用された順番ではなく、ローカルサーバーに到着した順序で保存します。旧バージョン形式の更新履歴ログは、マルチマスターレプリケーション環境で機能するようには設計されていません。「 更新履歴ログ (change log) 」とは異なり、旧バージョン形式の更新履歴ログはレプリケーションで使用されません。Directory Server 4 との下位互換性を維持します。
再利用可能なコンポーネント (reusable component)	(n.) 複数の容量、たとえば複数のリソースまたはアプリケーションが使用できるように作成されたコンポーネント。
リバース DNS 検索 (reverse DNS lookup)	(n.) 「DNS (ドメインネームシステム)」のクエリーを行なって、数値「 IP アドレス (IP address) 」を同等の「 ゲートウェイ (gateway) 」に解決するプロセス。
リバースプロキシ (reverse proxy)	(n.) クライアントとサーバーの間で、双方向の URL リライティングと変換を実行するプロキシ。クライアント側に置かれる通常のプロキシとは異なり、リバースプロキシはネットワークのサーバー側に配置されます。Java Enterprise System Portal Server でのリバースプロキシは、Java Enterprise System Portal Server Secure Remote Access Pack にあります。
リライト (Rewriter)	(n.) リライトは、URL 参照を HTML、Javascript、HTTP などのさまざまな Web 言語および HTTP ロケーションヘッダー (リダイレクション) 形式で書き換えるための Java™ クラスライブラリを提供します。リライトは、Java Enterprise System Directory Server Access Management Edition サービスを定義して、書き換え方法と書き換えデータを規定した規則を保存します。またリライトには、これらの規則を編集するための管理コンソールモジュールも組み込まれています。
書き換え規則 (rewrite rule)	(n.) 「 MTA (メッセージ転送エージェント) 」が配信メッセージを正しいホストにルーティングするために使用するツール。書き換え規則には、次の機能があります。(1) 受信メッセージのアドレスからホストまたはドメインの仕様を抽出する。(2) ホストまたはドメイン仕様を書き換え規則のパターンと照合する。(3) ドメインテンプレートに基づいてホストまたはドメイン仕様を書き換える。(4) メッセージを置くチャネルキューを決定する。ドメイン書き換え規則とも呼ばれます。
RFC	(request for comments) (n.) Internet Engineering Task Force によって保持されるドキュメントシリーズ。インターネットプロトコル群および関連する実験結果について説明

しています。インターネット標準について記述した RFC の数はわずかですが、すべてのインターネット標準が RFC として公開されています。
<http://www.imc.org/rfcs.html> を参照してください。

RMI(リモートメソッド呼び出し)

(RMI (remote method invocation)) (n.) ある Java 仮想マシン内で実行中のオブジェクトが、別の Java 仮想マシン内で実行中のオブジェクト上のメソッドを呼び出せるようにするテクノロジー。

RMI-IIOP

(n.) **CORBA (共通オブジェクトリクエストブローカーアーキテクチャー)** **IIOP** プロトコルを使用するために実装されている RMI のバージョン。すべてのリモートインタフェースが最初から RMI インタフェースとして定義されている場合に、RMI over IIOP によって、実装されている言語に関わりなく COBRA オブジェクトとの相互運用が可能になります。

RMIC

(n.) リモートメソッド呼び出しコンパイラ。

ロボット (robot)

(n.) ネットワークの特定の部分に配置されたすべてのリソースを検出するプログラム。

ロボットアプリケーション関数 (robot application function)

「**RAF (ロボットアプリケーション関数)**」を参照してください。

ロール (role)

(1) (n.) ロールは、ユーザーを論理的にグループ化した抽象的なもので、アプリケーションアセンブラによって定義されます。アプリケーションが配備されると、ロールは操作環境でユーザー (主体) やグループなどのセキュリティーアイデンティティーにマップされます。「**ユーザー**」、「**グループ (group)**」も参照してください。

(2) (n.) J2EE サーバー「**認証 (authentication)**」サービスで、特定のリソースセットにアクセスするためのアクセス権の抽象名。

(3) (n.) Java Enterprise System Directory Server Access Management Edition で、権限が与えられた操作の選択を表すグループ化。ユーザーまたはサービスにロールを割り当てることにより、主体は操作を実行できます。たとえば、特定の権限を Employee ロールまたは Manager ロールに限定してそのロールをユーザーに適用することにより、ユーザーのアクセス可能範囲は、適用されたロールに与えられている権限の範囲に限定されます。ロールは、アクセス制御命令 (「**ACI (アクセス制御命令)**」) を使用して定義します。

(4) (n.) J2EE テクノロジーを使用して開発されるアプリケーションの開発および開発フェーズで、関係者が果たす役割。ロールには、アプリケーションコンポーネントプロバイダ、アプリケーションアセンブラ、デプロイヤ、J2EE 製品プロバイダ、EJB コンテナプロバイダ、EJB サーバープロバイダ、Web コンテナプロバイダ、Web サーバープロバイダ、ツールプロバイダ、およびシステム管理者があります。

ルールに基づく属性 (role-based attribute)	(n.) エントリが、関連付けられた「 CoS(サービスクラス) 」テンプレート内に特定のルールを所有しているために、エントリに表示される属性。
ルールマッピング (role mapping)	(n.) 配備記述子に指定されたセキュリティールールに基づいてコンテナによって識別されたグループまたは主体(あるいはその両方)を関連付ける処理。セキュリティールールは、コンポーネントをサーバーにインストールする前に、デプロイヤによりマップされる必要があります。
ロールバック (rollback)	(n.) 「 トランザクション(transaction) 」の取り消し。トランザクションに関係するすべてのリソースに加えられた更新がすべて元に戻される、「 トランザクション(transaction) 」の一時点。
ルート (root)	(1) (n.) UNIX のみ。UNIX マシンで最大の権限が与えられているユーザー。root ユーザーは、マシンのすべてのファイルへのアクセス権限を持っています。 (2) (n.) XML ドキュメントの最も外側の要素。この要素は他のすべての要素を含んでいます。
ルート DN (root DN)	(n.) 「 ディレクトリマネージャー(Directory Manager) 」の「 DN 」。
Root DSE	(n.) Directory Server によって自動的に生成され、空(ゼロバイト長)の「 DN 」による baseObject 検索から返されるエントリ。Root DSE は、「 サブスキーマエントリ(subschema entry) 」へのポインタ、サーバーが保持するネーミングコンテキストの DN のリスト、サーバーがサポートする「 LDAPv3 」の制御機能や拡張機能のリストなど、サーバーの設定に関する情報をクライアントに提供します。「 DSE(ディレクトリサーバーエントリ) 」も参照してください。
ルートエントリ (root entry)	(n.) 「 DIT 」階層の最上位エントリ。
ルートサフィックス (root suffix)	(n.) 1つ以上の「 サブサフィックス(sub suffix) 」の親。ディレクトリツリーには複数のルートサフィックスを収容することができます。
ルーター (router)	(n.) ネットワークトラフィックが流れるパスを決定するシステム。ルーターは、ルーティングプロトコルを使用してネットワークに関する情報を取得し、アルゴリズムを使用して「 ルーティングマトリクス 」と呼ばれるいくつかの条件に基づいて最善のパスを選択します。Open Systems Interconnect 用語でのルーターは、ネットワーク層の中間システムです。「 ゲートウェイ(gateway) 」も参照してください。
ルーティング (routing)	「 メッセージルーティング(message routing) 」を参照してください。
ルーティングテーブル (routing tables)	(n.) メッセージの発信者と受取人についての情報を保持している内部データベース。

行 (row)	(n.) テーブル内の各列の値を格納する1つのデータレコード。
RowSet オブジェクト (RowSet object)	(n.) データベースまたはほかのソースの表形式データから取得した一連の行をカプセル化するオブジェクト。RowSet オブジェクトは、 <code>java.sql.ResultSet</code> インタフェースを拡張することにより、「ResultSet オブジェクト (ResultSet object)」が JavaBeans™ コンポーネントアーキテクチャーに基づくコンポーネントとして動作できるようにします。
RPC (遠隔手続き呼び出し)	(RPC (remote procedure call)) (n.) リモートオブジェクトまたはサービスにアクセスするメカニズム。
RTT (ラウンドトリップ時間)	RTT (round trip time) (n.) サーバー - クライアント - サーバーの閉回路で、信号を送信するまでの経過時間。双方向の対話型通信では RTT の値がスループットレートに直接影響するので、このような通信を行う必要があるシステムでは、この遅延が重要になります。Java Enterprise System Directory Server において、RTT および TCP ウィンドウの値は、ワイドエリアネットワークでのレプリケーションのパフォーマンスに著しい影響を与えます。ラウンドトリップ遅延時間とも呼ばれます。
規則 (rules)	(n.) 条件が満たされているかどうかを判断するために適用される論理的なテスト。ロボットは、インデックス化するコンテンツのタイプを判断するフィルタの一部として規則を使用し、リソースに割り当てるカテゴリを判断するために分類規則で規則を使用します。

S

- SAAJ** (SOAP with Attachments API for Java) (n.)SOAP (シンプルオブジェクトアクセスプロトコル) メッセージングの基本パッケージ。SAAJにはSOAPメッセージの作成と読み込みのためのAPIが含まれています。
- SAF** (サーバーアプリケーション関数) (SAF (server application function)) (n.) 要求処理やその他のサーバーアクティビティーに関係する関数。
- 安全なファイルシステム (safe file system)** (n.) ログを実行するファイルシステム。これにより、システムがクラッシュした場合に、データをクラッシュ前の状態にロールバックしてすべてのデータを復元することができます。安全なファイルシステムの例としては、Veritas File System (VxFS) などがあります。
- SASL** (SASL (simple authentication and securitylayer)) (n.) POP、IMAP、または「SMTP」クライアントが、サーバーに対して自身を識別するメカニズムを制御する手段。Java Enterprise System Messaging Server でサポートするSMTP SASLの使用法は、RFC 2544 (ESMTP AUTH) に準拠しています。SASLは、RFC 2222に定義されています。「POP3 (ポストオフィスプロトコルバージョン3)」、「IMAP4 (インターネットメッセージアクセスプロトコルバージョン4)」も参照してください。
- SAX** (Simple API for XML) (n.) 構文解析イベント発生時に「パーサー (parser)」が呼び出し元から提供されているメソッドのいずれかを呼び出す、イベント駆動型インタフェース。イベントには、「XML (拡張可能マークアップ言語)」タグの認識、エラーの検出、外部エンティティーへの参照の検索、「DTD (ドキュメントタイプ定義)」仕様の処理が含まれます。
- スキーマ** (1) (n.) ディレクトリにエントリとして格納可能な情報のタイプについて規定した定義。スキーマと一致しない情報がディレクトリに格納されていると、ディレクトリにアクセスするクライアントが適切な結果を表示できない場合があります。
- (2) (n.) 「データベース (database)」の表と列の構造。Application Serverでは、スキーマは「エンティティー Bean (entity bean)」から自動的に生成されます。

(3)(n.) XML ベースの言語を使用して「XML (拡張可能マークアップ言語)」ドキュメントに対する制約を指定する、データベースに着想を得たメソッド。スキーマは、特定のフィールドに対してデータの種別を制限できないといった、「DTD (ドキュメントタイプ定義)」ファイルの不備に対応します。スキーマは XML 上に配置されて、階層をなします。したがって、曖昧さのない仕様を作成することが一層容易になり、コメントを適用することが意図されている範囲を判定することもできます。

スキーマ検査
(schema
checking)

(n.) ディレクトリに追加または変更されたエントリが、定義されたスキーマに準拠していることを検証するプロセス。スキーマ検査はデフォルトで有効になっており、ユーザーがスキーマに準拠していないエントリを保存しようとするときエラーが表示されます。

スキーマ名
(schema name)

(n.) 「SOIF」のスキーマまたはタイプ。たとえば、ドキュメントの SOIF のスキーマ名は @DOCUMENT、「リソース記述メッセージ (resource description message)」ヘッダーの SOIF のスキーマ名は @RDMHeader になります。

SCM

「サービス制御マネージャー (service control manager)」を参照してください。

スコーピング
(scoping)

(n.) インポートエージェントによってインポートされるリソース記述に適用される制限。使用する構文は、ユーザー検索の場合と同じです。

検索ベース

「ベース DN (ベース識別名)」を参照してください。

検索データベース
(Search
database)

(n.) 通常はロボットによって生成されるリソース記述の検索可能なデータベース。「ロボット (robot)」も参照してください。

検索データ隠ぺい規則
(search
data hiding rule)

(n.) Directory Proxy Server が検索操作の結果にフィルターを適用してクライアントに返す方法を決定する規則。

検索エンジン
(Search Engine)

(n.) Portal Server 6.0 に組み込まれている検索機能。以前は Compass Server (Portal Server 3.0) と呼ばれていました。検索エンジンは、ロボットによって収集されたリソース記述のデータベース (通常は分類される) を保持します。ユーザーは、リソース記述を検索するか、カテゴリ全体を参照して特定のリソースを検出することができます。

二次データ
ビュー
(secondary data
view)

(n.) 結合データビューを構成する 2 つの Directory Proxy Server データビューの一方。一般に、二次データビューは主データビューのエントリに関する追加情報を提供します。「主データビュー (primary data view)」も参照してください。

二次ディレクトリ
サーバー
(secondary
directory server)

(n.) 優先ディレクトリサーバーを使用できないときに、Identity Synchronization for Windows が使用することのできる、マルチマスターレプリケーション環境のマスターディレクトリサーバーマスターインスタンス。優先ディレクトリサーバーが使用できない間、Identity Synchronization for Windows は Active Directory または Windows NT で行われた変更を二次ディレクトリサーバーに同期することはできますが、二次

ディレクトリサーバーやその他のディレクトリサーバーマスターで行われた変更は、優先ディレクトリサーバーが使用可能になるまで同期されません。

Secure Remote Access (SRA)	(n.) SRA により、ほとんどのクライアントデバイスは、個人用のポータルアプリケーション、コンテンツ、ファイル、およびサービスにセキュリティ保護された接続によってアクセスできます。Sun Java™ System Portal Secure Remote Access (SRA) とも呼ばれます。
Secure Socket Layer	「 SSL 」を参照してください。
セキュリティ (security)	(n.) 認証されたクライアントだけがアプリケーションリソースにアクセスできるようにしたスクリーニングメカニズム。
セキュリティ属性 (security attribute)	(n.) 「 主体 (principal) 」に関連付けられる「 属性 (attribute) 」。セキュリティ属性は、「 認証 (authentication) 」プロトコルまたは J2EE 製品プロバイダ、あるいはその両方によって主体に関連付けることができます。
セキュリティ制約 (security constraint)	(n.) 宣言によって Web コンテンツの対象となる保護に注釈を付けること。セキュリティ制約は、「 Web リソースコレクション (web resource collection) 」、「 承認制約 (authorization constraint) 」および「 ユーザーデータ制約 (user data constraint) 」からなります。
セキュリティコンテキスト (security context)	(n.) 2つのエンティティ間のセキュリティに関する共有状態情報をカプセル化するオブジェクト。
セキュリティモジュールデータベース (security-module database)	(n.) 「 SSL 」暗号化方式用のハードウェアアクセラレータを記述する情報を含むファイル。secmod と呼ばれます。
セキュリティアクセス権 (security permission)	(n.) J2SE で定義されるメカニズムで、アプリケーションコンポーネント開発者に課されるプログラミング上の制約を表すために J2EE プラットフォームで使用されています。
セキュリティアクセス権セット (security permission set)	(n.) J2EE 製品プロバイダが各コンポーネントタイプの実行のために提供する必要のある、セキュリティアクセス権の最小セット。
セキュリティポリシードメイン (security policy domain)	「 レルム (realm) 」を参照してください。
セキュリティロール (security role)	「 ロール (role) 」を参照してください。

セキュリティ テクノロジド メイン (security technology domain)	(n.) 同一のセキュリティメカニズムを使用してセキュリティポリシーを実施する有効範囲。1つのテクノロジドメイン内に複数のセキュリティテクノロジドメインが存在することは可能です。
セキュリティ ビュー (security view)	(n.) アプリケーションアセンブラーによって定義されるセキュリティロールのセット。
自己アクセス (self access)	(n.) 「 バインド DN (バインド識別名) 」がターゲットエントリに一致する場合に、権限を付与されると、ユーザーが所有するエントリに対するアクセス権を持つことを示します。
自己生成証明書 (self-generated certificate)	(n.) 自己の公開値のメッセージダイジェストを使用してエントリが名前指定された場合、およびそれらの名前がセキュリティ保護された状態で通信される場合にのみ使用される公開鍵の値。「 発行された証明書 (issued certificate) 」も参照してください。
sendmail	(n.) UNIXのみ。共通の「 MTA (メッセージ転送エージェント) 」です。ほとんどのアプリケーションでは、sendmailの代わりにJava Enterprise System Messaging Serverを使用できます。
直列化可能オブ ジェクト (serializable object)	(n.) 解体および再構築できるオブジェクト。複数のサーバーに保存したり分散したりすることができます。
サーバー	(n.) 外部インタフェースを通してサービスにアクセスするクライアントに分散型または関連する一連のサービスを提供する、マルチスレッド対応のソフトウェアプロセス。ハードウェアのサーバーとは区別されます。
サーバー管理者 (server administrator)	(n.) サーバー管理タスクを実行するユーザー。サーバー管理者は、タスク「 ACI (アクセス制御命令) 」に応じて、特定のサーバーへのタスクに制限付きのアクセス権を提供します。構成管理者は、ユーザーにサーバーへのアクセス権を割り当てる必要があります。サーバーへのアクセス権を与えられたユーザーは、サーバー管理者となり、サーバーへのアクセス権をユーザーに与えることができます。
サーバーアセン ブリ (server assembly)	(n.) ユニットとしてインストールまたは配備される場合に、相互に依存または緊密に連携するJava Enterprise Systemサーバーのグループ。
サーバー認証 (server authentication)	(n.) セキュリティ保護されたサーバーに接続していることをクライアントが確認できる認証方法。これにより、別のコンピュータがサーバーを装ったり、サーバーがセキュリティ保護されていないときにセキュリティ保護されているかのように見せかけたりすることができないようにします。

サーバー証明書 (server certificate)	(n.) HTTPS プロトコルとともに、Web アプリケーションの認証に使用されます。証明書は自己署名付きまたは認証局 (CA) による承認付きにすることができます。Application Server の HTTPS サービスは、サーバー証明書がインストールされるまで実行されません。
サーバーデーモン (server daemon)	(n.) 実行されているときに、クライアントからの要求を待機して受け入れるプロセス。
サーバーファーム (server farm)	(n.) Web Server において、サーバーファームとは異なる設定を実行する1つ以上のノードのネットワークです。それとは対照的に、「 クラスタ 」は同一の設定および Web アプリケーションを実行するノードのネットワークです。
サーバーインスタンス (server instance)	(1) (n.) Application Server は、同じマシンの同じインストールに複数のインスタンスを持つことができます。各インスタンスには、専用のディレクトリ構造、設定、配備アプリケーションがあります。各インスタンスに複数の仮想サーバーを持たせることもできます。「 仮想サーバー (virtual server) 」も参照してください。 (2) (n.) Directory Server または Directory Proxy Server のインスタンス。インスタンスはインスタンスパスによって定義され、関連するデータベースおよび設定ファイルがあります。複数のインスタンスを1つのホストシステム上で実行することができます。
Server Message Block プロトコル (Server Message Block protocol)	(n.) コンピュータのクライアントアプリケーションが、ファイルへの読み書きを行ったり、コンピュータネットワーク内のサーバープログラムからサービスを要求できるようにするプロトコル。SMB プロトコルは、トランスミッション・コントロール・プロトコル (TCP) の上位プロトコル、または Internetwork Packet Exchange や NetBEUI など、ほかのネットワークプロトコルの上位プロトコルとして、インターネット上で使用できます。Java Enterprise System Portal Server では、NetFile 対応の SMB を使用します。
サーバープラグイン API (server plug-in API)	(n.) Java Enterprise System サーバーの主要機能の拡張およびカスタマイズを可能にするとともに、HTTP サーバーとバックエンドアプリケーションの間にインタフェースを構築する、スケーラブルで有効なメカニズムを提供する拡張機能。NSAPI とも呼ばれます。
サーバー主体	(n.) サーバーが実行されるときのオペレーティングシステム「 主体 (principal) 」。
サーバープロセス (server process)	(n.) オペレーティングシステムによって設定される、自己完結で完全に機能する実行環境。アプリケーションの各インスタンスは、一般に別々のプロセスで実行されます。
サーバールート (server root)	(1) (n.) サーバープログラムと設定ファイル、保守ファイル、および情報ファイルを保持する、サーバーマシンの専用ディレクトリ。ServerRoot または「 ドメインディレクトリ (domain directory) 」とも呼ばれます。

(2) (n.) サーバー上のほかのファイルに対する相対的なディレクトリ位置。たとえば、Solaris システムにインストールされた Calendar Server は、デフォルト時、サーバールートとしてパス /opt/SUNWics5/ を使用します。

(3) (n.) 特定のホスト上にある Java Enterprise System Administration Server に関連付けられたすべての Java Enterprise System サーバーがインストールされているディレクトリ。「インストールディレクトリ (installation directory)」および「インスタンスディレクトリ (instance directory)」も参照してください。

サーバー側規則
(server side rules)

(n.) サーバー側でのメールのフィルタを可能にする一連の規則。Sieve メールフィルタ言語に基づいています。

サービス

(1) (n.) サーバーが提供する機能。たとえば、Java Enterprise System Messaging Server は、SMTP、POP、IMAP、HTTP などのサービスを提供します。

(2) (n.) 1 つ以上のクライアントのために実行されるソフトウェア機能。この機能は、メモリー管理のように非常に低レベルで実行されたり、信用検査「ビジネスサービス (business service)」のように高レベルで実行されたりします。高レベルサービスは、個々のサービスのファミリで構成できます。またサービスには、ローカルクライアントが使用可能なローカルサービスや、リモートクライアントが使用可能な分散サービスがあります。

サービス制御マネージャー
(service control manager)

(n.) Windows NT のみ。サービス管理で使用する管理プログラム。

サービス要素
(service element)

(n.) 着信要求の処理に単一のエンジンコンポーネントを共有する 1 つまたは複数の「コネクタ (connector)」要素の組み合わせ。

サービスエンドポイントインタフェース
(service endpoint interface)

(n.) クライアントが「Web サービス (web service)」上に呼び出すことのできるメソッドを宣言する Java インタフェース。

サービス品質コンポーネント
(service quality component)

(n.) Java Enterprise System に組み込まれている多くの種類の「システムコンポーネント (system component)」の 1 つ。アクセスコンポーネントや管理コンポーネントなどのサポートコンポーネントにより、「システムサービス」をサポートします。

サービス指向アーキテクチャー
(service-oriented architecture)

「SOA (サービス指向アーキテクチャー)」を参照してください。

サービスプロバイダ (service provider)	(n.) Web ベースのサービスを提供する商業目的または非営利の組織。インターネットポータル、小売業、輸送手段の提供者、金融機関、エンターテインメント企業、図書館、大学、政府機関などがあります。
Service Registry	(n.) 「Java Enterprise System」に組み込まれている「ebXML レジストリ (ebXML registry)」製品。
サービススタック (service stack)	(n.) 分散エンタープライズアプリケーションのサポートに必要な、分散サービスの階層構造。この階層構造には、スタック内の下位サービスに対する高レベルサービスの依存性が反映されます。
サーブレット (servlet)	<p>(1)(n.) 動的コンテンツを生成したり、要求応答パラダイムを使用して Web アプリケーションと対話したりすることにより Web サーバーの機能を拡張する、Java™ プログラミング言語で作成されたサーバーサイドプログラム。サーブレットは、サーバー側で実行される点でアプレットに似ていますが、ユーザーインタフェースを使用しません。</p> <p>(2)(n.) Servlet クラスのインスタンス。サーブレットは、サーバー上で実行する再利用可能なアプリケーションです。Java Enterprise System Application Server では、サーブレットは、プレゼンテーションロジックの実行、ビジネスロジックの起動、およびプレゼンテーションレイアウトの起動または実行によって、アプリケーションでの対話ごとに中心的なディスパッチャとしての役割を果たします。</p>
サーブレットコンテナ (servlet container)	(n.) 要求と応答の送信、要求のデコード、および応答の書式設定で使用するネットワークサービスを提供する「コンテナ (container)」。すべてのサーブレットコンテナは要求と応答のプロトコルとして HTTP を必ずサポートしていなければなりません。そのほかにも HTTPS などの要求応答プロトコルをサポートすることもできます。
サーブレットコンテナ、distributed (servlet container, distributed)	(n.) distributable としてタグ付けされ、同一または異なるホスト上で稼働する複数の Java 仮想マシンの中で動作する Web アプリケーションを実行可能なサーブレットコンテナ。
サーブレットコンテキスト (servlet context)	(n.) その中でサーブレットが実行中である Web アプリケーションに対するサーブレットのビューを含むオブジェクト。このコンテキストを使用することにより、サーブレットはイベントのログを記録し、リソースへの URL 参照を取得し、コンテキスト内の他のサーブレットで使用可能な属性を設定および格納することができます。
サーブレットエンジン (servlet engine)	(n.) すべてのサーブレットメタ関数を処理する内部オブジェクト。一般的には、インスタンス化および実行などのサービスをサーブレットに提供する一連のプロセスです。

サーブレットマッピング (servlet mapping)	(n.) URL パターンとサーブレットの間の関連付けを定義します。サーブレットに対する要求の割り当てには、マッピングが使用されます。
サーブレットランナー (servlet runner)	(n.) 要求オブジェクトおよび応答オブジェクトを持つサーブレットを起動するサーブレットエンジンの一部。「セッション Bean (session bean)」を参照してください。
セッション	<p>(1) (n.) 複数の HTTP 要求によるユーザーと J2EE または Web アプリケーションとの間の対話を追跡するために、サーブレットまたはステートフルセッション Bean が使用するオブジェクト。「持続性 (persistence)」も参照してください。</p> <p>(2) (n.) クライアント - サーバー接続のインスタンス。「クライアントサーバーモデル (client-server model)」も参照してください。</p> <p>(3) (n.) Java Enterprise System Portal Server の場合は、ログインで開始されてログアウトまたはタイムアウトで終了する、ユーザーと1つ以上のアプリケーションの間で行われる一連の対話操作。</p> <p>(4) (n.) Message Queue の場合は、メッセージを送受信するためのシングルスレッドのコンテキスト。キューセッション、またはトピックセッションのどちらかになります。</p>
セッション Bean (session bean)	(n.) クライアントによって作成され、通常は、1回のクライアントサーバーセッションの期間にのみ存在するエンタープライズ Bean。セッション Bean は、クライアントのために計算やほかのエンタープライズ Bean へのアクセスなどを実行します。セッション Bean は、トランザクションで使用されることもありますが、システムがクラッシュした場合にはリカバリ不能です。セッション Bean オブジェクトには、特定のクライアントに関連付けられないステートレス、および特定のクライアントと関連付けられるステートフルがあり、メソッドやトランザクションの間で対話状態を保持することができます。「ステートフルセッション Bean (stateful session bean)」も参照してください。
セッション cookie (session cookie)	(n.) ユーザーセッション識別子が含まれているクライアントに返される cookie。「スティッキー cookie (sticky cookie)」も参照してください。
セッションフェイルオーバー (session failover)	(n.) Access manager におけるフェイルオーバーの実装。Sun Java System Message Queue を通信ブローカーとして、Berkeley DB をセッションストアデータベースとして使用します。この実装では、Web コンテナセッション管理機能は使用しません。Access Manager セッションフェイルオーバーは、単一のハードウェアまたはソフトウェアの障害発生時に、ユーザーの認証セッション状態を維持するため、セッション情報を失ったり、ユーザーの再ログインを必要としたりせずに、ユーザーのセッションをセカンダリ Access Manager インスタンスにフェイルオーバーできます。「フェイルオーバー (failover)」、「持続性 (persistence)」も参照してください。

セッション鍵	(n.) 別々の鍵を使用して2ユーザー間での個々の会話を暗号化する、一般的な暗号化技術。
セッションタイムアウト (session timeout)	(n.) ユーザーセッションの有効期限。この特定の時間が経過すると、サーバーによってユーザーセッションが無効になります。
SGML	(Standard Generalized Markup Language) (n.) 「HTML (ハイパーテキストマークアップ言語)」と「XML (拡張可能マークアップ言語)」の母体。HTMLは表示情報をマークアップに埋め込むというSGMLの性質を受け継いでいますが、XMLは情報コンテンツそのものをコンテンツをレンダリングするメカニズムから全く分離できる標準です。
共有コンポーネント (shared component)	(n.) Java Enterprise System に組み込まれている多くの種類の「システムコンポーネント (system component)」の1つ。ライブラリなどの共有コンポーネントは、ほかのシステムコンポーネントにローカルサービスを提供します。一方、「システムサービス」は、ほかのシステムコンポーネントまたは「アプリケーションコンポーネント (application component)」に、分散インフラストラクチャーサービスを提供します。
共有コンポーネントの記述子 ファイル (shared component descriptor file)	(n.) 所定の共有コンポーネントのメタデータを、通常はXML形式で記述したファイル。
共有フォルダ (shared folder)	(n.) 複数のユーザーが読み取り可能なフォルダ。共有フォルダに対しては所有者が指定されます。所有者は、フォルダに対する読み取りアクセス権を指定したり、共有フォルダからメッセージを削除したりできます。共有フォルダにはモデレータを指定することもできます。モデレータは、受信メッセージの編集、ブロック、転送を行うことができます。共有できるのは、IMAPフォルダだけです。「個人用フォルダ (personal folder)」、「公開フォルダ (public folder)」も参照してください。
共有鍵暗号方式 (shared-key cryptography)	(n.) 当事者それぞれが同じ鍵を使用して暗号化テキストの暗号化または復号化を行う暗号化方式。対称鍵暗号方式とも呼ばれます。
SHTML	(server-side include Hypertext markup language) (n.) サーバー側インクルード (SSI) が埋め込まれているHTMLファイル。
Sieve	(n.) メールフィルタ用に提案されている言語。
Simple API for XML	「SAX」を参照してください。
シンプル認証とセキュリティ層 (simple authentication and security layer)	「SASL」を参照してください。

シンプルインデックス (simple index)	(n.) 画像要素がなくファイル名だけを表示する、ディレクトリのリスト。ファンシーインデックスの反対です。
Simple Mail Transfer Protocol	「SMTP」を参照してください。
Simple Network Management Protocol	「SNMP」を参照してください。
シンプルオブジェクトアクセスプロトコル (Simple Object Access Protocol)	「SOAP (シンプルオブジェクトアクセスプロトコル)」を参照してください。
SIMS	(n.) Solstice Internet Mail Server™ および Sun Internet Mail Server™。
単一フィールド置換文字列 (single field substitution string)	(n.) 書き換え規則において、ホストまたはドメインアドレスの指定アドレストークンを動的に書き換えるドメインテンプレートの一部分。「ドメインテンプレート (domain template)」も参照してください。
シングルアイデンティティ (single identity)	(n.) Java Enterprise System ディレクトリの1つのユーザーエントリに基づいてユーザーが保持しているアイデンティティ。この1つのユーザーエントリに基づき、ユーザーは、ポータルや Web ページなどの各種 Java Enterprise System リソースや、メッセージング、カレンダー、およびインスタントメッセージングなどのサービスにアクセスできます。
シングルログアウト (single logout)	(n.) アイデンティティプロバイダまたはサービスプロバイダからログアウトすると、その認証ドメインのすべてのサービスプロバイダまたはアイデンティティプロバイダからログアウトされる仕組み。
シングルサインオン (SSO) (single sign-on (SSO))	<p>(1) (n.) 分散システム内の1つのサービスに対するユーザーの認証が、同じシステムのほかのサービスにも自動的に適用される機能。</p> <p>(2) (n.) 1つの仮想サーバーインスタンスの複数の J2EE アプリケーションでユーザーの認証状態を共有できる状態。「SSO」を参照してください。</p> <p>(3) (n.) 連携アイデンティティを持つユーザーがアイデンティティプロバイダに対して認証操作を実行するときに確立される認証プロセス。ユーザーは、連携アイデンティティを持っているので、再認証しなくても連携サービスプロバイダにアクセスできます。</p>
サイト (site)	(n.) 「ロボット (robot)」がリソースを検索するネットワーク上の位置。サイトのアドレス、およびインデックス化するサイトのドキュメントの種類は、「サイト定義 (site definition)」で指定できます。

サイト設定 (site configuration)	(n.) Access Manager クライアントが複数の負荷分散 Access Manager インスタンスと通信できるようにした、シンプルな設定を提供する機能。サイト設定は、各サイトの周囲に複数のロードバランサとファイアウォールを備えた配備をサポートします。
サイト定義 (site definition)	(n.) ロボットがリソースの位置を検索できる場所に適用される制約。サイト定義を使用することにより、ロボットの動作範囲を、特定のサーバー、指定されたサーバーグループ、またはドメインに限定できます。サイト定義には、ロボットがインデックス化するサイトのドキュメントのタイプを記述したフィルタもあります。
SIZE	(n.) クライアントが特定のメッセージのサイズをサーバーに対して宣言できるようにする「SMTP」拡張機能。サーバーは、宣言されたメッセージサイズに基づいて、メッセージを受け入れること、または受け入れないことを、クライアントに示します。サーバーは、受け入れるメッセージの最大サイズをクライアントに宣言できません。RFC 1870 に定義されています。
slapd.exe	(n.) Windows のみ。Directory Server のすべての動作の責任を持つプロセスまたはサービス。UNIX システムの「 ns-slapd 」に相当します。
スレーブチャネルプログラム (slave channel program)	(n.) リモートシステムによって開始された転送を受け入れるチャネルプログラム。「 マスターチャネルプログラム (master channel program) 」も参照してください。
スマートホスト (smart host)	(n.) ほかのメールサーバーが受取人を認識できない場合に、メッセージの転送先となる、ドメイン内のメールサーバー。
SMB プロトコル (SMB protocol)	「 Server Message Block プロトコル (Server Message Block protocol) 」を参照してください。
SMTP	(Simple Mail Transfer Protocol) (n.) インターネットで一般的に使用されている電子メールプロトコルで、Java Enterprise System Messaging Server でもサポートされています。RFC 821 で定義され、関連するメッセージ形式は RFC 822 で説明されています。
SMTP AUTH	「 AUTH 」を参照してください。
SMTP プロキシ (SMTP proxy)	(n.) ネットワークを介して1つのコンピュータから別のコンピュータにメッセージを送信し、電子メールの経路指定のためにインターネットで使用される SMTP の一種。
sn 属性 (sn attribute)	(n.) 苗字を表す LDAP エイリアス。
SNMP	(Simple Network Management Protocol) (n.) ネットワークの稼動状況に関するデータを交換するために使用されるプロトコル。SNMP を使用して、管理対象デバイス

(ネットワーク上のホスト、ルーター、Web サーバー、そのほかのサーバーなどの SNMP に従うすべてのデバイス)と「NMS(ネットワーク管理ステーション)」の間でデータを送受信します。

SNMP マスターエージェント (SNMP master agent)

(n.) さまざまなサブエージェントと「NMS(ネットワーク管理ステーション)」の間で情報を交換するソフトウェア。

SNMP SOCKS

(n.) たとえば、ルーター設定など、ファイアウォールソフトウェアまたはハードウェアによって回避される直接接続の場合に、ファイアウォールの内側から外側に向かって接続を確立するファイアウォールソフトウェア。

SNMP サブエージェント (SNMP subagent)

(n.) 管理対象デバイスの情報を収集し、その情報をマスターエージェントに渡すソフトウェア。

SOA (サービス指向アーキテクチャ)

(SOA (service-oriented architecture)) (n.) サービスのコンシューマとプロバイダからなる複合アプリケーションを記述します。コンシューマとプロバイダは、互いの具体的な場所を参照せずにメッセージを交換できます。またこのアーキテクチャでは、アプリケーションのコアプロセスは他のサービスプロバイダとコンシューマから分離されます。

SOAP (シンプルオブジェクトアクセスプロトコル)

(SOAP (Simple Object Access Protocol)) (n.) 集中管理されていない、分散環境で構造化情報の交換に使用される軽量なプロトコル。このプロトコルは、「XML(拡張可能マークアップ言語)」テクノロジーを使用して、拡張「メッセージング(messaging)」フレームワークを定義します。このフレームワークには、さまざまな基本プロトコルを介して交換可能なメッセージ構文が含まれます。

SOAP with Attachments API for Java

「[SAAJ](#)」を参照してください。

ソフト再起動 (soft restart)

(n.) 設定ファイルを再読み込みすることにより、サーバーを内部で再起動させるサーバーの再起動方法。ソフト再起動では、プロセスに HUP シグナル(シグナル番号 1)を送信します。プロセス自体は終了せず、ハード再起動で終了します。

SOIF

(summary object interchange format) (n.) リソース記述およびほかの種類の構造化オブジェクトを送信する場合の構文。リソース記述それぞれは、属性値ペアのリストとして表されます。SOIF は、テキストデータとバイナリデータの両方を値として、また若干拡張された多値属性を処理します。SOIF は、1つの効率的なストリームで多くのリソース記述をバルク転送することができるストリーム形式です。

SOIF 属性 (SOIF attribute)

(n.) データベース属性のタイプ。検索データベース内のリソース記述それぞれには、複数の属性またはフィールドがあります。これらの属性は SOIF 属性と呼ばれます。

Solaris™ 論理名 (Solaris logical name)	(n.) Solaris オペレーティングシステムのデバイスを管理するために一般に使用される名前。ディスクの場合、これらは通常、 <code>/dev/rdisk/c0t2d0s2</code> のようになります。Solaris 論理デバイス名ごとに、基礎となる Solaris 物理デバイス名があります。「 Solaris 物理名 (Solaris physical name) 」も参照してください。
Solaris 物理名 (Solaris physical name)	(n.) Solaris オペレーティングシステムにおいて、デバイスドライバによってデバイスに与えられる名前。デバイス名は、 <code>/devices</code> ツリーの下のパスとして Solaris マシンに表示されます。たとえば、一般的な SI ディスクの Solaris 物理名は、 <code>devices/sbus@1f,0/SUNW,fas@e,8800000/sd@6,0:c,raw</code> のような形式になります。「 Solaris 論理名 (Solaris logical name) 」も参照してください。
ソリューションライフサイクル (solution life cycle)	(n.) 配備プロジェクトを計画および追跡するツール。このライフサイクル構造には、配備計画を成功させるために必要な準備、分析、および設計が一連の段階として順番に組み込まれています。各フェーズは、後続フェーズへのインプットとして渡されるアウトプットを生成する関連性のあるタスクで構成されています。各フェーズ内のタスクは、そのフェーズのアウトプットを生成する前に徹底的な分析と設計を必要とするもので、繰り返し実行されます。
スペアノード (spare node)	(n.) 障害の発生したアクティブノードに置き換わることが可能な HADB ノード。アクティブノードで障害が発生すると、スペアノードがミラーノードからデータをコピーしてアクティブになります。「 HADB ノード (HADB node) 」、「 アクティブノード (active node) 」、「 ミラーノード (mirror node) 」、および「 データ冗長ユニット (DRU) 」も参照してください。
スパイダ (spider)	「 ロボット (robot) 」を参照してください。
スプーフィング (spoofing)	(n.) ネットワーク攻撃の形態の1つ。虚偽のホスト名を使って、クライアントがサーバーへアクセスまたはメッセージ送信しようとしています。
SQL (構造化クエリ言語)	(SQL (structured query language)) (n.) データベースオブジェクトの定義およびデータの操作のための標準化リレーショナルデータベース言語。SQL2 および SQL3 は、この言語のバージョンを表します。
SQL/J	(n.) Java プログラミング言語で SQL 文をメソッドに組み込むための仕様、および Java 静的メソッドを SQL ストアドプロシージャおよびユーザー定義関数として呼び出すための仕様を含む標準の一式。SQL チェッカは、JDBC ドライバを使用した場合のように実行時ではなく、プログラム開発時に静的 SQL 文内のエラーを検出することができます。
SSL	(Secure Socket Layer) (n.) HTTP や FTP のようなほかのプロトコルで使用される、セキュリティ保護された低レベル暗号化の形式。SSL プロトコルには、サーバー認証、伝送中のデータの暗号化、およびオプションのクライアント認証の仕組みが組み込まれています。このプロトコルは、データ改ざんや盗聴をされない方法でクライアントサーバーアプリケーションが通信できるようにします。

SSL 認証 (SSL authentication)	(n.) 本人であることの証明としてクライアント証明書の情報を使用して、または LDAP ディレクトリで発行されたクライアント証明書を検証して、セキュリティー証明書によりユーザーのアイデンティティーを確認する認証方法。
SSL 証明書 (SSL certificate)	(n.) 「PKI (公開鍵基盤)」を使用してセキュリティー保護されたトランザクションを暗号化および復号化することを、ユーザーまたはベンダーが承認したことを示す電子トークン。Java Enterprise System Portal Server ソフトウェアのインストール時に自己署名済みの SSL 証明書を作成します。ただし、インターネット上での安全な通信サービスを承認する証明書ベンダーから SSL 証明書を取得することも可能です。
SSO	「シングルサインオン (SSO) (single sign-on (SSO))」を参照してください。
SSR	「サーバールート (server root)」を参照してください。
標準インデックス (standard index) 開始位置 (starting points) 状態	<p>(n.) デフォルトで維持されるインデックス。</p> <p>(n.) 検索エンジンロボットがリソースの列挙を開始するサイトのリスト。</p> <p>(1) (n.) 指定された時間におけるエンティティーの環境または状態。</p> <p>(2) (n.) Java Enterprise System Application Server の機能インタフェース IState2 を使用して、アプリケーションの状態を保存するために使用できる分散データ保存メカニズム。「対話状態 (conversational state)」、「持続状態 (persistent state)」も参照してください。</p>
ステートフルセッション Bean (stateful session bean)	(n.) 特定のクライアントとのセッションを表すセッション Bean。複数のクライアント起動メソッドの対話状態を自動的に維持します。
ステートレスセッション Bean (stateless session bean)	(n.) 状態のないサービスを表すセッション Bean。ステートレスセッション Bean は完全に一時的なものであり、ビジネスロジックのうち限られた時間に特定のクライアントが必要とする一時的な部分をカプセル化します。ステートレスセッション Bean のインスタンスはすべて同一です。
スタティックグループ (static group)	(n.) 各グループメンバーを列挙することにより静的に定義されたメールグループ。「ダイナミックグループ (dynamic group)」も参照してください。
静的 Web コンテンツ (static web content)	(n.) 静的 HTML ファイル、画像、アプレット Java アーカイブ (JAR) ファイル、および Java Web コンテナを使用せずに Web サーバーによって直接サービスを提供可能なその他の要素。Java Enterprise System Portal Server の場合、Web ファイルは Web サーバーにインストールされます (動的 Web アプリケーションと同じ場所)。
ステータスイベント (status event)	(n.) オンラインかどうかを含むユーザーのステータス。

スティッキ cookie (sticky cookie)	(n.) 常に同じサーバープロセスにクライアントを強制的に接続させるためにクライアントに返される「cookie」。「セッション cookie (session cookie)」も参照してください。
スティッキロー ドバランス (sticky load balancing)	(n.) 最初のクライアント要求は負荷分散させても、その後の要求は最初の要求と同じプロセスに振り分ける「負荷分散 (load balancing)」の手法。
ストップワード (stop word)	(n.) 検索機能で検索しないワードとして指定されたワード。たとえば、「the」、「a」、「an」、「and」などのワードがあります。ドロップワードとも呼ばれます。
ストアドプロ シーチャー (stored procedure)	(n.) 「SQL (構造化クエリー言語)」で書かれ、データベースに保存される文のブロック。ストアドプロシーチャーを使って、レコードの変更、挿入、または削除などのすべてのタイプのデータベース操作を実行できます。ストアドプロシーチャーを使うと、ネットワークを介して送信される情報量が減るので、データベースのパフォーマンスが向上します。
ストリーミング (streaming)	(n.) 「HTTP」によるデータの通信方法を管理するための技術。結果がストリーミングされれば、そのデータの最初の部分をすぐに使用できます。結果がストリーミングされない場合は、結果全体が取得されるまでどの部分も使用できません。ストリーミングを使うと、大量のデータを効率よく返すことができるため、アプリケーションの体感的なパフォーマンスが向上します。
strftime 関数 (strftime function)	(n.) 日付と時刻を文字列に変換する関数。この関数は、トレーラを追加するときにサーバーが使用します。strftime 関数には、日付と時刻に使用する特殊な形式の言語があり、サーバーはその言語をトレーラで使用して、ファイルの最終変更日付を表示できます。
サブエージェン ト	「SNMP サブエージェント (SNMP subagent)」を参照してください。
サブドメイン (subdomain)	(n.) ドメイン名を保有する企業または組織内部の部門または部署を識別する「ゲートウェイ (gateway)」(support.example.com や sales.example.com など) の最後から2番目の部分。サブドメインは必ず指定されるとは限りません。
サブネット	(n.) ホスト ID のブロックを識別する、「IP アドレス (IP address)」の一部分。
従属参照 (subordinate reference)	(n.) ディレクトリサーバーによって保持されているネーミングコンテキストの子となるネーミングコンテキスト。「知識情報 (knowledge information)」も参照してください。
サブ組織 (suborganization)	(n.) Java Enterprise System Directory Server Access Management Edition では、組織の下に作成され、部門とリソースをより細かに制御するためにエンタープライズが使用す

るオブジェクト。たとえば、Java Enterprise System Portal Server を設定する場合は、最上位オブジェクト `isp` の下に `mycompany` というサブ組織を作成できます。

サブスキーマエントリ
(**subschema entry**)

(n.) ディレクトリツリーの一部のエントリによって使用される、すべてのスキーマ定義(オブジェクトクラス、属性、マッチングルールなどの定義)が収められているエントリ。

部分文字列インデックス
(**substring index**)

(n.) エントリ内の部分文字列に対する検索を効率よく実行できるようにする検索フィルタ。部分文字列インデックスは、インデックスキーあたり最大3文字に限定されています。

サブサフィックス
(**sub suffix**)

(n.) ルートサフィックスの下の分岐。

接尾辞

(n.) それより下位にデータを保存するディレクトリツリーのエントリ名。同じディレクトリ内で複数のサフィックスを使用できます。各データベースにはサフィックスが1つだけあります。

summary object interchange format

「[SOIF](#)」を参照してください。

Sun™ Cluster ソフトウェア (Sun Cluster software)

可用性の高いスケーラブルなサービスを作成するために使用する Sun Cluster ソフトウェア。

Sun Java System Application Server

「[Application Server](#)」を参照してください。

Sun Java System Communications Express

「[Communications Express](#)」を参照してください。

Sun Java System Compass Server

「[Compass Server](#)」を参照してください。

Sun Java System Connector for Microsoft Outlook

「[Connector for Microsoft Outlook](#)」を参照してください。

Sun Java System Delegated Administrator Sun Java System Directory Server

「[Delegated Administrator](#)」を参照してください。

「[Directory Server](#)」を参照してください。

Sun Java System Instant Messaging Client

「[Instant Messaging クライアント \(Instant Messaging Client\)](#)」を参照してください。

Sun Java System Message Queue

「[Message Queue](#)」を参照してください。

Sun Java System Portal Secure Remote Access (SRA)

「[Secure Remote Access \(SRA\)](#)」を参照してください。

**Java System
Portal Server**

「Portal Server」を参照してください。

**Sun Java System
Synchronization**

(n.) Microsoft Windows パーソナルコンピュータで動作し、カレンダーイベントやタスクを、モバイルデバイスおよび Microsoft Outlook などの個人情報マネージャー (PIM) と同期するのを可能にするソフトウェア。

**Sun Java System
Web Server**

「Web Server」を参照してください。

**サプライヤ
(supplier)**

(n.) コンシューマサーバーにレプリケートされるディレクトリツリーまたはサブツリーのマスターコピーを保持しているサーバー。

**サプライヤレプリカ
(supplier replica)**

(n.) ディレクトリ情報のマスターコピーを持ち、更新可能なレプリカ。サーバーは任意の数のマスターレプリカを保持できます。

**サプライヤ
ディレクトリ
サーバー
(supplier
directory server)
symlinks**

(n.) 変更をほかのディレクトリサーバーに送信するディレクトリサーバー。「**コンシューマディレクトリサーバー (consumer directory server)**」も参照してください。

(n.) UNIX のみ。別のファイルまたはディレクトリをポイントして、両方のファイルまたはディレクトリが同じコンテンツを持つようにする特別なファイルまたはディレクトリ。

**対称暗号化
(symmetric
encryption)**

(n.) 暗号化と復号化に同じ鍵を使用する暗号方式。DES (Data Encryption Standard) は、対称暗号化アルゴリズムの一例です。

対称鍵暗号化

「**共有鍵暗号方式 (shared-key cryptography)**」を参照してください。

**同期
(synchronization)**

(1) (n.) マスターディレクトリサーバーのデータによるレプリカディレクトリサーバーのデータの更新。

(2) (n.) 「**MTA (メッセージ転送エージェント)**」ディレクトリキャッシュの更新。

**同期ユーザーリス
ト
(Synchronization
User List)**

(n.) 同期する Sun および Windows ディレクトリのユーザーを定義します。同期ユーザーリストは、LDAP 基本 DN またはフィルタに基づいて同期するユーザーの範囲を制限できます。

**システムコン
ポーネント
(system
component)**

(n.) Java Enterprise System に付属し、Java Enterprise System インストールプログラムによってインストールされる、ソフトウェアパッケージまたはパッケージセット。システムコンポーネントには、分散インフラストラクチャー「**サービス**」を提供する「**サーバー**」、アクセスサービスと管理サービスを提供してシステムサービスコ

ンポーネントをサポートする「システムサービス」、ほかのシステムコンポーネントにローカルサービスを提供する「共有コンポーネント (shared component)」など、数種類のコンポーネントがあります。

システムイン
デックス
(system index)

(n.) Directory Server の動作に欠かせないために、削除または変更することができないインデックス。

システムサー
ビス

(n.) Java Enterprise System によって提供される固有の機能を定義する、1つ以上の分散「サービス」。システムサービスでは、通常、多くの「サプライヤ (supplier)」、または多くの「共有コンポーネント (shared component)」のサポートを必要とします。

システムサー
ビスコンポー
ネン
ト (system
service
component)

(n.) Java Enterprise System に組み込まれている多くの種類の「システムコンポーネント (system component)」の1つ。システムサービスコンポーネントは、主要な Java Enterprise System インフラストラクチャーサービスであるポータルサービス、通信と連携動作サービス、アイデンティティーとセキュリティーサービス、Web とアプリケーションサービス、および可用性サービスを提供します。

T

タグ (tag)	(n.) 「XML (拡張可能マークアップ言語)」ドキュメントで、データの単位または要素を記述するテキスト部分。タグは山括弧(<と>)で囲まれているので、データではなく、マークアップとして識別可能です。このようなマークアップ構文をデータとして扱うには、エンティティー参照またはCDATA セクションを使用します。
テイクオーバー (takeover)	「フェイルオーバー (failover)」を参照してください。
ターゲット (target)	(1) (n.) アクセス制御において、ターゲットは、特定の「ACI (アクセス制御命令)」が適用されるディレクトリ情報を識別します。 (2) (n.) Application Server において、ターゲットはアプリケーション配備や設定の変更が適用されるサーバーインスタンスです。 (3) (n.) Apache Ant において、ターゲットは実行する一連の作業です。「asant」、 「構築ファイル (build file)」も参照してください。
ターゲットのエントリー (target entries)	(n.) 「CoS (サービスクラス)」の範囲内のエントリー。
タスク (task)	(n.) クライアント側の Calendar Express において、行うべきことを指定するカレンダーコンポーネント。サーバー側では、タスクのことを「todo」とも呼びます。
分類 (taxonomy)	(n.) Java Enterprise System Portal Server Search Engine のリソースのカテゴリ分類方法。
telnet プロキシ (telnet proxy)	(n.) telnet クライアントと telnet サーバーの間に存在し、インテリジェントリレーとして動作するアプリケーション。
テンプレート (template)	(n.) XPath 式によって選択されたノードに適用されるフォーマット指示一式。
テンプレートエントリー (template entry)	「連携サーバー (cooperating server)」を参照してください。

タイムアウト (timeout)	(n.) サーバーが、ハングしたように見えるサービスルーチンの終了処理を中止するまでの指定された時間。
時間帯 (time zone)	(n.) 同じ時間帯を使用する地域。-12 から +12 まで(「GMT (グリニッジ標準時)」は0) の、25 種類の時間単位のタイムゾーンがあります。各タイムゾーンは、GMT を基準としています。ほとんどのタイムゾーンは、地域対応化された3文字の略語による記号で表示されます。Calendar Server では、America/Los_Angeles や Asia/Calcutta といったタイムゾーン ID (TZID) によってもタイムゾーンを識別します。
TLS	(Transport Layer Security) (n.) トランスポート層で暗号化と証明書を提供するプロトコル。クライアントおよびサーバーアプリケーションに対して大きな変更を加える必要なく、データをセキュリティー保護されたチャンネル経由で送受信することができます。「SSL」の標準であり、公開鍵ベースのプロトコルです。
todo	(n.) サーバー側において、行うべきことを指定するカレンダーコンポーネント。クライアント側の Calendar Express では、「 タスク (task) 」にあたります。
ツールプロバイダ (tool provider)	(n.) J2EE アプリケーションの開発、パッケージ、および配備に使用するツールを提供する組織またはソフトウェアベンダ。
top	(n.) UNIX のみ。システムリソースの現在の使用状態を表示する、UNIX システムプログラムの1つ。
トピック (topic)	(n.) 「 パブリッシュ/サブスクライブ配信モデル (publish/subscribe delivery model) 」を実装するために、管理者が作成するオブジェクト。トピックは、アドレス指定されたメッセージの収集および配信を担うコンテンツ階層のノードとして表示できます。中間段階としてトピックを使用することにより、メッセージパブリッシャーがメッセージサブスクライバから分離されます。
トップレベル管理者 (top-level administrator)	(n.) Delegated Administrator for Messaging and Collaboration の GUI または CLI を使用して、Messaging Server 名前空間全体のメールユーザー、メールリスト、ファミリアアカウント、およびドメインの作成、変更、および削除を行うための管理権限を持つユーザー。デフォルトでは、このユーザーがトポロジ内のすべてのメッセージサーバーに対するメッセージストア管理者となります。
トップレベルドメイン権限 (top-level domain authority)	(n.) ホスト名分類の最上位カテゴリ。通常は、ドメインが存在する組織のタイプ(たとえば、.com は企業、.edu は教育機関)またはドメインの元となる国(たとえば、.us はアメリカ合衆国、.jp は日本、.au はオーストラリア、.fi はフィンランド)を意味します。
トポロジ (topology)	(1) (n.) 物理サーバー間でディレクトリツリーを分割し、それらのサーバーどうしが相互にリンクする方法。

	(2)(n.) マシン、Application Server インスタンス、および HADB ノードの配列、およびこれらの間の通信フロー。「サーバーインスタンス (server instance)」、「HADB ノード (HADB node)」を参照してください。
トランザクション (transaction)	(1)(n.) グループとして成功または失敗する一連のデータベースコマンド。トランザクション全体が成功するには、そのトランザクションに関連するすべてのコマンドが正常に実行される必要があります。 (2)(n.) データを修飾する不可分な作業単位。トランザクションには1つ以上のプログラム文が含まれますが、それらすべてのプログラム文の末尾は「コミットする (commit)」または「ロールバック (rollback)」のどちらかです。トランザクションを使用することにより、複数のユーザーが同一のデータストアに並行してアクセスできます。
トランザクション属性 (transaction attribute)	(n.) エンタープライズ Bean のメソッドを呼び出すときに、EJB コンテナがトランザクション有効範囲を制御するために使用する「エンタープライズ Bean (enterprise bean)」の配備記述子に指定される値。トランザクション属性は次の値を取ることができます。Required、RequiresNew、Supports、NotSupported、Mandatory、または Never。
トランザクションコンテキスト (transaction context)	(n.) ローカルまたはグローバルなトランザクションの範囲。「トランザクションコンテキスト (transaction context)」を参照してください。
トランザクション遮断レベル (transaction isolation level)	(n.) トランザクションが並行する他のトランザクションとの間で、変更中のデータの中間状態を相互に可視にする度合い。
トランザクションマネージャ (transaction manager)	(n.) トランザクション境界、トランザクションリソース管理、同期化、およびトランザクションコンテキスト伝達のサポートに必要なサービスと管理機能を提供します。通常は「XA プロトコル (XA protocol)」を使用します。「グローバルトランザクション (global transaction)」も参照してください。
トランザクションリカバリ (transaction recovery)	(n.) 自動または手動による分散トランザクションのリカバリ。
トランジエンス (transience)	(n.) 使用されていないときにリソースを解放するプロトコル。「持続性 (persistence)」とは反対の概念です。
一時的な失敗 (transient failure)	(n.) メッセージ処理中に発生するエラー状態。リモート「MTA (メッセージ転送エージェント)」の場合、メッセージが配信された時点でそのメッセージを処理できなくても、あとで処理できることがあります。ローカル MTA は、メッセージをキューに戻し、あとで再転送されるようにメッセージをスケジュールします。

Transport Layer Security

(TLS) (n.) SSL の標準化形式。「[Secure Socket Layer](#)」も参照してください。

**トランスポート
プロトコル
(transport
protocols)**

(n.) SMTP や X.400 など、「[MTA \(メッセージ転送エージェント\)](#)」間でのメッセージ転送手段を提供するプロトコル。

**信頼データベース
(trust
database)**

(n.) 公開鍵と非公開鍵を保存したセキュリティーファイル。「[鍵ペアファイル \(key-pair file\)](#)」とも呼ばれます。

**信頼できるプロ
バイダ (trusted
provider)**

(n.) 「[トラストサークル \(circle of trust\)](#)」のサービスプロバイダおよびアイデンティティプロバイダのグループの1つ。ユーザーは、セキュリティー保護された環境で信頼できるプロバイダとトランザクションおよび通信を実行できます。

U

UAProf	(n.) Open Mobile Alliance によって定義された規格。モバイルデバイスは、自身の機能をネットワークサーバーに伝達できます。
UBE	「 大量の迷惑メール (unsolicited bulk email) 」を参照してください。
UDDI	(Universal Description, Discovery, and Integration) (n.) 検出および統合を行う Web サービスの世界的なレジストリ。インターネットを使用したサービスの記述、ビジネスの発見、ビジネスサービスの統合のための、プラットフォームに依存しないオープンなフレームワーク作りを主導する業界団体、およびそのレジストリ。ベンダーのコンソーシアムによって策定が行われている。
Unicode	(n.) ISO 10646 と Unicode Consortium によって定義された 16 ビットの文字セットで、数字の組み合わせを世界の諸言語の文字にマップする。16 ビットで 32,768 コードが網羅されるので、Unicode には世界のすべての言語を十分組み込むことができます。ただし、概念ごとに異なる文字を持つ中国語などの表意文字言語は例外です。Java プログラミングのすべてのソースコードは Unicode で作成されています。詳細については、 http://www.unicode.org/ を参照してください。
統一メッセージング (unified messaging)	(n.) 電子メール、ボイスメール、FAX、およびその他の通信形態に関して単一のメッセージストアを使用するという概念。Java Enterprise System Messaging Server では、完全な統一メッセージングソリューションの基盤を提供します。
Uniform Resource Indicator	「 URI 」を参照してください。
アンインストール (uninstallation) universal principal name	(n.) ソフトウェアコンポーネント全体を削除するプロセス。 (n.) ログインユーザーを表す値。ユーザーのログイン名と所属ドメインで構成されます。たとえば、ドメイン example.com に属するユーザー bill の Universal Principal Name は、bill@example.com です。UPN とも呼ばれます。

Universal Standard Products and Services Classification (UNSPSC)

(n.) 商品を分類および識別する「スキーマ」。販売側と購入側の目録に使用され、支出分析の標準化された勘定科目コードとして使われている。

解析対象外エンティティ (unparsed entity)

(n.) XML 以外の内容を含む汎用エンティティ。解析対象外エンティティには、その性質上、バイナリデータが含まれます。

大量の迷惑メール (unsolicited bulk email)

(n.) 一般に宣伝目的でメール送信業者から大量に送信される迷惑メール。スパムとも呼ばれます。

上位参照 (upper reference)

(n.) 「DIT」内で、ディレクトリサーバーのネーミングコンテキストの上位にあるネーミングコンテキストを保持するディレクトリサーバーを示します。

URI

(uniform resource identifier) (n.) 抽象リソースまたは物理リソースの大域的に一意な識別子。URL は URI の一種で、検索プロトコル (Web アプリケーションの場合 http または https) とリソースの物理的な場所 (ホスト名およびホスト相対パス) を指定します。

URL データベース修復 (URL database repair)

(n.) ソフトウェアの障害、システムのクラッシュ、ディスクの動作停止、または満杯のファイルシステムによって損傷した URL データベースを修復および更新するプロセス。

URL マッピング (URL mapping)

(n.) ドキュメントディレクトリの物理パス名をユーザー定義エイリアスにマッピングするプロセス。これにより、ディレクトリ内部のファイルが、ファイルの完全物理パス名の代わりにディレクトリのエイリアスを参照するだけで済むようになります。ファイルを `usr/JES/servers/docs/index.html` として指定する代わりに、`/myDocs/index.html` として指定できます。このマッピングにより、サーバーファイルの物理的な位置をユーザーが知る必要がなくなるので、サーバーのセキュリティが高まります。

URL パス (URL path)

(n.) サブレットを呼び出すために HTTP 要求によって渡される URL の部分。URL パスは、次のようにコンテキストパス、サブレットパス、およびパス情報で構成されます。

- コンテキストパスは、サブレットが含まれるサブレットコンテキストに関連付けられたパス接頭辞です。このコンテキストが Web サーバーの URL 名前空間をルートに持つデフォルトコンテキストである場合、パス接頭辞は空文字列になります。それ以外の場合、パス接頭辞は / 文字で始まりますが、/ 文字では終わりません。
- サブレットパスは、この要求を有効にしたマッピングに直接対応するパスセクションです。このパスは / 文字で始まります。

- パス情報は、要求パスのうち、コンテキストパスとサブレットパス以外の部分です。

URL プール (URL pool)	(n.) ロボットが処理する URL のリスト。ロボットの起動時に URL プールは開始位置によって構成されていますが、列挙処理時に検出されたすべてのリソースによって瞬時に増大します。
URN	(uniform resource name) (n.) エンティティは識別しても、エンティティのある場所は明示しない一意の識別子。URN を使用することで、システムはエンティティを Web 上で検索する前にローカルで検索できます。Web 上の場所を変更しても、引き続きエンティティが検出されるようにすることが可能です。
ユースケース (use case)	(n.) 「分散エンタープライズアプリケーション (distributed enterprise application)」によって実行され、アプリケーションのパフォーマンスの設計、テスト、および測定的基础として使用される、エンドユーザータスクまたは一連のタスク。
ユーザー	(1) (n.) アプリケーションを使用する人またはサービス。プログラマ的には、アプリケーションがユーザーを識別する際の手掛かりとなるユーザー名、パスワード、および一連の属性で構成されています。 (2) (n.) 認証された個人 (またはアプリケーションプログラム) のアイデンティティ。ユーザーはそのアイデンティティに関連付けられたロールのセットを持つことができます。このセットはロールが保護するすべてのリソースにアクセスする権利をユーザーに付与します。「主体 (principal)」、「グループ (group)」および「ロール (role)」も参照してください。
ユーザーアカウント (user account)	(n.) サーバーにアクセスするためのアカウント。ディレクトリサーバー上のエントリとして管理されます。
userAgent	(n.) Portal Server Mobile Access で使用する、HTTP ユーザーエージェントヘッダーを参照するプロパティ。ユーザーエージェントヘッダーは、特定のモバイルデバイスに固有の値になることが多く、クライアントタイプのデータを検出および取得する場合に使用できます。
ユーザーエージェント (user agent)	(n.) ユーザーがメールメッセージを作成、送信、受信できるようにする、Netscape™ Communicator などのクライアントコンポーネント。UA とも呼ばれます。
ユーザーデータ制約 (user data constraint)	(n.) クライアントと Web コンテナの間のデータを保護する方法を指定します。この保護は、データ改ざんの防止またはデータ盗聴の防止です。
ユーザーエントリ (user entry)	(n.) 各ユーザーに関する必須および任意の情報を記述するフィールド。たとえば、識別名、氏名、役職、電話番号、ポケベルの番号、ログイン名、パスワード、ホームディレクトリなどがあります。ユーザープロフィールとも呼ばれます。

ユーザーフォルダ (user folders)	(n.) ユーザーの電子メールのメールボックス。
ユーザーグループ (user group)	(n.) 接続や送信先などの Message Queue メッセージサーバーのリソースへのアクセスを承認するために、Message Queue クライアントのユーザーが属するグループ。
User/Groups Directory Server	(n.) 組織内のユーザーおよびグループに関する情報を保持する Directory Server。
ユーザー制限容量 (user quota)	(n.) システム管理者により、電子メールメッセージ用にユーザーに割り当てられるスペース容量。
ユーザープロビジョニング (user provisioning)	(n.) エンドユーザーがサービスを使用できるようにするプロセス、またはエンドユーザーにサービスへのアクセスを提供するプロセス。プロビジョニングには、アイデンティティ、ポリシー、およびユーザーアカウント管理アクティビティがあります。たとえば、エンドユーザーごとにアカウントをディレクトリに作成することや、アカウントにさまざまなサービスで必要になるユーザー固有の情報を入力することなどが含まれます。
ユーザーセッション (user session)	(n.) サーバーによって追跡される、ユーザーとアプリケーションの間の一連の対話。セッションでは、ユーザーの状態、持続オブジェクト、および ID 認証が管理されません。

V

- 有効な (**valid**) (adj.)有効な「[XML \(拡張可能マークアップ言語\)](#)」ドキュメントは、整形式であるだけでなく、「[DTD \(ドキュメントタイプ定義\)](#)」によって課されるすべての制約に適合します。また、DTDによって許可されていないタグを含まず、タグの順序はDTDの仕様に適合します。
- 検証パーサー (**validating parser**) (n.)「[XML \(拡張可能マークアップ言語\)](#)」ドキュメントが、整形式であるだけでなく、有効であることを確認する「[パーサー \(parser\)](#)」。
- 値バインド式 (**value-binding expression**) (n.)バックイング Bean のプロパティを参照する「[JavaServer Faces 式言語 \(JavaServer Faces expression language\)](#)」式。コンポーネントタグはこの式を使用して、関連するコンポーネントの値またはコンポーネントインスタンスを Bean プロパティにバインドします。コンポーネントタグがコンポーネントの値属性によってプロパティを参照する場合は、コンポーネントの値がプロパティにバインドされます。コンポーネントタグがコンポーネントのバインド属性によってプロパティを参照する場合は、コンポーネントそのものがプロパティにバインドされます。
- バニティードメイン (**vanity domain**) (n.)特定のサーバーまたはホストドメインではなく、個別のユーザーに関連付けられているドメイン名。バニティードメインは、`MailAlternateAddress` 属性を使用して指定されます。バニティードメインのドメイン名には「[LDAP \(Lightweight Directory Access Protocol\)](#)」エントリがありません。バニティードメインは、個人または小さな組織が、独自のホストドメインをサポートするための管理負荷をかけずに、カスタマイズしたドメイン名を使用する場合に便利です。カスタムドメインとも呼ばれます。
- /var/mail** (n.)新しいメールメッセージを順番に単一のフラットテキストファイル内に格納する Berkeley 方式の受信箱を示すために使用される名前。
- バージョン管理 (**versioning**) 「[動的再読み込み \(dynamic reloading\)](#)」を参照してください。

仮想データビュー (virtual data view)	(n.) JDBC データソース、LDIF データソース、または複数の集約データソースの LDAP 表現。仮想データビューは、実際には、特定の变换アクションが定義された通常の Directory Proxy Server データビューです。
仮想ドメイン (virtual domain)	(1) (n.) ISP ホストドメイン。 (2) (n.) Messaging Multiplexor によってクライアントのユーザー ID に追加される、LDAP 検索やメールボックスサーバーへのログインのためのドメイン名。「ドメイン (domain)」も参照してください。
仮想ホスト (virtual host)	(n.) 単一の IP アドレスにマップされる複数のホストおよびドメイン名。
仮想リスト表示インデックス (virtual list view index)	(n.) ユーザーインタフェースを備えるクライアントが仮想リスト表示拡張を使用する場合に、ディレクトリサーバーコンソール(またはほかのグラフィカルユーザーインタフェース)のエントリの表示速度を高めるフィルタ方法。仮想リスト表示インデックスは、ディレクトリツリーの任意の分岐で作成し、特定の検索処理での表示速度を改善できます。ブラウズインデックスとも呼ばれます。
仮想プライベートネットワーク (virtual private network)	(n.) 外見や機能は標準のネットワークでも、公共のネットワーク内のプライベートネットワークに似ているネットワーク。下位のプロトコル層で暗号化を使用することにより、インターネットのように通常は安全が確保されていないネットワークでも、セキュリティが保護された接続を提供します。一般に VPN は、専用線を使用するプライベートネットワークよりもコストを抑えることができます。VPN は、接続の両端で同じ暗号化方式を使用することが前提です。暗号化処理は、ファイアウォールソフトウェアかルーターによって実行されます。
仮想サーバー (virtual server)	(1) (n.) 特定の URL のターゲットとなるコンテンツを提供する仮想 Web サーバー。複数の仮想サーバーが、同一または異なったホスト名、ポート番号、IP アドレスなどを使ってコンテンツを提供できます。HTTP サービスは、URL に従って、受信する Web 要求を複数の異なる仮想サーバーに送信できます。仮想ホストとも呼ばれます。 (2) (n.) 仮想サーバーは、1つのインストールサーバーで、複数のドメイン名、IP アドレス、およびサーバー監視機能を設定するための方法です。
仮想サーバークラス (virtual server class)	(n.) obj.conf ファイルの同じ基本設定情報を共有する仮想サーバーの集合。
仮想変換 (virtual transformation)	(n.) Directory Proxy Server 仮想データビューに物理データが表示される方法を判定する定義。仮想変換は、データの別のビューを取得するために、データビューに対して定義されます。
音声ポータルデスクトップ (voice Portal Desktop)	(n.) 電話または同様のデバイスによって提供される、Portal Server サイトの音声プレゼンテーション。

voiceXML	(n.) 対話式の音声応答アプリケーションで使用する音声ダイアログを作成するためのマークアップ言語。
VoIP	(voice over IP) (n.) IP ネットワークで音声テレフォニを実現するテクノロジー。
ボリュームマネージャー (volume manager)	(n.) ディスクのストライプ化、連結、ミラー化、メタデバイスやボリュームの動的成長によってデータの信頼性を提供するソフトウェア製品。
VPN	「 仮想プライベートネットワーク (virtual private network) 」を参照してください。
VPN ゲートウェイ (VPN gateway)	(n.) VPN へのエントリポイント。通常はファイアウォールによって保護されています。
VRFY	(n.) ユーザー名を確認するための SMTP コマンド。RFC 821 に定義されています。

W

- W3C** (World Wide Web Consortium) (n.) インターネット標準を統括する国際的な機関。Web サイトは <http://www.w3.org/> です。
- WAP (ワイヤレスアプリケーションプロトコル)** (WAP (Wireless Application Protocol)) (n.) ワイヤレス通信によってアプリケーションを実行するオープン標準。
- WAR ファイル (WAR file)** 「[Web アプリケーションアーカイブ \(web application archive\)](#)」を参照してください。
- 警告 (warning)** (n.) ドキュメントの「[DTD \(ドキュメントタイプ定義\)](#)」に重複する定義が含まれている場合、および必ずしもエラーではないものの、エラーになる可能性があることをドキュメント作成者が知っていた方がよい場合に、SAX パーサー警告が生成されます。「[致命的エラー \(fatal error\)](#)」も参照してください。
- WCAP** (Web Calendar Access Protocol) (n.) クライアントが Calendar Server との通信に使用する、高レベルのコマンドベースプロトコル。
- Web アプリケーション (web application)** (n.) サーブレット、「[JSP テクノロジ \(JSP technology\)](#)」によって作成されたページ、HTML ドキュメント、およびその他の Web リソースの集まり。Web リソースには、イメージファイル、圧縮アーカイブ、その他のデータが含まれる場合もあります。Web アプリケーションは、Web アーカイブ (WAR ファイル) にパッケージ化されている場合や、オープンディレクトリ構造に配備されている場合があります。Java Enterprise System Application Server では、「[SHTML](#)」や「[CGI](#)」など一部の Java 以外の Web アプリケーションテクノロジーもサポートしています。
- Web アプリケーションアーカイブ (web application archive)** (n.) Web アプリケーション全体を圧縮された形式で保持するアーカイブファイル。Java Enterprise System Web Server は、WAR ファイルのアプリケーションにアクセスできません。Java Enterprise System Web Server でサービスを提供するには、Web アプリケーションの圧縮を解除し、wdeploy ユーティリティを使用して配備する必要があります。

Web アプリケーション、distributable (web application, distributable)	(n.) J2EE テクノロジを使用する Web アプリケーションで、同一または異なるホスト上で稼働する複数の Java 仮想マシン間で分散される Web コンテナに配備できるように作成されたもの。このようなアプリケーションの配備記述子は distributable 要素を使用します。
Web キャッシュ (web cache)	(n.) Java Enterprise System Application Server の機能の 1 つ。パフォーマンスを向上させるために、サーブレットまたは「 JSP テクノロジ (JSP technology) 」によって作成されたページに、その結果を特定の時間キャッシュできるようにします。その時間内にサーブレットまたは JSP ページを呼び出すと、キャッシュに保存された結果が返されるので、サーブレットまたは JSP ページを実行し直す必要はありません。
Web コンポーネント (web component)	(n.) 要求に回答してサービスを提供するコンポーネント。「 サーブレット (servlet) 」または「 JSP ページ (JSP page) 」。
Web コネクタプラグイン (web connector plug-in)	(n.) Web サーバーが Java Enterprise System Application Server と通信できるようにする、Web サーバーの拡張機能。
Web コンテナ (web container)	(n.) J2EE アーキテクチャーの Web コンポーネント規約を実装する「 コンテナ (container) 」。この規約は、セキュリティ、並行性、ライフサイクル管理、トランザクション、配備、およびその他のサービスなど、Web コンポーネントの実行時環境を指定します。Web コンテナは JSP コンテナと同じサービスおよび J2EE プラットフォーム API の統一されたビューを提供します。Web コンテナは Web サーバーまたは J2EE サーバーによって提供されます。
Web コンテナ、distributed (web container, distributed)	(n.) distributable としてタグ付けされ、同一または異なるホスト上で稼働する複数の Java 仮想マシンの間で動作する Web アプリケーションを実行可能な Web コンテナ。
Web コンテナプロバイダ (web container provider)	(n.) Web コンテナを提供するベンダー。
Web メール (webmail)	(n.) ブラウザベースの電子メールサービスを示す一般的な用語。ブラウザベースのクライアントは、多くの処理をサーバーに任せるので、「 シンクライアント 」とも呼ばれ、常にサーバー上に格納されるメールにアクセスします。「 Messenger Express 」も参照してください。
Web モジュール (web module)	(n.) J2EE アプリケーション内ではなく、個別に配備された Web アプリケーション。「 Web アプリケーション (web application) 」を参照してください。
Web リソース (web resource)	(n.) URL で参照可能な Web アプリケーションに含まれる静的または動的オブジェクト。

Web リソースコレクション (web resource collection)	(n.) 保護対象の Web リソースのセットを記述する URL パターンおよび HTTP メソッドのリスト。
Web Server	(n.) Portal Server の Web サーバー。Portal Server および Portal Server パック Web アプリケーションの Web コンテナとして使用されます。Sun Java System Web Server は、Directory Server Access Management Edition 製品に組み込まれています。
Web サーバー (web server)	(n) インターネット、イントラネット、またはエキストラネットにアクセスするサービスを提供し、Web アプリケーションを格納および管理するホスト。完全な J2EE アプリケーションではありません。Web サーバーは Web サイトを管理し、HTTP およびその他のプロトコルのサポートを提供し、特定の機能を実行するためのサーバー側のプログラム (CGI スクリプトやサーブレットなど) を実行する。J2EE アーキテクチャーでは、Web サーバーは Web コンテナへのサービスを提供します。たとえば、Web コンテナは、通常、Web サーバーを使用して HTTP メッセージを処理します。J2EE アーキテクチャーでは、Web コンテナは同じベンダーの Web サーバーによって処理されることを前提としているので、この 2 つのエンティティー間の規約は指定されていません。1 つの Web サーバーで 1 つまたは複数の Web コンテナをホストすることができます。
Web サーバープラグイン (web server plug-in)	(n.) Java Enterprise System Web Server または Java Enterprise System Application Server に指示して特定の HTTP 要求を別のサーバーに転送することができるようにする、HTTP リバースプロキシプラグイン。
Web サーバープロバイダ (web server provider)	(n.) Web サーバーを提供するベンダー。
Web サービス (web service)	(1) (n.) アクセス可能性、サービスのカプセル化、および検出に関する標準インターネットプロトコルに準拠しているサービス。標準には、SOAP (Simple Object Access Protocol) メッセージングプロトコル、WSDL (Web Service Definition Language) インタフェース定義、UDDI (Universal Discovery, Description, and Integration) レジストリ標準などがあります。Web サービスは要求を受け入れ、要求に基づいてその機能を実行し、応答を返します。要求と応答を同じ操作の一部にすることも、またはそれらが別々に発生するようにすることもできます。後者の場合は、コンシューマが応答を待つ必要はありません。通常、要求と応答は共に XML の形式 (ポータブルデータ交換形式) を取り、HTTP などのワイヤプロトコルで配信されます。 (2) (n.) Web 経由で提供されるサービス。インターネットまたはイントラネットを経由してシステムからの要求を受け入れ、それを処理し、応答を返す、自己完結型の自己記述式モジュラーアプリケーションです。
Web サービスコンシューマ (web service consumer)	(n.) Web サービスコンシューマは、Web サービスプロバイダに要求することにより、Web サービスが提供する操作を呼び出します。

Web サービスプロバイダ (web service provider)

(n.) Web サービスプロバイダは、Web サービスコンシューマからの要求に基づいて、Web サービスを実装します。Web サービスコンシューマが使用するのと同じ Java™ 仮想マシンで実行されることもあります。

整形式の (well-formed)

(adj.) 構文的に正しい XML ドキュメント。構文的に正しいドキュメントは、タグの一部ではない山括弧が含まれておらず、すべてのタグが終了タグを持っているかそれ自身が自己終了タグであり、すべてのタグが完全に入れ子になっています。整形式であることが分かっているドキュメントは処理をすることが可能です。ただし、整形式のドキュメントでも有効でない場合があります。有効かどうかを判別するには、「**検証パーサー (validating parser)**」と「**DTD (ドキュメントタイプ定義)**」が必要です。

Windows CGI

(n.) Windows NT のみ。Visual Basic などの Windows ベースプログラミング言語で記述された「**CGI**」プログラム。

ワイヤレスデスクトップディスパッチャー (wireless desktop dispatcher)

(n.) ユーザー要求を、どのポータルデスクトップ、モバイルポータルデスクトップ、または音声ポータルデスクトップに経路指定するかを決定するコンポーネント。

取り消しパッチ (withdrawn patch)
WML (ワイヤレスマークアップ言語)

(n.) 配信システムから削除されたパッチ。

(WML (wireless markup language)) (n.) WAP の一部である XML に基づくマークアップ言語。

ワークグループ (workgroup)

(n.) ローカルワークグループ環境。サーバーは、ローカルオフィスまたはワークグループ内で、独自の経路指定および配信を実行します。部門間のメールは、バックボーンサーバーにルーティングされます。「**バックボーン (backbone)**」も参照してください。

WS-BPEL (Web サービスプロセス実行言語)

(WS-BPEL (web services business process execution language)) (n.) BPEL の変異形で、Web サービス記述言語 (**WSDL (Web サービス記述言語)**) から派生した構文を使用します。WS-BPEL は、これらの構文を使用して、インバウンドおよびアウトバウンドのプロセスインタフェースを記述し、プロセスを他のプロセスやアプリケーションに容易に統合できるようにしています。

WSDL (Web サービス記述言語)

(WSDL (web service description language)) (n.) 標準化された方法で Web サービスを定義するために使用される、XML ベースの言語。Web サービスの 3 つの基本的なプロパティ (Web サービスの定義、Web サービスにアクセスする方法、および Web サービスの位置) を記述します。

X ~ Z

- X.400** (名詞) メッセージ処理システムの標準。
- X.500 標準 (X.500 standard)** (名詞) Directory Server 実装で使用する推奨情報モデル、オブジェクトクラス、および属性の概要を述べた一連の ISO/ITU-T ドキュメント。「[LDAP \(Lightweight Directory Access Protocol\)](#)」は、X.500 標準で使用される Directory Access Protocol (DAP) の軽量バージョンです。
- Xalan** (名詞) [XSLT \(拡張可能スタイル言語変換\)](#) の処理系バージョン。
- XA プロトコル (XA protocol)** (名詞) 分散トランザクション対応のデータベース業界標準プロトコル。
- XHTML (拡張可能ハイパーテキストマークアップ言語)** (XHTML (extensible hypertext markup language)) (n.) 新規要素と属性を追加することによって拡張可能な HTML 4.0 の再編版。数種類ある XHTML DTD のいずれかによって定義され、HTML の外観はあたかも XML であるかようになります。XHTML を使ってすべてを表記しようとする、XML の目的が損なわれてしまいます。XML は情報の内容を識別することを意図したものであって、表示する方法を指定するだけではないからです。DTD 内で XHTML を参照することもできます。それによって要素内のテキストをプレーンテキストに限定せずに `` と `` のタグを含めるといった指定なども可能になります。
- XLink** (名詞) XLL 仕様の中の、ドキュメント間のリンクの指定に関係する部分。
- XLL** (名詞) XML リンク言語 (XML Link Language) 仕様。XLink と XPointer とで構成されます。
- XML (拡張可能マークアップ言語)** (XML (extensible markup language)) (n.) WWW コンソーシアム ([W3C](#)) によって開発された柔軟なプログラミング言語。これにより、共通の情報形式を作成でき、Web、イントラネット、その他のあらゆる場所で、形式およびデータを共有できます。このマークアップ言語を使用することによって、XML ドキュメント内のコンテンツ、データ、およびテキストの識別に必要なタグ (マークアップ) を定義することが可能

になります。この言語はHTMLとは異なります。HTMLはインターネット上で情報を表示するために使われることの多いマークアップ言語で、主にスタイルや表示形式を指定する既定のタグが備わっています。XMLドキュメントは、スタイルシートの制御下でスタイルタグを含む言語に一度変換されてはじめて、ブラウザや他のプレゼンテーションメカニズムで表示できるようになります。XMLで使用される2つのタイプのスタイルシートは、「CSS」と「XSL(拡張可能スタイル言語)」です。一般に表示用には、XMLからHTMLへと変換されます。XMLドキュメントを生成する場合に必要な応じてタグを定義することは可能ですが、ドキュメントタイプ定義(DTD)を使って特定のドキュメントタイプに使用可能な要素を定義できます。DTDのルールとドキュメントを比較し、ドキュメントの有効性を判定したり、特定の要素がドキュメント内のどこにあるかを確認することができます。WebサービスアプリケーションのJ2EE 配備記述子は、XMLでは、許可された要素を定義するスキーマを使って表現されます。XMLドキュメントを処理するプログラムはSAX APIまたはDOM APIを使用します。Calendar Serverは、XMLとXSLを使用してCalendar Express ユーザーインターフェースを生成します。

**XML 名前空間
(XML namespace)**

(名詞) DTD (ドキュメントタイプ定義) で定義される一連の要素名に一意のラベルを指定することができる標準。DTD を使用するドキュメントは、要素名どうしの競合を起こさずに、ほかのドキュメントに組み込むことができます。DTD 内で定義されている要素は一意に識別されるので、たとえばパーサーは、別のドキュメントタイプ定義ではなく、使用する DTD に従って要素に割り込むタイミングを決定できます。

**XML レジストリ
(XML registry)**

「レジストリ (registry)」を参照してください。

XML スキーマ

(名詞) XML ドキュメントの構造、コンテンツ、およびセマンティクスを定義する W3C 仕様。

XPath

(名詞) XML ドキュメントの特定の箇所を識別するアドレス指定メカニズム。

XPointer

(名詞) XLL 仕様の中のドキュメントのセクションの識別に関係する部分で、セクションをリンクで参照したり、他のドキュメントに組み込めるようにします。

XSL(拡張可能スタイル言語)

(XSL (extensible style language)) (n.) HTML で使用されるカスケード型スタイルシート (CSS) に似た、XML のスタイルシートを作成するために使用する言語。XML では、コンテンツとプレゼンテーションが分離されています。XML タグは、表示方法を指示しません。XML ドキュメントは、読み取る前に書式設定する必要があります。XSL 標準を使用することにより、次の操作が可能になります。

- アドレス指定メカニズムの指定。変換を適用する XML ドキュメントの対象箇所を識別できます。(XPath)
- タグ変換の指定。XML データを異なるフォーマットに変換できます。(XSLT)

-
- ページサイズ、マージン、フォントの高さと幅、各ページのフローオブジェクトなどの表示特性の指定。情報は、ページ内のある一区画に埋め込まれ、その区画がいっぱいになると、自動的に次のオブジェクトに流し込まれます。これにより、画像の周囲でテキストを折り返したり、記事を別のページに続けることができます。(XSL-FO)

XSL-FO

(名詞) フォントサイズ、ページレイアウト、および複数ページにまたがって情報を流し込む方法を記述するために使用する XSL のサブコンポーネント。

**XSLT (拡張可能
スタイル言語変
換)**

(XSLT (extensible style language transformation)) (n.) XML スタイルシートによって使用され、1つの形式の XML ドキュメントを別の XML 形式に変換する言語。この変換は、多くのプラットフォームとさまざまな XML ドキュメントコーディング全体で共通のサービスを提供するので、電子商取引と e ビジネスにおいて非常に便利です。ターゲットドキュメントには、ブラウザや他のプレゼンテーションメカニズムでレンダリングされる方法を指定するプレゼンテーション関連のタグが含まれることがあります。XSLT は以前は XSL の一部であり、スタイルフローオブジェクトのタグ言語も含まれていました。

XSLTC

(名詞) **XSLT (拡張可能スタイル言語変換)** のコンパイル系バージョン。

**Zulu 時間 (Zulu
time)**

(名詞) 「**GMT (グリニッジ標準時)**」と UTC (協定世界時) を指す軍用語。

