



Sun Java™ System

Sun Java Enterprise System

用語集

---

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

Part No: 819-1933

Copyright © 2005 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. All rights reserved.

Sun Microsystems, Inc. は、この製品に含まれるテクノロジーに関する知的所有権を保持しています。特に限定されることなく、これらの知的所有権は <http://www.sun.com/patents> に記載されている 1 つ以上の米国特許および米国およびその他の国における 1 つ以上の追加特許または特許出願中のものが含まれている場合があります。

このソフトウェアは SUN MICROSYSTEMS, INC. の機密情報と企業秘密を含んでいます。SUN MICROSYSTEMS, INC. の書面による許諾を受けることなく、このソフトウェアを使用、開示、複製することは禁じられています。

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.

この配布には、第三者が開発したソフトウェアが含まれている可能性があります。

本製品の一部は、カリフォルニア大学からライセンスされている Berkeley BSD システムに基づいていることがあります。UNIX は、X/Open Company, Ltd が独占的にライセンスしている米国およびその他の国における登録商標です。

Sun、Sun Microsystems、Sun のロゴマーク、Java、Solaris、JDK、Java Naming and Directory Interface、JavaMail、JavaHelp、J2SE、iPlanet、Duke のロゴマーク、Java Coffee Cup のロゴ、Solaris のロゴ、SunTone 認定ロゴマークおよび Sun ONE ロゴマークは、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。

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に基づくものです。

Legato および Legato のロゴマークは Legato Systems, Inc. の商標であり、Legato NetWorker は同社の商標または登録商標です。

Netscape Communications Corp のロゴマークは Netscape Communications Corporation の商標または登録商標です。

OPEN LOOK および Sun Graphical User Interface は、米国 Sun Microsystems 社が自社のユーザ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インタ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、このライセンスは米国 Sun Microsystems 社のライセンス実施権者にも適用されます。

この製品は、米国の輸出規制に関する法規の適用および管理下にあり、また、米国以外の国の輸出および輸入規制に関する法規の制限を受ける場合があります。核、ミサイル、生物化学兵器もしくは原子力船に関連した使用またはかかる使用者への提供は、直接的にも間接的にも、禁止されています。このソフトウェアを、米国の輸出禁止国へ輸出または再輸出すること、および米国輸出制限対象リスト (輸出が禁止されている個人リスト、特別に指定された国籍者リストを含む) に指定された、法人、または団体に輸出または再輸出することは一切禁止されています。

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ぬものとします。

# Sun Java Enterprise System 用語集

この用語集では、Java Enterprise System で使用される用語を定義します。品詞については、形容詞を「adj.」、名詞を「n.」、動詞を「v.」で示します。

**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) (MX record (mail exchange record))」も参照してください。

**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。

**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。

**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。

**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。

**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。

**API (アプリケーションプログラミングインタフェース) (API (application programming interface))** (1) (n.) コンピュータプログラムが、API を解釈するように設計されたほかのソフトウェアまたはハードウェアと通信するために使用できる命令の集まり。  
(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。

**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ったりできるようにします。

**Bean 管理による持続性 (bean-managed persistence)** (n.) エンティティ Bean の変数とデータストアの間で行われるデータ転送。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.) エンタープライズ Bean のトランザクション境界設定の制御が、プログラムを使用して開発者により行われません。「[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と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。

**BLOB (バイナリラージオブジェクト) (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。

**CA ( 認証局) (CA (certificate authority))** (1) (n.) 暗号化トランザクションで使用されるデジタルファイルを発行する、内部またはサードパーティの組織。  
(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 \(公開鍵インフラストラクチャ\) \(PKI \(public key infrastructure\)\)](#)」も参照してください。

**Calendar Express** (n.) エンドユーザーが Calendar Server にアクセスするとき使用できる Web ベースのカレンダークライアントプログラム。

**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 \(Calendar Server アプリケーションプログラミングインタフェース\) \(CSAPI \(Calendar Server application programming interface\)\)](#)」を参照してください。

**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。

**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。

**CCR ( クラスタ設定リポジトリ) (CCR (cluster configuration repository))** (n.) Sun™ Cluster ソフトウェアがクラスタ設定情報を持続的に保存するために使用する、レプリケートされた高可用性データストア。

**CGI (共通ゲートウェイインタフェース) (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 を使用するために記述されたプログラムは、CGI プログラムまたは CGI スクリプトと呼ばれます。CGI プログラムは、サーバーが通常は処理できないフォームの処理や、通常は解析できない出力の解析を行います。

**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。

**chTML** (n.) モバイルデバイスに適した HTML の簡易版。

**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ければ、誰もその情報を読み取ることができません。

**CKL (危険化鍵リスト) (CKL (compromised key list))** (n.) 危険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。「**CRL (証明書失効リスト) (CRL (certificate revocation list))**」も参照してください。

**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。

**CLI (コマンド行インタフェース) (CLI (command-line interface))** (n.) ユーザープロンプトで実行可能な命令を入力できるインタフェース。「**管理インタフェース (administration interface)**」も参照してください。

**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。

**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。

**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。

**CMM (クラスタメンバーシップモニター) (CMM (cluster membership monitor))** (n.) 一貫性のあるクラスタメンバーシップの名簿を保持するソフトウェア。残りのクラスタリングソフトウェアは、このメンバーシップ情報を使用して、高可用性サービスの配置場所を決定します。CMM は、非クラスタメンバーがデータを壊したり、壊れたデータや整合性のないデータをクライアントに転送したりしないようにします。

**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。

**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。

**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。

**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。

**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を脅かすことはありません。

**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) (CORBA (common object request broker architecture))** (n.) オブジェクト指向分散コンピューティングの標準アーキテクチャ定義。

**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 (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。CoS 定義エントリは、LDAP サブエントリとして、影響を与える分岐の下に保存されます。

**CoS テンプレートエントリ (CoS template entry)** (n.) 共有属性値のリストが記述されたエントリ。

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

**COSNaming サービス (COSNaming Service)** (n.) IIOP ベースのネームサービス。

**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。

**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。

**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 \(危険化鍵リスト\) \(CKL \(compromised key list\)\)](#)」も参照してください。

**cronjob** (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。「[crontab ファイル \(crontab file\)](#)」も参照してください。

**crontab ファイル (crontab file)** (n.) UNIX のみ。指定した時間に自動的に実行されるコマンドのリストです。1 行に 1 つずつ記述されています。

**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。

**CUA (カレンダーユーザエージェント) (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。

**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。

**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 \(ディレクトリ情報ツリー\) \(DIT \(directory information tree\)\)](#)」。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。

**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フェース (GUI とユーティリティ)。

**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。



**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。

**DHCP (動的ホスト設定プロトコル) (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。

**DID** 「**デバイス ID (device ID)**」を参照してください。

**DID ドライバ (DID driver)** (n.) Java Enterprise System Messaging Server ソフトウェアによって実装されるデバイス ID ドライバ。クラスタ全体で一貫した名前空間を実現します。「**DID 名 (DID name)**」も参照してください。

**DID 名 (DID name)** (n.) SunPlex™ システム内のグローバルデバイスを識別するデバイス ID 名。DID 名は、Solaris™ オペレーティングシステムの論理名と、1 対 1 または 1 対多の関係にあるクラスタリング識別子です。Solaris の論理名の形式は dXsY で、X は整数、Y はスライス名になります。「**Solaris™ 論理名 (Solaris™ logical name)**」も参照してください。

**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。

**DIT (ディレクトリ情報ツリー) (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、ツリーのルート階層の最上位に表示します。

**DN (識別名) (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 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 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。

**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。「[ドキュメントタイプ定義](#)」も参照してください。

**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。

**Domain Administration Server** (n.) Domain Administration Server は、Sun Java System 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。

**DRU** 「[データ冗長ユニット \(DRU\) \(data redundancy unit \(DRU\)\)](#)」を参照してください。

**DSA (ディレクトリシステムエージェント) (DSA (directory system agent))** (n.) Directory Server での X.500 用語。

**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。

**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ー。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。

**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 cvarid が必要です。

**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 ファイルのクラスの構造とプロパティを記述したものの。

**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。

**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取めたアーカイブファイル。EAR ファイルの拡張子は .ear です。「**JAR ファイル (Java™ アーカイブファイル) (JAR file (Java™ Archive file))**」も参照してください。

**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.) パッケージ化されたエンタープライズアプリケーション、トランザクションシステム、またはユーザーアプリケーションと言い換えることができます。EIS の例には、R/3、PeopleSoft、Tuxedo、CICS などがあります。

**EJB コンテナ (EJB container)** 「**コンテナ (container)**」を参照してください。

**ejbc ユーティリティ (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。

**EJB™ QL (EJB™ Query Language)** (n.) コンテナ管理の関係によって定義されるエンティティ Bean がネットワーク上を移動するためのクエリ言語。

**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 Bean は、アプリケーションのビジネスロジックをカプセル化するサーバー側のコンポーネントです。ビジネスロジックは、アプリケーションの目的を果たすコードです。たとえば、在庫管理アプリケーションでは、エンタープライズ Bean はビジネスロジックを `checkInventoryLevel` や `orderProduct` というメソッドに実装します。これらのメソッドを呼び出すことで、リモートクライアントはアプリケーションが提供する在庫サービスにアクセスできます。「**コンテナ (container)**」、「**エンティティ Bean (entity bean)**」、「**メッセージ駆動型 Bean (message-driven bean)**」、および「**セッション Bean (session bean)**」も参照してください。

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

**ERP (エンタープライズリソース計画) (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。

**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。

**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。

**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。

**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。

**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。

**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。

**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。

**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。

**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。

**GUI** (n.) グラフィカルユーザーインターフェース。

**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) (high availability database (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) (high availability database (HADB))」、「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) (data redundancy unit (DRU))」も参照してください。  
(n.) エンタープライズ 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。

**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。

**HTML (ハイパーテキストマークアップ言語) (HTML (hypertext markup language))** (n.) Web ブラウザに表示できるドキュメントを作成するためのコーディングマークアップ言語。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。

**HTML ページ (HTML page)** (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。

**HTTP サーブレット (HTTP servlet)** (n.) javax.servlet.HttpServlet を拡張するサーブレット。HTTP サーブレットには、HTTP プロトコルのサポートが組み込まれています。「汎用サーブレット (generic servlet)」も参照してください。

**HTTP (ハイパーテキスト転送プロトコル) (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「TCP/IP (トランスミッションコントロールプロトコル / インターネットプロトコル) (TCP/IP (Transmission Control Protocol/Internet Protocol))」ベースのインターネットプロトコル。

**HTTP-NG (hypertext transfer protocol - next generation)** (n.) 次世代 HTTP。

**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。

**HTTPS (hypertext transfer protocol secure)** (n.) 「SQL (構造化クエリー言語) (SQL (structured query language))」を使用した HTTP 実装のセキュリティ保護バージョン。

**hypertext transfer protocol secure** 「[HTTP-NG \(hypertext transfer protocol - next generation\)](#)」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リング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ースを使用して、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 と呼ばれます。RFC 1413 に定義されています。

**IDL (インタフェース定義言語) (IDL (interface definition language))** (n.) リモート「[CORBA \(共通オブジェクトリクエストブローカーアーキテクチャ\) \(CORBA \(common object request broker architecture\)\)](#)」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 \(遠隔手続き呼び出し\) \(RPC \(remote procedure call\)\)](#)」)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整理化するコードをスタブできるようにします。

**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 \(リモートメソッド呼び出し\) \(RMI \(remote method invocation\)\)](#)」)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(「[CORBA \(共通オブジェクトリクエストブローカーアーキテクチャ\) \(CORBA \(common object request broker architecture\)\)](#)」)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。

**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 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 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) (HTTPD (hypertext transfer protocol daemon))**」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 (共通ゲートウェイインタフェース) (CGI (common gateway interface))**」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 Message-Based Interoperability Protocol**」を参照してください。

**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。

**imsimta コマンド (imsimta commands)** (n.) 「**MTA (メッセージ転送エージェント) (MTA (message transfer agent))**」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のセット。

**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。

**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の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、`/etc/inittab` 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。

**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。



**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 サーバーへの接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。

**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。「TCP/IP (トランスミッションコントロールプロトコル / インターネットプロトコル) (TCP/IP (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。

**IP (インターネットプロトコル) (IP (Internet Protocol))** (n.) 世界中のネットワークを接続する「TCP/IP (トランスミッションコントロールプロトコル / インターネットプロトコル) (TCP/IP (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。

**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 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、ISO 8601 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。

**iTIP** 「[iCalendar Transport-Independent Interoperability Protocol](#)」を参照してください。

**J2EE™ プラットフォーム (Java™ 2 Platform, Enterprise Edition) (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 \(アプリケーションプログラミングインタフェース\) \(API \(application programming interface\)\)](#)」、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。

**J2ME™ プラットフォーム (Java™ Platform, Micro Edition) (J2ME™ platform (Java™ Platform, Micro Edition))** (n.) モバイルデバイスに適した小規模アプリケーション環境。



**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 (Java™ アーカイブファイル) (JAR file (Java™ Archive file))** (n.) 多くのファイルを 1 つのファイルに集合させる場合に使用するファイル。JAR ファイルの拡張子は .jar です。

**JAR ファイル形式 (JAR file format)** (n.) 多数のファイルを 1 つのファイルに統合する、プラットフォームに依存しない Java™ アーカイブファイル形式。複数のアプレットと必要なコンポーネント (クラスファイル、イメージ、サウンド、その他のリソースファイル) を JAR ファイルにまとめて、1 回の HTTP トランザクションでブラウザにダウンロードできます。JAR ファイル形式はファイルの圧縮とデジタル署名もサポートしています。

**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 \(拡張可能マークアップ言語\) \(XML extensible markup language\)](#)」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。

**JavaBeans™ Activation Framework** 「[JAF \(JavaBeans™ Activation Framework\)](#)」を参照してください。

**JavaBeans™ コンポーネントアーキテクチャ (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。

**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。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。

**JavaMail™ (API, 拡張) (JavaMail™ (API, extension))** (n.) メールストアとの通信でアプリケーションが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 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 Pages™ テクノロジ (JavaServer Pages™ technology)** 「[JSP™ テクノロジ \(JSP™ technology\)](#)」を参照してください。

**Java™** (n.) 現代のプログラミング手法におけるさまざまな問題を解決するために Sun Microsystems, Inc. によって開発された、プラットフォームに依存しないオブジェクト指向プログラミング言語。

**Java™ 2 Platform, Enterprise Edition** 「[J2EE™ プラットフォーム \(Java™ 2 Platform, Enterprise Edition\) \(J2EE™ platform \(Java™ 2 Platform, Enterprise Edition\)\)](#)」を参照してください。

**Java™ 2 Platform, Micro Edition** 「[J2ME™ プラットフォーム \(Java™ Platform, Micro Edition\) \(J2ME™ platform \(Java™ Platform, Micro Edition\)\)](#)」を参照してください。

**Java™ Development Kit** 「[JDK™ ソフトウェア \(Java™ Development Kit\) \(JDK™ software \(Java™ Development Kit\)\)](#)」を参照してください。

**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™ IDL (API, 拡張) (Java インタフェース定義言語) (Java™ IDL (API, extension) (Java Interface Definition Language))** (n.) 「[CORBA \(共通オブジェクトリクエストブローカーアーキテクチャ\) \(CORBA \(common object request broker architecture\)\)](#)」との標準ベースの互換性および接続性を実現する、Java プログラミング言語によって記述された「[API \(アプリケーションプログラミングインタフェース\) \(API \(application programming interface\)\)](#)」。

**Java™ System 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。

**Java™ System 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 のインタフェース。

**Java™ System Directory Server** (n.) 「[LDAP \(ライトウェイトディレクトリアクセスプロトコル\) \(LDAP \(Lightweight Directory Access Protocol\)\)](#)」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。

**Java™ System 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。

**Java™ System 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。

**Java™ System 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。Message Queue は JMS プロバイダです。

**Java™ System 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。Portal Server 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。

**Java™ System 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。Java System Instant Messaging Server と呼ばれます。

**Java™ System Portal Server Pack** (n.) Portal Server のアドオン製品を指す一般的な用語。

**Java™ System Portal Server デスクトップ (Java™ System 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、コンテナ階層や、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。

**Java™ System 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。Web Server は、Directory Server Access Management Edition 製品に組み込まれています。

**Java™ Web Start ソフトウェア (Java™ Web Start software)** (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。

**JAX-RPC (Java™ API for XML-based RPC)** (n.) XML ベースの「RPC (遠隔手続き呼び出し) (RPC (remote procedure call))」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。

**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.) DOM、SAX、および XSLT を使用して、XML ドキュメントの処理をサポートする Java API。アプリケーションが、特定の XML 処理実装に依存せずに、XML ドキュメントを解析および変換できるようにします。

**JAXR (Java™ API for XML Registries)** (n.) さまざまな種類の XML 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。

**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™ テクノロジ (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™ 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。

**JDK™ ソフトウェア (Java™ Development Kit) (JDK™ software (Java™ Development Kit))** (n.) Java アプレットまたはアプリケーションプログラムを記述するために使用するソフトウェアツール。

**JHTML (J-Sky ハイパーテキストマークアップ言語) (JHTML (J-Sky hypertext markup language))** (n.) 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 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 (Java™ Message Service)** (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 (Java™ Message Service 管理対象オブジェクト) (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) (JMS connection factory (Java™ Message Service connection factory))」または「JMS 送信先 (Java™ Message Service 送信先) (JMS destination (Java™ Message Service destination))」)。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 検索を使用してこれらのオブジェクトにアクセスします。

**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 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 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後に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 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 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消費される非同期要求、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す。メッセージヘッダーには、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.) 企業の複数のネーミング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、企業の異種ネーミング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 ネーミング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。

**JRE (Java™ 実行環境) (JRE (Java™ runtime environment))** (n.) Java™ Virtual Machine、Java コアクラス、およびサポートファイルで構成される「**JDK™ ソフトウェア (Java™ Development Kit) (JDK™ software (Java™ Development Kit))**」のサブセット。Java プログラミング言語で記述されたアプリケーションを実行時にサポートします。

**jspc ユーティリティ (jspc utility)** (n.) 「**JSP™ テクノロジ (JSP™ technology)**」によって作成されたページのコンパイラ。このユーティリティで、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™ テクノロジ (JSP™ technology)** (1) (n.) HTML または XML タグ、JSP タグ、および Java™ プログラミング言語によって記述されたコードを組み合わせることで記述したテキストページ。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。  
(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、および設定管理など、すべての JavaServer ページ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。

**JSS** 「**Network Security Services for Java (JSS)**」を参照してください。

**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™ サーバーによるトランザクションへのアクセスを可能にする API。

**JTS (Java トランザクションサービス) (JTS (Java transaction service))** (n.) トランザクションを処理する Java サービス。

**LDAP URL** (n.) 「**DNS (ドメインネームシステム) (DNS (domain name system))**」を使用してディレクトリサーバーの位置を特定し、LDAP によってクエリを完了する手段となる「**URL (uniform resource locator)**」。たとえば LDAP URL は、`ldap://ldap.example.com` のような URL になります。

**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 (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 データ交換形式 (LDAP Data Interchange Format)** 「LDIF (LDAP データ交換形式) (LDIF (Data Interchange Format))」を参照してください。

**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 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 (ライトウェイトディレクトリアクセスプロトコル) (LDAP (Lightweight Directory Access Protocol))** (n.) TCP/IP を介して複数のプラットフォーム上で実行でき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、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 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。

**LDAPv3** (n.) 「LDAP (ライトウェイトディレクトリアクセスプロトコル) (LDAP (Lightweight Directory Access Protocol))」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 (Data Interchange Format))** (n.) *type:value* ペアを使用して、Directory Server のエンタリをテキスト形式で表現するために使用する形式。

**Legato Networker® ソフトウェア (Legato NetWorker® software)** (n.) Legato Systems, Inc. が提供するサードパーティ製バックアップユーティリティ。



**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)** (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。

**LMTP (ローカルメール転送プロトコル) (LMTP (Local Mail Transfer Protocol))** (n.) 「**SMB プロトコル (SMB protocol)**」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。

**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 (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。

**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。

**Messaging Multiplexor** 「**MMP (Messaging Multiplexor)**」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、*msg\_svr\_base* 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。

**MHS (メッセージ処理システム) (MHS (message handling system))** (n.) 接続されている「[MTA \(メッセージ転送エージェント\) \(MTA \(message transfer agent\)\)](#)」、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 \(ネットワーク管理ステーション\) \(NMS \(network management station\)\)](#)」から確認できます。「[AUTH](#)」も参照してください。

**MIME (多目的インターネットメール拡張) (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。

**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。

**Mobile 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。

**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 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 `imta.cnf` 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とします。「**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 (メッセージ転送エージェント) (MTA (message transfer agent))**」間でメッセージをルーティングする処理。

**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。

**MUA** 「**ユーザーエージェント (user agent)**」を参照してください。

**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) (DNS (domain name system))**」レコードの一種。

**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。

**NAFO** 「**ネットワークアダプタフェイルオーバーグループ (network adapter failover group)**」を参照してください。

**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) (MTA (message transfer agent))**」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。

**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 (applet)**」。Netlet を使用すると、「**TCP/IP (トランスミッションコントロールプロトコル / インターネットプロトコル) (TCP/IP (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 接続を通し、セキュリティ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 のいずれか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。

**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ブレットから「SQL (構造化クエリー言語) (SQL (structured query language))」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。

**NIS (ネットワーク情報サービス) (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。

**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。

**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。

**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。

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

**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。

**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。

**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。「**LDAP (ライトウェイトディレクトリアクセスプロトコル) (LDAP (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。

**OSI ツリー (開放型システム間相互接続ツリー) (OSI tree (Open System Interconnect tree))** (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 (ディレクトリ情報ツリー) (DIT (directory information tree))**」。OSI ツリー内の識別名 (「**DN (識別名) (DN (distinguished name))**」) は、cn=billt,o=bridge,c=us のようになります。

**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。

**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。

**PKI (公開鍵インフラストラクチャ) (PKI (public key infrastructure))** (n.) ユーザーのアイデンティティ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。

**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) (MTA (message transfer agent))**」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。

**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。「**Sun Java™ System Portal Secure Remote Access (SRA)**」も参照してください。

**PTA Directory Server** (n.) 「**PTA (パススルー認証) (PTA (pass-through authentication))**」では、PTA Directory Server が、受信するバインド要求を認証する側の Directory Server に送信、つまりパススルーします。

**PTA LDAP URL** (n.) 「**PTA (パススルー認証) (PTA (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。

**PTA (パススルー認証) (PTA (pass-through authentication))** (n.) 1 つの Java Enterprise System Directory Server が別の Directory Server に問い合わせるバインドルールを確認するためのメカニズム。

**Public Network Management** (n.) 障害モニタリングとフェイルオーバー機能によって、単一のネットワークアダプタまたはケーブルの障害によるノードの可用性の喪失を防止するソフトウェア。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 (network adapter failover group)**」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。障害モニタリングとフェイルオーバー機能は、連携動作してリソースの可用性を確保します。

**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。

**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) (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。

**RAR ファイル (リソースアーカイブファイル) (RAR file (resource archive file))** (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。

**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 とも呼ばれます。

**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。

**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。

**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 ( 識別名 ) (DN (distinguished name))**」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。

**Resource Management API** (n.) クラスタ環境でのアプリケーションの可用性を高める、SunPlex™ システム内部のアプリケーションプログラミングインタフェース。RMAPI とも呼ばれます。

**ResultSet オブジェクト (ResultSet object)** (n.) `java.sql.ResultSet` 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。

**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。

**RGM ( リソースグループマネージャー ) (RGM (resource group manager))** (n.) 選択されたクラスタノードのクラスタリソースを自動的に起動または停止することにより、それらのリソースの可用性とスケーラビリティを高めるソフトウェア機能。RGM は、ハードウェアまたはソフトウェアで障害が発生したり、再起動されたときに、あらかじめ設定されているポリシーに従って動作します。

**RMI ( リモートメソッド呼び出し ) (RMI (remote method invocation))** (n.) Java API の標準セット。開発者は、オブジェクトをリモートプロセスに渡すことが可能なリモートインタフェースを記述することができます。

**RMIC** (n.) リモートメソッド呼び出しコンパイラ。

**root** (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。

**root DN** (n.) 「**ディレクトリマネージャー (Directory Manager)**」の「**DN ( 識別名 ) (DN (distinguished name))**」。

**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 ( 識別名 ) (DN (distinguished name))**」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 ) (DSE (directory server entry))**」も参照してください。

**ResultSet オブジェクト (ResultSet 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。ResultSet オブジェクトは、`java.sql.ResultSet` 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。

**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に関係する関数。

**SASL (シンプル認証とセキュリティ層) (SASL (simple authentication and security layer))** (n.) POP、IMAP、または「[SMB プロトコル \(SMB protocol\)](#)」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 \(ポストオフィスプロトコルバージョン 3\) \(POP3 \(Post Office Protocol Version 3\)\)](#)」と「[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 \(Internet Message Access Protocol Version 4\)\)](#)」も参照してください。

**Scalable Coherent Interface** (n.) 「[クラスターインターコネクタ \(cluster interconnect\)](#)」として使用される高速インターコネクタハードウェア。

**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。

**Secure Socket Layer 「SSL (Secure Sockets Layer)」** を参照してください。

**sendmail** (n.) UNIX のみ。共通の「[MTA \(メッセージ転送エージェント\) \(MTA \(message transfer agent\)\)](#)」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。

**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、「[TCP/IP \(トランスミッションコントロールプロトコル / インターネットプロトコル\) \(TCP/IP \(Transmission Control Protocol/Internet Protocol\)\)](#)」プロトコルの上位プロトコル、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。



**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.) メールのフィルタ用に提案されている言語。

**Simple Mail Transfer Protocol** 「[SMB プロトコル \(SMB protocol\)](#)」を参照してください。

**Simple Network Management Protocol** 「[SNMP \(Simple Network Management Protocol\)](#)」を参照してください。

**SIMS** (n.) Solstice Internet Mail Server™ および Sun Internet Mail Server™。

**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 \(Simple Mail Transfer Protocol\)](#)」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す。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。

**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 AUTH** 「[AUTH](#)」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 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する 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 \(ネットワーク管理ステーション\) \(NMS \(network management station\)\)](#)」の間でデータを送受信します。

**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。

**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) (NMS (network management station))**」の間で情報を交換するソフトウェア。

**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.) インターネット全体の多様なオペレーティングシステム環境に分散されたオブジェクトで、メソッドを呼び出す標準化された方法を定義するプロトコル。XML ベースのデータ構造とハイパーテキスト転送プロトコル (Hypertext Transfer Protocol、HTTP) の組み合わせを使用します。

**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 (SOF 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™ 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、/devices 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、devices/sbus@1f,0/SUNW,fas@e,8800000/sd@6,0:c,raw 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。

**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、通常の名前は /dev/rdisk/c0t2d0s2 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。「**DID 名 (DID name)**」と「**Solaris™ 物理名 (Solaris™ Physical name)**」も参照してください。

**Solstice DiskSuite™ ソフトウェア (Solstice DiskSuite™ software)** (n.) SunPlex™ システムによって使用されるボリュームマネージャー。「**仮想サーバークラス (virtual server class)**」も参照してください。

**SQL (構造化クエリ言語) (SQL (structured query language))** (n.) リレーショナルデータベースアプリケーションで一般的に使用される言語。SQL2 および SQL3 は、この言語のバージョンを表します。

**SSL (Secure Sockets 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。

**SSL 証明書 (SSL certificate)** (n.) 「PKI (公開鍵インフラストラクチャ) (PKI (public key infrastructure))」を使用してセキュリティ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。

**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、セキュリティ証明書によりユーザーのアイデンティティを確認する認証方法。

**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** 「サーバー側規則 (server side rules)」を参照してください。

**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。

**summary object interchange format** 「SOIF (summary object interchange format)」を参照してください。

**Sun Java™ System Communication 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。

**Sun Java™ System 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。

**Sun Java™ System Portal 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 Synchronization** (n.) Microsoft Windows のパーソナルコンピュータで動作し、カレンダーイベントおよびタスクを、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。

**SunPlex™ システム (SunPlex™ system)** (n.) 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、ハードウェアと Sun™ Cluster ソフトウェアの統合システム

**Sun™ Cluster ソフトウェア (Sun™ Cluster software)** (n.) SunPlex™ システムのソフトウェア部分。「SunPlex™ システム (SunPlex™ system)」も参照してください。

**symlinks** (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。

**System Service Processor** (n.) Sun Enterprise™ 10000 サーバー構成で、特にクラスタメンバーとの通信で使用される、クラスタ外のデバイス。

**TCP/IP (トランスミッションコントロールプロトコル/インターネットプロトコル)**  
**(TCP/IP (Transmission Control Protocol/Internet Protocol))** (n.) インターネットおよびエンタープライズネットワークの主要なネットワークプロトコル。

**telnet** (n.) インターネットプロトコル群の仮想端末プロトコル。1つのホストのユーザーがリモートホストにログインし、そのホストの通常の端末ユーザーとして対話式で操作できます。

**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。

**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。「**SQL (構造化クエリー言語) (SQL (structured query language))**」の標準で、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。

**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。

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

**UA** 「**ユーザーエージェント (user agent)**」を参照してください。

**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。

**uid** (n.) UNIX のみ。UNIX システムの各ユーザーに関連付けられた固有番号。

**Uniform Resource Indicator** 「**URI (universal resource identifier)**」を参照してください。

**uniform resource locator** 「**URL (uniform resource locator)**」を参照してください。

**universal principal name** (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。

**URI (universal resource identifier)** (n.) ドメインの固有リソースを記述します。ローカルには、ベースディレクトリのサブセットとして記述され、`/ham/burger` はベースディレクトリになり、URI は `toppings/cheese.html` を指定します。対応する URL は、`http://domain.port/toppings/cheese.html` のようになります。

**URL (uniform resource locator)** (n.) ドキュメントを要求するためにサーバーとクライアントが使用するアドレス指定方式。多くの場合、URL は位置と呼ばれます。URL の書式は、`プロトコル://マシン.ポート/ドキュメント` です。たとえば URL は、`http://www.example.com/index.html` のようになります。

**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 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。

**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。

**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。

**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。

**UUCP (UNIX to UNIX Copy Protocol)** (n.) UNIX のみ。同意する UNIX システム間の通信に使用されるプロトコルです。

**/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。

**Veritas Cluster Server** (n.) Java Enterprise System Messaging Server と統合できる Veritas Software 製の高可用性クラスタリングソフトウェア。

**VERITAS Volume Manager** (n.) SunPlex™ システムによって使用されるボリュームマネージャー。「[仮想サーバークラス \(virtual server class\)](#)」も参照してください。

**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ーを実現するテクノロジー。

**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 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。

**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。

**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 \(server-side include Hypertext markup language\)](#)」や「[CGI \(共通ゲートウェイインタフェース\) \(CGI \(common gateway interface\)\)](#)」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 キャッシュ (web cache)** 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)** 「[コンテナ \(container\)](#)」を参照してください。

**Web サーバー (web server)** (1) (n.) HTML ページと Web アプリケーションを格納、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 ブラウザからのユーザー要求に応答します。

(2) (n.) HTTP や FTP などの Web 要求に応答するアプリケーション。

(3) (n.) World Wide Web アクセスを提供するために導入されるソフトウェアプログラムまたはサーバーコンピュータ。Web サーバーは、ユーザーからの要求を受け取り、要求されたファイルやアプリケーションを検索し、さらにエラーメッセージを発行します。

**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 サービス (web service)** (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化インターネットプロトコルに準拠するサービス。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。

(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。

**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.) 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。

**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 (共通ゲートウェイインタフェース) (CGI (common gateway interface))**」プログラム。

**WML (ワイヤレスマークアップ言語) (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。

**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.400** (n.) メッセージ処理システムの標準。

**X.500 標準 (X.500 standard)** (n.) 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 ISO/ITU-T ドキュメント。「**LDAP (ライトウェイトディレクトリアクセスプロトコル) (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。

**XA プロトコル (XA protocol)** (n.) 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。

**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。XML が拡張可能であるのは、HTML とは違ってマークアップ記号に制限がなく自己定義性があるためです。Calendar Server は、XML と XSL を使用して Calendar Express ユーザーインタフェースを生成します。

**XML 名前空間 (XML namespace)** (n.) 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。



**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、フォーマットしなければ読み取ることができず、通常、フォーマットはスタイルシートを使用して実行されます。スタイルシートは、コンピュータ画面または印刷ページでのドキュメントの表示に対する特定の XML タグの働きを規定した、フォーマット規則で構成されています。

**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。

**Zulu 時間 (Zulu time)** (n.) 「GMT (グリニッジ標準時) (GMT (Greenwich Mean Time))」と UTC (協定世界時) を指す軍用語。

**アーキテクチャ (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、アーキテクチャの設計には通常、「論理アーキテクチャ (logical architecture)」と「配備アーキテクチャ (deployment architecture)」があります。

**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、リソースで活動する Web サービスです。主体 (アイデンティティ) に関連するデータ属性で、何らかの動作を取得、更新、または実行します。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。

**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。

**アイデンティティ連携 (identity federation)** (n.) 「アカウント連携 (account federation)」を参照してください。

**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。

**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。



**アカウント連携 (account 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。アイデンティティ連携とも呼ばれます。

**アクセス権 (permissions)** (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。

(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。

(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を使って、「[ACE \(アクセス制御エントリ\) \(ACE \(access control entry\)\)](#)」の文字列としてアクセス権を設定します。「[ACL \(アクセス制御リスト\) \(ACL \(access control list\)\)](#)」も参照してください。

**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、すべてを付与または拒否できます。

**アクセス制御 (access control)** (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保護するための方法。

**アクセス制御エントリ (access control entry)** 「[ACE \(アクセス制御エントリ\) \(ACE \(access control entry\)\)](#)」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。

**アクセス制御命令 (access control instruction)** 「[ACI \(アクセス制御命令\) \(ACI \(access control instruction\)\)](#)」を参照してください。

**アクセス制御リスト (access control list)** 「[ACL \(アクセス制御リスト\) \(ACL \(access control list\)\)](#)」を参照してください。

**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。

**アクティブでないブート環境 (inactive boot environment)** (n.) 現在ブートしていないか、次のリブ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 \(active boot environment\)](#)」も参照してください。

**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 \(DRU\) \(data redundancy unit \(DRU\)\)](#)」も参照してください。

**アクティブブート環境 (active boot environment)** (n.) 現在、起動して稼働している環境。

**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に転送するプロセス。

**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。

**アップグレード (upgrade)** (n.) 更新済み製品ビットを既存の製品ビットにインストールするプロセス。

**アドレス (address)** (1) (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ープの両方に表示されます。エンベロープのアドレスは、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。

(2) (n.) ネットワーキングでは、ネットワークでノードを識別するための一意のコード。example.corp.com のような名前は、DNS により「ドットで区切られた 4 個で 1 セットとなる」アドレス (168.124.0.0) に変換されます。

**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。

**アドレストークン (address token)** (n.) 書き換え規則パターンのアドレス要素。

**アプリケーションクライアントコンテナ (application client container)** 「[コンテナ \(container\)](#)」を参照してください。

**アプリケーションコンポーネント (application component)** 特定の計算機能を実行するカスタム開発ソフトウェア「[コンポーネント \(component\)](#)」。「[エンドユーザー \(end user\)](#)」やほかのアプリケーションコンポーネントに「[ビジネスサービス \(business service\)](#)」を提供します。アプリケーションコンポーネントは、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います。これらのコンポーネントは、単独または組み合わせて、「[Web サービス \(web service\)](#)」としてカプセル化できます。

**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。

**アプレット (applet)** (m.) Java™ プログラミング言語で書かれた、Web ブラウザで動作する小さなアプリケーション。アプレットは通常、Web ページに呼び出されたり埋め込まれたりして、特別な機能を提供します。Java アプレットは、Java™ プログラミング言語によって書かれた小さなアプリケーションプログラムで、Web ページと一緒にユーザーのブラウザに送信できます。Java アプレットは、ユーザー要求をサーバーに送信せずにタスクを実行できます。Instant Messaging クライアントは Java アプレットです。「[サーブレット \(servlet\)](#)」も参照してください。

**アムネシア (amnesia)** (n.) 「[カテゴリ \(category\)](#)」が古くなったことによりシャットダウンしたあとにクラスタが再起動する状態。たとえば、ノード 1 だけが動作する 2 ノード構成のクラスタにおいて、ノード 1 でクラスタ設定に変更があると、ノード 2 の CCR は古くなります。クラスタをシャットダウンしてノード 2 で再起動すると、ノード 2 の古い CCR によってアムネシア状態が発生します。

**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。

**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。

**アンインストール (uninstallation)** (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。

**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。

**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。

**安全なファイルシステム (safe file system)** (n.) ログイン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。

**移行 (migration)** (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行するプロセス。

**委譲 (delegation)** (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せん。

**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) (MTA (message transfer agent))」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。

**イベント (event)** (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。

(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する名前付きアクション。

(3) (n.) 管理対象オブジェクトの状態、制御、重要度、または記述の変更。

**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。

**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)」。ある入れ子のロールを構成するメンバーは、そのメンバーに含まれている複数のロールのすべてのメンバーから成る和集合です。入れ子のロールにより、拡張範囲を定義し、ほかのサブツリーにあるロールのメンバーを取り込むこともあります。

**インスタンスディレクトリ** (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、インスタンスディレクトリは「**サーバールート (server root)**」のサブディレクトリ `server-root/msg-instance/` です。*instance* は、インストール時に指定されるサーバー名です。「**インストールディレクトリ (installation directory)**」も参照してください。

**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 `server-root/bin/msg/` です。「**インスタンスディレクトリ**」も参照してください。

**インターネットプロトコル (Internet Protocol)** 「**IP (インターネットプロトコル) (IP (Internet Protocol))**」を参照してください。

**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) (IMAP4 (Internet Message Access Protocol Version 4))**」を参照してください。

**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。

**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。

**イントラネット (intranet)** (n.) 企業や組織内における複数「**TCP/IP (トランスミッションコントロールプロトコル/インターネットプロトコル) (TCP/IP (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。イントラ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 (firewall)**」、「**エクストラネット (extranet)**」も参照してください。

**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。

**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。

**エージェント (agent)** (n.) ルータ、ホスト、または X 端末など、ネットワークデバイスの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 \(intelligent agent\)](#)」も参照してください。

**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 \(メッセージ転送エージェント\) \(MTA \(message transfer agent\)\)](#)」はメッセージを送信側に戻し、メッセージのコピーを削除します。

**エイリアス (alias)** (n.) 電子メールアドレスの別名。

**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。

**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。

**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せん。

**エクспан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。メールエクспанダは、メール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、メールエクспан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。

**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。

**エラーメッセージ (error message)** (n.) エラーまたはほかの状況について報告するメッセージ。Java Enterprise System Messaging Server は、特に処理不能な電子メールメッセージを受信したときなど、さまざまな状況でメッセージを生成します。情報の通知だけを目的とする通知エラーと呼ばれるメッセージもあります。

**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す。

**エンティティ Bean (entity bean)** (n.) データベースの行などの物理的なデータに関連するエンタープライズ Bean。エンティティ Bean は、持続データに結び付けられているので生存期間は長くなります。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。「メッセージ駆動型 Bean (message-driven bean)」、「読み取り専用 Bean (read-only bean)」、および「セッション Bean (session bean)」を参照してください。

**エンドユーザー (end user)** (n.) 分散アプリケーションを使用するユーザー。多くの場合、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ースを介して使用します。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 (deployment architecture)」の重要な決定要因です。

**エントリ (entry)** (n.) 属性のグループと一意の識別名。

**エントリ 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。

**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に使用します。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。

**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。

**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。



**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) (OID (object identifier))」を参照してください。

**オブジェクトの持続性 (object persistence)** 「[持続性 \(persistence\)](#)」を参照してください。

**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。

**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。

**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。

**親アクセス (parent access)** (n.) 「[バインド DN \(バインド識別名\) \(bind DN \(bind distinguished name\)\)](#)」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。

**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。

**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。

**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。

**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。

**開始位置 (starting points)** (n.) 検索エンジンロボットが訪問し、リソースの列挙を開始するサイトのリスト。

**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 \(deployment architecture\)](#)」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。

**書き換え規則 (rewrite rule)** (n.) 「[MTA \(メッセージ転送エージェント\) \(MTA \(message transfer agent\)\)](#)」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、以下の機能があります。(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。

**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。

**拡張可能スタイル言語 (extensible style language)** 「[XSL \(拡張可能スタイル言語\) \(XSL \(extensible style language\)\)](#)」を参照してください。

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

**拡張可能マークアップ言語 (Extensible Markup Language)** 「[XML \(拡張可能マークアップ言語\) \(XML \(extensible markup language\)\)](#)」を参照してください。

**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。

**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。「ドキュメントタイプ定義」、「ホストドメイン (hosted domain)」も参照してください。

**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。

**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。

**カプセル化する (encapsulate)** (v.) 情報をモジュール内に局所化すること。オブジェクトは、データと実装をカプセル化するので、オブジェクトのユーザーはそのオブジェクトをサービスを提供するブラックボックスと見なすことができます。インスタンスの変数とメソッドは、追加、削除、または変更できますが、オブジェクトの提供するサービスが同じままであれば、オブジェクトを使用するコードは、そのオブジェクトを書き換えずに使い続けることができます。

**可用性サービス (availability service)** (n.) Sun Java System 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。

**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 と呼ばれます。

**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) (CAP (calendar access protocol))」を参照してください。

**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 \(カレンダー検索データベース\) \(CLD \(Calendar Lookup Database\)\)](#)」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 \(カレンダーユーザーエージェント\) \(CUA \(Calendar user agent\)\)](#)」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、それを後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。

**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。

**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。たとえば `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。

**管理インタフェース (administration interface)** (n.) Java™ Enterprise System サーバーの設定と管理に使用するブラウザベースのフォームの集まり。「[CLI \(コマンド行インタフェース\) \(CLI \(command-line interface\)\)](#)」も参照してください。

**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。

**管理コンソール (admin console)** (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。

**管理コンソール (administration console)** 「[コンソール \(console\)](#)」を参照してください。

**管理コンソール (administrative console)** (n.) クラスタ管理ソフトウェアを実行するために使用するワークステーション。

**管理サーバー (administration server)** (n.) Java Enterprise System Directory Server の管理機能を提供する専用のアプリケーションサーバーインスタンス。

**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 \(Directory Manager\)](#)」、「[管理サーバー \(administration server\)](#)」、「[サーバー管理者 \(server administrator\)](#)」、「[トップレベル管理者 \(top-level administrator\)](#)」、「[ドメイン管理者 \(domain administrator\)](#)」、「[組織管理者 \(organization administrator\)](#)」、「[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。

**管理情報ベース (management information base)** 「[MIB \(管理情報ベース\) \(MIB \(management information base\)\)](#)」を参照してください。

**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。

**管理対象オブジェクト (managed object)** (n.) 「MIB (管理情報ベース) (MIB (management information base))」の一部を構成する「SNMP (Simple Network Management Protocol)」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。

**管理ドメイン (administration domain)** (n.) 管理制御の対象範囲。「ドキュメントタイプ定義」も参照してください。

**管理ドメイン (administrative domain)** (n.) Sun Java Enterprise 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。「ドメイン (domain)」も参照してください。

**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。

**危殆化鍵リスト (compromised key list)** 「CKL (危殆化鍵リスト) (CKL (compromised key list))」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。

**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。

**キュー (queue)** (n.) ポイントツーポイント配信モデルを実装するために、管理者が作成するオブジェクト。メッセージを消費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。

**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。Directoy Server 4 との下位互換性を維持します。

**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。

**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空間にコンポーネントを配備すること。

**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) ドメイン全体で動作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。

**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。

**共有アドレス (shared address)** (n.) クラスタ内のノードで実行されるすべてのスケールラブルサービスによってバインド可能なネットワークアドレス。これにより、それらのノードでサービスを拡大することができます。クラスタには複数の共有アドレスを設定することができ、1 つのサービスは、複数の共有アドレスにバインドすることができます。

**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。

**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、「**システムサービスコンポーネント (system service component)**」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。

**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タ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。

**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。「**許可された属性 (allowed attributes)**」も参照してください。

**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。

**クライアント (client)** ソフトウェアの「**サービス (service)**」を要求するソフトウェア。ユーザーではありません。「**エンドユーザー (end user)**」を参照してください。クライアントは、別のサービスを要求するサービスであったり、エンドユーザーがアクセスする GUI コンポーネントであったりします。

**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。

**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。

\* **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。

**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。

**クライアント認証 (client authentication)** (n.) クライアントの証明書を認証するプロセス。証明書の署名と証明書チェーンが信頼できる CA のリストに載っている CA からのものであることを暗号化方式で検証します。「[認証 \(authentication\)](#)」、「[認証局 \(certificate authority\)](#)」も参照してください。

**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。

**クライアントプロファイル (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。

**クラスター (cluster)** (1)(n.) 高速ネットワークに接続され、全体として複数の CPU を搭載した 1 台のマシンのように動作するコンピュータのグループ。クラスターに属するサーバーの 1 台で障害が発生した場合は、そのサーバーのサービスを稼働中のサーバーにフェイルオーバーできます。「[フェイルオーバー \(failover\)](#)」も参照してください。

(2) (n.) マスター管理サーバーに追加され、マスター管理サーバーによって制御される、リモートスレーブ管理サーバーのグループ。クラスタ内のすべてのサーバーは、同じプラットフォームで構築し、同じユーザー ID とパスワードを設定する必要があります。

(3) (n.) 相互接続された 2 つ以上のノードまたはドメイン。クラスタファイルシステムを共有し、フェイルオーバーのため、並行稼働のため、またはスケーラブルなリソースを実行するために構成されています。

(4) (n.) メッセージングサービスを提供するために、並行して処理を行う複数の連結したブローカ。

**クラスタインターコネクト (cluster interconnect)** (n.) ケーブル、クラスタトランスポート接続点、クラスタトランスポートアダプタなどで構成されるハードウェアネットワークインフラストラクチャ。Messaging Server とデータサービスソフトウェアは、このインフラストラクチャを使用してクラスタ間通信を実行します。

**クラスタ構成リポジトリ (cluster configuration repository)** 「[カテゴリ \(category\)](#)」を参照してください。

**クラスタトランスポートアダプタ (cluster transport adapter)** (n.) ノードに常駐し、ノードをクラスタインターコネクトと接続するネットワークアダプタ。「[クラスタインターコネクト \(cluster interconnect\)](#)」も参照してください。

**クラスタトランスポートケーブル (cluster transport cable)** (n.) 端点に接続するネットワーク接続。クラスタトランスポートアダプタとクラスタトランスポート接続点の間、または 2 つのクラスタトランスポートアダプタの間の接続です。「[クラスタインターコネクト \(cluster interconnect\)](#)」も参照してください。

**クラスタトランスポート接続点 (cluster transport junction)** (n.) クラスタインターコネクトの一部として使用されるハードウェアのスイッチ。「[クラスタインターコネクト \(cluster interconnect\)](#)」も参照してください。

**クラスタノード (cluster node)** (n.) クラスタメンバーとして構成されているノード。クラスタノードは、現在のメンバーである場合とない場合があります。「[クラスタメンバー \(cluster member\)](#)」も参照してください。

**クラスタファイルシステム (cluster file system)** (n.) クラスタ全体で既存のローカルシステムへのアクセスにおいて高可用性を実現するクラスタサービス。

**クラスタメンバー (cluster member)** (n.) 現在のクラスタ構成のアクティブなメンバー。このメンバーは、ほかのクラスタメンバーとリソースを共有し、ほかのクラスタメンバーとクラスタのクライアントの両方にサービスを提供することができます。「[クラスタノード \(cluster node\)](#)」も参照してください。

**クラスタメンバーシップモニター (cluster membership monitor)** 「**CMM (クラスタメンバーシップモニター) (CMM (cluster membership monitor))**」を参照してください。

**クラス定義 (class definition)** (n.) 特定のオブジェクトのインスタンスを作成するために必要な情報について指定した定義。

**クラスパス (classpath)** (n.) Java クラスが格納されているディレクトリと JAR (Java™ Archive) ファイルを識別するパス。「**クラスローダー (classloader)**」も参照してください。

**クラスローダー (classloader)** (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。「**クラスパス (classpath)**」も参照してください。

**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.) 何らかの関連があるユーザーの集合。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 (user)**」、「**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 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。

**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。

**グローバルインタフェース (global interface)** (n.) 共有アドレスを物理的にホストするネットワークインタフェース。

**グローバルインタフェースノード (global interface node)** (n.) グローバルインタフェースをホストするマシンまたはドメイン。

**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。

**グローバルデバイス (global device)** (n.) ディスク、CD-ROM、およびテープなど、すべてのクラスタメンバーがアクセス可能なデバイス。

**グローバルデバイス名前空間 (global device namespace)** (n.) グローバルデバイスの論理的なクラスタ全域での名前を包含した名前空間。Solaris™ オペレーティングシステムのローカルデバイスは、/dev/dsk、/dev/rdisk および /dev/rmt の各ディレクトリで定義されます。グローバルデバイス名前空間では、グローバルデバイスを、/dev/global/dsk、/dev/global/rdisk、および /dev/global/rmt の各ディレクトリで定義します。

**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。

**グローバルリソース (global resource)** (n.) Java Enterprise System Messaging Server ソフトウェアのカーネルレベルで利用できる可用性の高いリソース。グローバルリソースには、ディスク (HA デバイスグループ)、クラスタファイルシステム、グローバルネットワーク接続などがあります。

**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。

**権限 (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。

**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。

**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。

**検索ベース (search base)** 「**ベース DN (ベース識別名) (base DN (base distinguished namebase))**」を参照してください。

**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 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メソッド。

**子 (child)** (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。

**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。

**公開鍵 (public key)** (n.) 公開鍵暗号化で使用する暗号化鍵。

**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。

**公開鍵暗号方式 (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保護されたチャネルがなくても、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。

**公開鍵インフラストラクチャ (public key infrastructure)** 「[PKI \(公開鍵インフラストラクチャ\)](#) (PKI (public key infrastructure))」を参照してください。

**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。

**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 \(アクセス制御リスト\) \(ACL \(access control list\)\)](#)」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。

**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なります。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ストラクチャ。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 \(active node\)](#)」、「[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 \(DRU\) \(data redundancy unit \(DRU\)\)](#)」も参照してください。

**更新履歴ログ (change log)** (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。

**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。

**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。

**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 \(ディレクトリ情報ツリー\) \(DIT \(directory information tree\)\)](#)」内の情報検索速度を高めます。

**個人デジタル証明書 (personal digital certificate)** 「[PDC \(個人デジタル証明書\) \(PDC \(personal digital certificate\)\)](#)」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。

**コネクタ (connector)** (n.) EIS 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。

**コネクタアーキテクチャ (connector architecture)** (n.) J2EE™ アプリケーションと EIS を統合するためのアーキテクチャ。このアーキテクチャは、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。

**コマンド行インタフェース (command line interface)** 「[CLI \(コマンド行インタフェース\) \(CLI \(command-line interface\)\)](#)」を参照してください。

**コミットする (commit)** (v.) 必要なコマンドをデータベ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)**」も参照してください。

**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 (collocation)** (n.) 同じノードに存在するプロパティ。この概念を活用して、クラスタ構成時にパフォーマンスの改善を計ります。

**コンシューマ (consumer)** (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。  
(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。

**コンシューマレプリカ (consumer replica)** (n.) マスターレプリカへの追加、変更、および削除操作のすべてを参照するレプリカ。サーバーは、ネーミングコンテキストが異なる任意の数のコンシューマレプリカを保持できます。

**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。

**コンソール (console)** (n.) 多くのサーバーソフトウェアコンポーネントの設定、監視、管理、およびトラブルシューティングを行う場合に使用する GUI。

**コンテナ (container)** (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するエンティティ。Java Enterprise System Application Server には Web コンテナと EJB™ コンテナがあり、アプリケーションクライアントコンテナをサポートしています。「**コンポーネント (component)**」も参照してください。

(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。

**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持続性 (container-managed persistence)** (n.) EJB™ コンテナは、エンティティ Bean の持続性を管理します。データアクセスロジックが Java Enterprise System Application Server によって提供される、エンティティ Bean の変数とデータストアの間でデータが転送されます。「[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、自動的に制御されます。「[Bean 管理による持続性 \(bean-managed persistence\)](#)」も参照してください。

**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プロパティだけでなく、Bean メソッドの個々のプロパティオーバーライド (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。

**コンポーネント (component)** (1) (n.) 分散アプリケーションの基礎となるソフトウェアロジック単位。コンポーネントは、Java Enterprise System に組み込まれている「[システムコンポーネント \(system component\)](#)」のいずれか、またはカスタム開発された「[アプリケーションコンポーネント \(application component\)](#)」です。アプリケーションコンポーネントは、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。

(2) (n.) Web アプリケーション、エンタープライズ Bean、メッセージ駆動型 Bean、アプリケーションクライアント、またはコネクタ。「[モジュール \(module\)](#)」も参照してください。

**コンポーネント規約 (component contract)** (n.) エンタープライズ Bean とそのコンテナ間の関係を確立する規約。

**コンポーネント製品の記述子ファイル (component product descriptor file)** (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。

**コンポーネントの状態 (component state)** (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。

**サーバー (server)** (n.) ハードウェアサーバーとは区別される、マルチスレッドのソフトウェアプロセス。外部インタフェースを使用してサービスにアクセスする「[クライアント \(client\)](#)」に、分散または結合した一連のサービスを提供します。

**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。

**サーバーインスタンス (server instance)** (n.) サーバーは、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。

**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 \(アクセス制御命令\) \(ACI \(access control instruction\)\)](#)」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、サーバーの安全性が確保されていないときに安全であるかのよう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し、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 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 \(installation directory\)](#)」と「[インスタンスディレクトリ](#)」も参照してください。

**サービス (service)** (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 (client)**」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。

**サービスクラス (class of service)** 「**CoS (サービスクラス) (CoS (class of service))**」を参照してください。

**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。

**サービス品質コンポーネント (service quality component)** 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および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、「**システムサービスコンポーネント (system service component)**」をサポートします。

**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。

**サーブレット (servlet)** (1) (n.) クライアントの要求に応答してコンテンツを生成するために Web サーバーが実行する、Java™ プログラミング言語で記述されたサーバー側プログラム。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。

(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。

**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。全般的には、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。「[サーブレットエンジン \(servlet engine\)](#)」を参照してください。

**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。

**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。

**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。

**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。

**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徴を述べ、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。

**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えるように作成されたコンポーネント。

**サフィックス (suffix)** (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タリ名。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。

**サブエージェント (subagent)** 「[SNMP サブエージェント \(SNMP subagent\)](#)」を参照してください。

**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。

**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。

**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「**完全修飾ドメイン名 (fully qualified domain name)**」 (`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 (subnet)** (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。

**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。

**サプライヤサーバー (supplier server)** (n.) レプリケーションにおいて、異なるサーバーにコピーされたレプリカを保持するサーバー。このサーバーはそのレプリカのサプライヤサーバーと呼ばれます。

**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。

**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。

**参照整合性 (referential integrity)** (n) 「**DN (識別名) (DN (distinguished name))**」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。

**識別名 (distinguished name)** 「**DN (識別名) (DN (distinguished name))**」を参照してください。

**自己アクセス (self access)** (n.) 「**バインド DN (バインド識別名) (bind DN (bind distinguished name))**」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。

**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 `enumerate-urls` 関数を呼び出して、`max` および `type` パラメータを渡します。

```
Enumerate fn=enumerate-urls max=1024 type=text/html
```

**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。

**システム管理者 (system administrator)** (n.) Java Enterprise System ソフトウェアを管理し、Java Enterprise System アプリケーションを配備するユーザー。

**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「**サービス (service)**」を提供する「**システムサービスコンポーネント (system service component)**」、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「**サービス品質コンポーネント (service quality component)**」、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。

**システムサービス (system service)** (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 (service)**」。システムサービスでは、通常、多くの「**サービス品質コンポーネント (service quality component)**」、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、アイデンティティとセキュリティサービス、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。

**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、後ほど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。

**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。

**持続性 (persistence)** (1) (n.) エンタープライズ Bean で、インスタンス変数と基礎となるデータベースとの間でエンティティ Beans の状態を転送するプロトコル。「**トランジエンス (transience)**」も参照してください。

(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。

**持続マネージャー (persistence manager)** (n.) コンテナにインストールされた「エンティティ Bean (entity bean)」の「持続性 (persistence)」に対する責任を持っているエンティティ。

**実行システム (runtime system)** (n.) プログラムを実行するソフトウェア環境。実行システムには、Java™ プログラミング言語で記述されたプログラムのロード、ネイティブメソッドの動的リンク、メモリ管理、および例外処理に必要なコードすべてが用意されています。また、Java インタプリタなどの Java 仮想マシンも実装されています。

**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。

**自動フェイルバック (automatic failback)** (n.) 主ノードで障害が発生し、その後クラスタメンバーとして再起動した後に、リソースグループまたはデバイスグループを主ノードに戻すプロセス。「スイッチバック (switchback)」も参照してください。

**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。

**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。

**主 (primary)** (n.) リソースグループまたはデバイスグループが現在オンラインになっている「ノード (node)」。

主は、リソースに関連付けられているサービスを現在ホストまたは実装しているノードです。「二次 (secondary)」も参照してください。

**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。

**主キー (primary key)** (n.) クライアントが特定のエンティティ Bean を検出できるようにする一意の識別子。

**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) 検索で使用します。

**主体 (principal)** (n.) 主体は、連携アイデンティティを取得でき、決定を下す能力があり、その利益となるように認証されたアクションが行われるエンティティです。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、または Liberty アーキテクチャのコンポーネントなどがあります。



**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。

**主ホスト名 (primary host name)** (n.) 主パブリックネットワークに接続されているノード名。主ホスト名は常に、/etc/nodename で指定されているノード名になります。「**二次ホスト名 (secondary host name)**」も参照してください。

**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、「**スパイダ (spider)**」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。

**上位参照 (upper reference)** (n.) 「**DIT (ディレクトリ情報ツリー) (DIT (directory information tree))**」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。

**障害モニター (fault monitor)** (n.) データサービスのさまざまな部分を精査して処置を講じるために使用される障害デーモンとプログラム。「**リソースモニター (resource monitor)**」も参照してください。

**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。

**状態 (state)** (1) (n.) 指定された時間におけるエンティティの環境または状態。  
(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。

**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。

**承認 (authorization)** (n.) 特定のアクセス権限をユーザーに付与するプロセス。承認は認証に基づいて行われ、アクセス制御によって実行されます。

**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。

**証明書失効リスト (certificate revocation list)** 「**CRL (証明書失効リスト) (CRL (certificate revocation list))**」を参照してください。

**証明書データベース (certificate database)** (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。

**証明書に基づく認証 (certificate-based authentication)** (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。

**証明書名 (certificate name)** (n.) 証明書とその所有者を特定する名前。

**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) (MTA (message transfer agent))**」コンポーネント。

**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。

**シングルインスタンスリソース (single-instance resource)** (n.) クラスタ全体で 1 つのリソースをアクティブにできるリソース。

**シングルサインオン (SSO) (single sign-on (SSO))** (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。

(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。

(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。

**シングルマスターレプリケーション (single-master replication)** (n.) マスターである 1 台のサーバーだけが、レプリケートされたデータへの「[LDAP \(ライトウェイトディレクトリアクセスプロトコル\) \(LDAP \(Lightweight Directory Access Protocol\)\)](#)」書き込みアクセスができるレプリケーションモデル。シングルマスターレプリケーションモデルでは、サブライヤかマスターサーバーが更新履歴ログを保持します。

**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。

**シンプルインデックス (simple index)** (n.) 画像要素のない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。

**シンプルオブジェクトアクセスプロトコル (Simple Object Access Protocol)** 「[SOAP \(シンプルオブジェクトアクセスプロトコル\) \(SOAP \(Simple Object Access Protocol\)\)](#)」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 \(シンプル認証とセキュリティ層\) \(SASL \(simple authentication and security layer\)\)](#)」を参照してください。

**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。

**スーパーユーザー (superuser)** (n.) UNIX のみ。UNIX マシンで選択可能な最大の権限が与えられるユーザー。root と呼ばれます。スーパーユーザーは、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。

**スイッチオーバー (switchover)** (n.) UNIX のみ。クラスタの 1 つのマスター (ノード) から別のマスター (リソースグループが複数の主に構成されている場合には複数のマスター) に対する、リソースグループまたはデバイスグループの順序どおりの転送。スイッチオーバーは、scswitch コマンドを使用して管理者によって開始されます。

**スイッチバック (switchback)** 「[フェイルバック \(failback\)](#)」を参照してください。

**水平方向のスケラビリティ (horizontal scalability)** (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。

**スキーマ (schema)** (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。

**スキーマ検査 (schema checking)** (n.) ディレクトリに追加されたり変更されたりしたエントリが、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 (summary object interchange format)**」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。

**スケラブルサービス (scalable service)** (n.) 複数のノードで同時に実行される、実装されたデータサービス。

**スケラブルリソース (scalable resource)** (n.) 「**クラスタインターコネクト (cluster interconnect)**」により複数のノード (各ノードのインスタンス) で実行されていても、そのサービスのリモートクライアントからは1つのサービスのように見えるリソース。

**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。

**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。

**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 Beans。複数のクライアント起動メソッドの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。状態のないセッション 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。

**スティッキー 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「**ロードバランス (load balancing)**」の手法。

**ストアードプロシージャ (stored procedure)** (n.) 「**スパイダ (spider)**」で書かれ、データベースに保存される文のブロック。ストアードプロシージャを使って、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。ストアードプロシージャを使うと、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。

**ストリーミング (streaming)** (n.) 「**HTTP (ハイパーテキスト転送プロトコル) (hypertext transfer protocol (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。

**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。

**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) (data redundancy unit (DRU))**」も参照してください。

**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。エントリ 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トリ 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。

**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。

**スレーブチャンネルプログラム (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ンネルプログラム。「**マスターチャンネルプログラム (master channel program)**」も参照してください。

**スレッド (thread)** (n.) プロセス内部の実行シーケンス。複数のスレッドが同時に実行される場合、そのプロセスはマルチスレッドです。各スレッドを逐次実行する場合、そのプロセスはシングルスレッドです。

**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った文字列。

**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。

**生成メソッド (create method)** (n.) エンタープライズ Bean の作成時にカスタマイズするメソッド。

**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)。

**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。

**セキュリティモジュールデータベース (security-module database)** (n.) 「**SQL (構造化クエリ言語) (SQL (structured query language))**」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 とも呼ばれます。

**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。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)**」と「**ステートレスセッション Bean (stateless 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 (session)** (1) (n.) サーブレットが複数の「**HTTP (ハイパーテキスト転送プロトコル) (hypertext transfer protocol (HTTP))**」要求でのユーザーと Web アプリケーションとの対話を追跡するために使用するオブジェクト。

(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。

(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。

(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。

**セッション鍵 (session key)** (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。

**接続 (connection)** (1) (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。

**接続ファクトリ (connection factory)** (n.) J2EE™ コンポーネントがリソースにアクセスできるように、接続オブジェクトを生成するオブジェクト。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) を作成します。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。

**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。

**設定 (configuration)** (n.) サーバーを調整する、またはコンポーネントのメタデータを提供するプロセス。通常、特定のコンポーネントの設定は、そのコンポーネント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 \(administrative console\)](#)」、「[配備記述子 \(deployment descriptor\)](#)」も参照してください。

**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.) Messaging システムの特定のコンポーネントに対する設定パラメータを記述したファイル。



**宣言によるセキュリティ (declarative security)** (n.) セキュリティプロパティをコンポーネントの設定ファイル内で宣言し、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 (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。

**潜在的な主ノード (potential primary)** (n.) 「**主 (primary)**」ノードで障害が発生した場合に、「**フェイルオーバーリソース (failover resource)**」タイプを制御可能なクラスタメンバー。「**デフォルトマスター**」も参照してください。

**潜在的マスター (potential master)** 「**潜在的な主ノード (potential primary)**」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 Java Enterprise System 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後にコンシューマに配信されます。この物理的な送信先は、管理対象オブジェクトによって特定され、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コンシュームする元となる送信先を指定します。

**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。

**相対識別名 (relative distinguished name)** 「**RDM**」を参照してください。

**属性 (attribute)** (1) (n.) サー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名前と値のペアでもあります。「**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。

(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。

(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。

**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「要求オブジェクト (request object)」を参照してください。

**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。「主 (primary)」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。

**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を再起動する方法。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。

**ターゲット (target)** (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) (ACI (access control instruction))**」が適用されるディレクトリ情報を識別します。

**ターゲットのエントリ (target entries)** (n.) 「**CoS (サービスクラス) (CoS (class of service))**」の範囲内のエントリ。

**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間の接続は待機ソケット上で確立されます。

**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。

**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。ブラウザは、MD5 アルゴリズムを使用してダイジェスト値を作成します。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。

**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。

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

**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。

**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。

**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。

**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。

**タイムゾーン (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) (GMT (Greenwich Mean Time))」は 0)、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。

**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。

**対話状態 (conversational state)** (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。

**多重ホストディスク (multihost disk)** (n.) 複数のノードに物理的に接続されているディスク。

**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。

**多目的インターネットメール拡張 (multipurpose internet mail extensions)** 「**MIME (多目的インターネットメール拡張) (MIME (multipurpose internet mail extensions))**」を参照してください。

**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。

**端点 (endpoint)** (n.) クラスタトランスポートアダプタまたはクラスタトランスポート接続点の物理ポート。

**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。「[デフォルトカレンダー \(default calendar\)](#)」も参照してください。

**端末集配信装置 (terminal concentrator)** (n.) Sun Enterprise™ 10000 構成ではない構成において、クラスタ外にあり、特にクラスタメンバーとの通信で使用されるのデバイス。

**チェックポイント (checkpoint)** (n.) 両者間のソフトウェアの同期が取れた状態を維持するために、主ノードから二次ノードに送信される通知。「[主 \(primary\)](#)」と「[二次 \(secondary\)](#)」も参照してください。

**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。

**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。

**チャット (chat)** (n.) Instant Messaging のインスタントメッセージのバージョン。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。

**チャンネル (channel)** (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。

(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。

**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。

**チャンネルブロック (channel block)** (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。

**チャンネルホストテーブル (channel host table)** (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。

**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。

**調査 (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応答することができます。

**直接従属 (immediate subordinate)** (n.) 「[DIT \(ディレクトリ情報ツリー\) \(DIT \(directory information tree\)\)](#)」では、自身の「[識別名 \(distinguished name\)](#)」が、親エントリの識別名に自身の「[RDM](#)」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 \(ディレクトリ情報ツリー\) \(DIT \(directory information tree\)\)](#)」では、自身の「[識別名 \(distinguished name\)](#)」に、ほかのエントリの「[RDM](#)」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。

**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。

**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。

**通知メッセージ (notification message)** (n.) Messaging Server によって送信されるメッセージの一種で、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。

**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。

**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。

**データサービス (data service)** (n.) RGM の制御下で高可用リソースとして実行するように装備されたアプリケーション。

**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する必要があります。「**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 (ディレクトリ情報ツリー) (DIT (directory information tree))**」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。

**データソース (data source)** (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。

**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。

**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)** 「**DWP (データベースワイヤプロトコル) (DWP (database wire protocol))**」を参照してください。

**テーブル (table)** (n.) データベースの行および列内に保存されている関連データの名前付きのグループ。

**デーモン (daemon)** (n.) UNIX のみ。バックグラウンドで動作し、端末に依存せず、必要に応じて機能を実行するプログラムです。デーモンプログラムの一般的な例として、メールハンドラ、ライセンスサーバー、印刷デーモンなどがあります。

**定義エン트리 (definition entry)** 「**CoS 定義エン트리 (CoS definition entry)**」を参照してください。

**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。

**ディスクグループ (disk group)** 「**デバイスグループ (device group)**」を参照してください。

**ディスクセット (diskset)** 「[デバイスグループ \(device group\)](#)」を参照してください。

**ディスクデバイスグループ (disk device group)** 「[デバイスグループ \(device group\)](#)」を参照してください。

**ディスパッチャ (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。

**定足数デバイス (quorum)** (n.) 実行するクラスタの定足数を決める場合のデバイス票に数えられる、2つ以上のノードで共有されるディスク。クラスタは、デバイス票による定足数が確保されている場合にのみ稼働できます。定足数デバイスは、クラスタが別々のノードセットに区分化されたときに、新規クラスタを構成するノードセットを確立するために使用されます。

**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取り用に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)** 「[CUA \(カレンダーユーザエージェント\) \(CUA \(Calendar user agent\)\)](#)」を参照してください。

**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 \(ベース識別名\) \(base DN \(base distinguished namebase\)\)](#)」も参照してください。

**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リポジトリから、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 \(ライトウェイトディレクトリアクセスプロトコル\) \(LDAP \(Lightweight Directory Access Protocol\)\)](#)」、「[Java™ System Directory Server](#)」、および「[Java™ System 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。



**ディレクトリサーバーコンソール (Directory Server Console)** (n.) ディレクトリの内容を参照、設定、および管理するグラフィカルユーザーインターフェースを備えた LDAP クライアントアプリケーション。ディレクトリサーバーコンソールは、Java Enterprise System Directory Server 製品のコンポーネントです。

**ディレクトリサービス (directory service)** (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。

**ディレクトリ情報ツリー (directory information tree)** 「[DIT \(ディレクトリ情報ツリー\) \(DIT \(directory information tree\)\)](#)」を参照してください。

**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。

**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。

**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。

**デジタル署名 (digital signature)** (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メカニズム。

**デスクトップ (Desktop)** 「[Java™ System Portal Server デスクトップ \(Java™ System Portal Server Desktop\)](#)」を参照してください。

**デバイス ID (device ID)** (1) (n.) Solaris™ オペレーティングシステムによって使用可能にされたデバイスを識別するための仕組み。デバイス ID については、`devid_get(3DEVID)` マニュアルページに記載されています。

(2) (n.) Messaging Server DID ドライバは、デバイス ID を使用して、異なるクラスタノードにある Solaris 論理名どうしの相互関係を判断します。DID ドライバは、各デバイスを精査してデバイス ID を調べます。デバイス ID が、クラスタ内の別の場所のデバイスと一致する場合は、両方のデバイスに同じ DID 名が付与されます。デバイス ID が以前のクラスタ内で検出されなかった場合は、新規 DID 名が割り当てられます。「[Solaris™ 論理名 \(Solaris™ logical name\)](#)」と「[DID](#)」も参照してください。

**デバイスグループ (device group)** (n.) クラスタの HA 設定の異なるノードから管理可能な、ディスクなどのデバイスリソースのユーザ定義グループ。このグループには、ディスク、Solstice DiskSuite™ ソフトウェアディスクセット、VERITAS Volume Manager ディスクグループなどを組み込むことができます。

**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)** (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。

**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す。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。

**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。

**デフォルトマスター** (n.) フェイルオーバーリソースタイプがオンラインにされたときのデフォルトクラスタメンバー。

**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。

**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。

**電子商取引 (e-commerce (electronic commerce))** (n.) インターネットで行うビジネス。

**転送 (forwarding)** 「[メッセージ転送 \(message forwarding\)](#)」を参照してください。

**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。

**テンプレートエントリ (template entry)** 「[coserver](#)」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。

**等価インデックス (equality index)** (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。

**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) (MTA (message transfer agent))**」ディレクトリキャッシュの更新。

**動的再読み込み (dynamic reloading)** (n.) サーバーを再起動せずにコンポーネントを更新して再読み込みするプロセス。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。動的再配備とバージョン管理とも呼ばれます。

**動的再配備 (dynamic redeployment)** 「**動的再読み込み (dynamic reloading)**」を参照してください。

**ドキュメント (document)** (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。

**ドキュメントタイプ定義 「DTD (ドキュメントタイプ定義) (DTD (document type definition))**」を参照してください。

**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。

**匿名アクセス (anonymous access)** (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。

**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、ファミリーアカウント、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。

**トピック (topic)** (n.) パブリッシュ / サブスクライブ配信モデルを実装するために、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がメッセージサブスクライバから分離されます。

**トポロジ (topology)** (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。

**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.) 単一のコンピュータシステムの制御下にあるリソース。「**管理ドメイン (administration domain)**」、「**DNS ドメイン (DNS domain)**」、「**ホストドメイン (hosted domain)**」、および「**仮想ドメイン (virtual domain)**」も参照してください。

(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。

**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。

**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) (DNS (domain name system))**」を参照してください。

**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。

**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。

**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.) 完了するか完全にロールバックする必要がある、不可分な作業の単位。

**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。「**ローカルトランザクション (local transaction)**」および「**グローバルトランザクション (global transaction)**」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.) データベース上で同時に実行されている複数のトランザクションをそれぞれに認識できる度合いを決定します。

**トランザクション属性 (transaction attribute)** (n.) トランザクションの範囲を制御する属性。

**トランザクションマネージャー (transaction manager)** (n.) 通常は「**XA プロトコル (XA protocol)**」を使用して、グローバルトランザクションを制御するオブジェクト。「**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。

**トランジェンス (transience)** (n.)使わ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。

**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) (MTA (message transfer agent))**」間でのメッセージ転送手段を提供するプロトコル。

**トランスミッションコントロールプロトコル (TCP) (Transmission Control Protocol (TCP))** (n.) 2つのホスト間での信頼性の高い接続指向のストリームサービスを提供するインターネットプロトコル群内の基本トランスポートプロトコル。

**トランスミッションコントロールプロトコル/インターネットプロトコル (TCP/IP) (Transmission Control Protocol/Internet Protocol (TCP/IP))** (n.) インターネットプロトコル群で使用される複数のネットワークプロトコルの総称。この名前は、トランスポート層のプロトコルである TCP (トランスミッションコントロールプロトコル) とネットワーク層のプロトコルである IP (インターネットプロトコル) の2つの主要ネットワークプロトコルを指します。

**取り消しパッチ (withdrawn patch)** (n.) 配信システムから削除されたパッチ。

**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。

**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) (DNS (domain name system))**」も参照してください。

**名前空間 (namespace)** (n.) LDAP ディレクトリのツリー構造。「**DIT (ディレクトリ情報ツリー) (DIT (directory information tree))**」も参照してください。

**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後で同じ「**DN (識別名) (DN (distinguished name))**」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。

**二次 (secondary)** (n.) 主で障害が発生した場合に、ディスクデバイスグループおよびリソースグループを制御できるクラスタメンバー。「**主 (primary)**」も参照してください。

**二次ホスト名 (secondary host name)** (n.) 二次パブリックネットワークのノードへのアクセスで使用される名前。「**主ホスト名 (primary host name)**」も参照してください。

**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) (data redundancy unit (DRU))」を参照してください。

**ニュースチャンネル (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。

**ニュースチャンネルリスト (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。

**認証 (authentication)** (1) (n.) ユーザーまたはマシンが、それが宣言するおりのユーザー本人またはマシン自体であるかどうかを判断するプロセス。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。

(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。

**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。

**認証局 (certificate authority)** 「**CA (認証局) (CA (certificate authority))**」を参照してください。

**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。



**ネーミングコンテキスト (naming context)** (n.) 「DN (識別名) (DN (distinguished name))」によって識別される「DIT (ディレクトリ情報ツリー) (DIT (directory information tree))」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。

**ネーミング属性 (naming attribute)** (n.) 「DIT (ディレクトリ情報ツリー) (DIT (directory information tree))」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。

**ネットワークアダプタフェイルオーバーグループ (network adapter failover group)** (n.) アダプタで障害が発生した場合には、相互にバックアップするように構成されている同じノードの同じサブネットにある、1つまたは複数のネットワークアダプタのセット。NAFO グループとも呼ばれます。

**ネットワークアドレスリソース (network address resource)** 「**ネットワーク資源 (network resource)**」を参照してください。

**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) (NMS (network management station))**」を参照してください。

**ネットワーク資源 (network resource)** (n.) 1つ以上の「**論理ホスト名 (logical host name)**」または「**共有アドレス (shared address)**」を持つリソース。

**ネットワークマネージャー (network manager)** (n.) 「**SNMP (Simple Network Management Protocol)**」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。

**ノード (node)** (n.) 「DIT (ディレクトリ情報ツリー) (DIT (directory information tree))」のエントリ。

(2) (n.) SunPlex™ システムに組み込むことが可能な、Sun™ Enterprise E10000 サーバー内の物理的なマシンまたはドメイン。「**ホスト (host)**」とも呼ばれます。

(3) (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。

**ノードエージェント (node agent)** (n.) Domain Administrator Server をホストするマシンなど、Application Server インスタンスをホストするすべてのマシンに必要な軽量エージェント。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。

**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。

**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。

**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。「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))」も参照してください。

**ハートビート (heartbeat)** (n.) 使用可能なすべてのクラスタインターコネクトトランスポートパス全体に送信される定期メッセージ。特定の間隔が経過し、何回か再試行してもハートビートがない場合は、内部で「フェイルオーバー (failover)」が働いて通信が別のパスに移行します。クラスタメンバーへのすべてのパスが失敗すると、「CMM (クラスタメンバーシップモニター) (CMM (cluster membership monitor))」はクラスタ定足数を再評価します。

**廃止パッチ (obsolete patch)** (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。

**配信 (delivery)** 「メッセージ配信 (message delivery)」を参照してください。

**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。

**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、フェイルオーバー、ラウンドロビンがあります。

**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度かぎりでの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。

**配備 (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、配備シナリオが配備設計に変換され、実装、プロトタイプの作成、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 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がある特定の設定要件を示しています。

**配備後 (postdeployment)** (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。

**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が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。

**配備前 (predeployment)** (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 (logical architecture)**」およびソリューションによって満たす必要がある一連のサービス品質要件に変換されます。

**配布 (distribution)** (n.) さまざまな形式のメディアおよびパッケージテクノロジーで自身を表現するビットの集合。

**配布リスト (distribution list)** 「**メーリングリスト (mailing 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。

**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。

**パススルーサブツリー (pass-through subtree)** (n.) パススルー認証では、PTA Directory Server が、「**DN (識別名) (DN (distinguished name))**」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。

**パススルー認証 (pass-through authentication)** 「**PTA (パススルー認証) (PTA (pass-through authentication))**」を参照してください。

**パスワード認証 (password authentication)** (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 (password policy)** (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。

**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。

**バックアップグループ (backup group)** 「**ネットワークアダプタフェイルオーバーグループ (network adapter failover group)**」を参照してください。

**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 (スリーパスレッド) 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。

**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。「[A レコード \(A record\)](#)」、「[配備 \(deployment\)](#)」も参照してください。

**発行された証明書 (issued certificate)** (n.) 「[認証局 \(certificate authority\)](#)」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。

**パッシベーション (passivation)** (n.) Bean を破棄せずにメモリから 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。

**パッチ (patch)** (n.) ルーチンまたはオブジェクトプログラムに対して即時リリースされた修正プログラム。「[累積パッチ \(accumulated patch\)](#)」、「[廃止パッチ \(obsolete patch\)](#)」、および「[パッチバージョン番号 \(patch version number\)](#)」も参照してください。

**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。

**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。

**バニティドメイン (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。バニティドメインは、MailAlternateAddress 属性を使用して指定されます。バニティドメインのドメイン名には「[LDAP \(ライトウェイトディレクトリアクセスプロトコル\) \(LDAP \(Lightweight Directory Access Protocol\)\)](#)」エントリがありません。バニティドメインは、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。

**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。

**ハブサプライヤ (hub supplier)** (n.) 「[レプリケーション \(replication\)](#)」において、異なるサーバーからコピーされた「[レプリカ \(replica\)](#)」を保持し、その後第三のサーバーにそのレプリカをレプリケートするサーバー。「[カスケード型レプリケーション \(cascading replication\)](#)」も参照してください。

**パブリッシュ / サブスクライブ配信モデル (publish/subscribe delivery model)** (n.) パブリッシャとサブスクライバが通常は匿名で、トピックを動的にパブリッシュまたはサブスクライブできるモデル。システムは、トピックの複数のパブリッシャから到着するメッセージを複数のサブスクライバに配信します。

**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 \(attribute\)](#)」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。

**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。

**汎用リソース (generic resource)** (n.) 汎用リソースタイプの一部として、Resource Group Manager の制御下に置かれたアプリケーションデーモンとその子プロセス。

**汎用リソースタイプ (generic resource type)** (n.) データサービスのテンプレート。汎用リソースタイプを使用して、単純なアプリケーションをフェイルオーバーデータサービスに変え、1 つのノードが停止したときには、別のノードを起動することができます。このタイプでは、SunPlex™ APL によるプログラミングは不要です。

**ピア (peer)** (n.) 別のカテゴリと同じ親カテゴリを持つサブカテゴリ。

**非クラスタモード (non-cluster mode)** (n.) `-x` ブートオプション付きでクラスタメンバーをブートした場合に生じる状態。この状態では、ノードがクラスタメンバーではなくなりますが、クラスタノードとしては残ります。「[クラスタメンバー \(cluster member\)](#)」と「[クラスタノード \(cluster node\)](#)」も参照してください。

**非公開鍵 (private key)** 「[公開鍵暗号方式 \(public key cryptography\)](#)」を参照してください。

**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 \(server\)](#)」になります。

**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、不可欠なビジネスルールを含むアプリケーションコード。

**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。

**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。

**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。

**非配信通知 (nondelivery notification)** 「**NDN (非配信通知) (NDN (nondelivery notification))**」を参照してください。

**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。

**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。

**標準インデックス (standard index)** (n.) デフォルトで維持されるインデックス。

**ブート環境 (boot environment)** (n.) 一連のディスクスライス、関連付けられたマウントポイント、およびファイルシステムで構成される起動可能な Solaris 環境。ディスクスライスは同じディスクにあっても、複数のディスクに分散していてもかまいません。

**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。

**ファイアウォール (firewall)** (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。「**障害モニター (fault monitor)**」も参照してください。

**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) (FTP (file transfer protocol))**」を参照してください。

**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。「**輻輳しきい値 (congestion thresholds)**」も参照してください。

**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。

**ファンシーインデックス (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。

**フィルタ (filter)** (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、ロ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。

(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。

**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。

**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。

**フェイルオーバー (failover)** (1) (n.) Bean がサーバー障害に透過的に耐えられるようにするリカバリプロセス。

(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。

(3) (n.) 障害が発生した後、現在の主ノードから新規の主ノードに、リソースグループまたはデバイスグループを自動的に再配置すること。

**フェイルオーバーリソース (failover resource)** (n.) 所有するリソースが、一度に1つのノードによってのみ正しく制御可能なリソース。「[シングルインスタンスリソース \(single-instance resource\)](#)」と「[スケーラブルリソース \(scalable resource\)](#)」も参照してください。

**フェイルバック (failback)** 「[承認 \(authorization\)](#)」を参照してください。

**フェイルファスト (failfast)** (n.) 障害の発生したノードを、その誤りの可能性がある動作によって実際に損害が生じる前に、順序正しくシャットダウンしてクラスタから取り除くこと。

**フォームアクションハンドラ (form action handler)** (n.) 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する、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。

**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。

**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。

**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。

**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。

**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。

**部分レプリケーション (fractional replication)** (n.) フィルタを適用した属性のサブセットの再生成。

**プライベートホスト名 (private host name)** (n.) 「クラスタインターコネクト (cluster interconnect)」を経由して「ノード (node)」と通信するために使用する「ホスト名 (host name)」エイリアス。

**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。

**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。

**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。たとえば「SQL (構造化クエリー言語) (SQL (structured query language))」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 (シンプル認証とセキュリティ層) (SASL (simple authentication and security layer))」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。

**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。

**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。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。

**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。

**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。

**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 (識別名) (DN (distinguished name))**」。「**プロキシ承認 (proxy authorization)**」で使用されます。

**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 (識別名) (DN (distinguished name))**」はプロキシ DN と呼ばれます。

**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。

**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。「**スレッド (thread)**」も参照してください。

(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。

**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。

**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。

**プロトコル (protocol)** (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。

(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。

**プロバイダ (provider)** (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附属しています。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。

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、JSPProvider、XMLProvider、URLScrapperProvider、および SimpleWebServicesProvider があります。

**プロパティ (property)** (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 つの属性。server.xml ファイルでのプロパティは、名前と値のペアを含む要素です。

**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。

**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またはインターネット環境にまで広がり (分散に関わる部分)、適用範囲と規模が本稼働環境またはサービスプロバイダの必要を満たす (エンタープライズに関わる部分) アプリケーション。

**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。

**分散ロックマネージャー (Distributed Lock Manager)** (n.) 共有ディスク Oracle Parallel Server 環境で使用されるロックングソフトウェア。分散ロックマネージャーにより、異なるノード上で実行されている Oracle プロセスがデータベースアクセスを同期化できるようにします。分散ロックマネージャーは、高可用性の実現を目標として設計されています。プロセスやノードで障害が発生しても、残りのノードをシャットダウンして再起動する必要はありません。分散ロックマネージャーの再設定がすばやく実行され、そうした障害からリカバリします。

**分離脳 (split brain)** (n.) クラスタが複数のパーティションに分離し、各パーティションがほかのパーティションの存在を認識せずに形成される状態。

**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。

**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。

**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 (ディレクトリ情報ツリー) (DIT (directory information tree))」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。

**平文 (cleartext)** (n.) 暗号化されていないテキスト。

**並列サービスインスタンス (parallel service instance)** (n.) 個々のノードで実行されている並列リソースタイプのインスタンス。

**並列リソースタイプ (parallel resource type)** (n.) クラスタ環境で稼働してクラスタ環境を 2 つ以上のノードによって同時に制御できるようにするために利用されている、並列データベースなどのリソースタイプ。

**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:」など名前付きの情報項目。ヘッダー行とも呼ばれます。

**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트리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。

**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 (port)** (n.) 「TCP/IP (トランスミッションコントロールプロトコル / インターネットプロトコル) (TCP/IP (Transmission Control Protocol/Internet Protocol))」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。

**ポート番号 (port number)** (n.) ホストマシンにある個々の「TCP/IP (トランスミッションコントロールプロトコル / インターネットプロトコル) (TCP/IP (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。

**ホームインタフェース (home interface)** (n.) クライアントによるエンタープライズ Bean の作成や削除を可能にするメソッドを定義するメカニズム。

**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트리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。

**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.) プロデューサがメッセージを特定のキューにアドレス指定し、コンシューマが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を取り出す形態の配信モデル。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。

**ホスト (host)** (n.) 1 つ以上のサーバーが置かれているマシン。

**ホスト -IP 認証 (host-IP authentication)** (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メカニズム。



**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) (POP3 (Post Office Protocol Version 3))」を参照してください。

**ホストドメイン (hosted domain)** (n.) ISP にアウトソースされた電子メールアドレスドメイン。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アドレス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。「**ドキュメントタイプ定義**」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。

**ホスト名 (host name)** (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、example.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。

**ホスト名の非表示 (host name hiding)** (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。

**ホップ (hop)** (n.) 2 台のコンピュータ間での送信。

**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せん。

**ボリュームマネージャー (volume manager)** (n.) ディスクのストライピング、連結、ミラーリング、およびメタデバイスまたはボリュームの動的な拡張によってデータ信頼性をもたらすソフトウェア製品。

**本稼働環境 (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。

**本体 (body)** (n.) 電子メールメッセージの一部。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。

**マスター (master)** 「**主 (primary)**」を参照してください。

**マスターエージェント (master agent)** 「**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ネルプログラム。「**スレーブチャネルプログラム (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を含む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。

**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。

**マルチホームホスト (multihome host)** (n.) 複数のパブリックネットワークに存在するホスト。

**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。

**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 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「**データ冗長ユニット (DRU) (data redundancy unit (DRU))**」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。

**メーリングリスト (mailing list)** 「**メーリングリスト (mailing list)**」を参照してください。

**メーリングリスト所有者 (mailing list owner)** 「**メールリスト所有者 (mail list owner)**」を参照してください。

**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) (MX record (mail exchange record))**」を参照してください。

**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.) 「**ユーザーエージェント (user agent)**」または「**MTA (メッセージ転送エージェント) (MTA (message transfer agent))**」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。

**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。

**メタデータ (metadata)** (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。

**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。

**メッセージ (message)** (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。

**メッセージ (messages)** (n.) JMS クライアントがコンシュームする非同期要求、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。

**メッセージ駆動型 Bean (message-driven bean)** 「[MDB \(メッセージ駆動型 Bean\) \(MDB \(message-driven bean\)\)](#)」を参照。

**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。

**メッセージ処理システム (message handling system)** 「[MHS \(メッセージ処理システム\) \(MHS \(message handling system\)\)](#)」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。

**メッセージセクタ (message selector)** (n.) JMS メッセージヘッダーのプロパティ値 (セクタ) 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、メッセージセクタ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。

**メッセージ送信 (message submission)** (n.) クライアントの「**ユーザーエージェント (user agent)**」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 (message forwarding)** (n.) 「**MTA (メッセージ転送エージェント) (MTA (message transfer agent))**」が、特定のアカウントに配信されたメッセージを、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) (MTA (message transfer agent))**」を参照してください。

**メッセージ配信 (message delivery)** (n.) 「**MTA (メッセージ転送エージェント) (MTA (message transfer agent))**」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) (MTA (message transfer agent))**」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。

**メッセージをパージする (purge message)** (v.) 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戻されます。「**メッセージを削除する (delete message)**」と「**メッセージを破棄する (expunge message)**」も参照してください。

**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。

**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。

**メンバー (member)** (n.) メールリスト宛で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。「**メーリングリスト (mailing list)**」、「**展開 (expansion)**」、および「**モデレータ (moderator)**」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。

**モジュール (module)** (1) (n.) アプリケーションの外部に個別に配備された Web アプリケーション、エンタープライズ Bean、メッセージ駆動型 Bean、アプリケーションクライアント、またはコネクタ。「**コンポーネント (component)**」と「**ライフサイクルモジュール (life-cycle module)**」も参照してください。

(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。

**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す。「**メーリングリスト (mailing list)**」、「**展開 (expansion)**」、および「**メンバー (member)**」も参照してください。

**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。

**ユーザー ID (user ID)** (1) (n.) ユーザー識別情報。システムでユーザーを識別するための一意の文字列。userid と呼ばれます。

**ユーザー (user)** (n.) アプリケーションを使用するユーザーまたはサービス。プログラムのには、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。「**グループ (group)**」と「**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트리として管理されます。

**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 と呼ばれます。

**ユーザーエン트리 (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。ユーザープロファイルとも呼ばれます。

**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。

**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、および測定の基礎として使用される、エンドユーザータスクまたは一連のタスク。

**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。

**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、「[JSP™ テクノロジ \(JSP™ technology\)](#)」によって作成されたサ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 ファイルでは、要素が基本構造単位です。XML 要素の中には、サブ要素またはデータがあり、属性を持つこともあります。

**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。

**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。

**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)** 「[LDAP \(ライトウェイトディレクトリアクセスプロトコル\) \(LDAP \(Lightweight Directory Access Protocol\)\)](#)」を参照してください。

**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)** (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。

**リーフエントリ (leaf entry)** (n.) それより下位には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。

**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。

**リソース (resource)** (1) (n.) リソースタイプのインスタンス。多数の同じタイプのリソースが存在する場合があります。それぞれのリソースは固有の名前とプロパティ値のセットを持っているので、基になるアプリケーションの多くのインスタンスをクラスターで実行できます。

(2) (n.) Web ページ、ドキュメント、FTP ディレクトリなど、「[URL \(uniform resource locator\)](#)」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。

(3) (n.) 要求したクライアントにサーバーがアクセスして送信することができる、すべての「[URL \(uniform resource locator\)](#)」、ディレクトリ、またはプログラム。

**リソースオフリング (resource offering)** (n.) ディスカバリサービスでは、リソースオフリングにより、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 \(uniform resource locator\)](#)」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 \(summary object interchange format\)](#)」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。

**リソースグループ (resource group)** (n.) 「[RGM \(リソースグループマネージャー\) \(RGM \(resource group manager\)\)](#)」により 1 単位として管理されるリソースのコレクション。RGM によって管理されるそれぞれのリソースは、リソースグループ内で設定される必要があります。一般には、関連するリソースと相互に依存するリソースごとにグループ化されます。

**リソースグループ状態 (resource group state)** (n.) 任意の指定ノードにあるリソースグループの状態。

**リソースグループマネージャー (resource group manager)** 「[RGM \(リソースグループマネージャー\) \(RGM \(resource group manager\)\)](#)」を参照してください。

**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。

**リソース状態 (resource state)** (n.) 指定ノードにある RGM リソースの状態。

**リソースステータス (resource status)** (n.) 障害モニターによって報告されるリソースの状況。

**リソースタイプ (resource type)** (n.) 「[データサービス \(data service\)](#)」、`LogicalHostname` または `SharedAddress` クラスタオブジェクトに与えられる一意の名前。データサービスのリソースタイプは、[フェイルオーバータイプ](#) か [スケーラブルタイプ](#) にすることができます。「[フェイルオーバーリソース \(failover resource\)](#)」、[「スケーラブルリソース \(scalable resource\)」](#) も参照してください。

**リソースタイププロパティ (resource type property)** (n.) 指定されたタイプのリソースを記述および管理する場合に使用し、リソースタイプの一部として RGM によって保存される鍵と値のペア。

**リソースマネージャー (resource manager)** (n.) データベースやメッセージブローカなどのリソースと、Java Enterprise System Application Server プロセスなどのクライアントの間で、まとめ役となるオブジェクト。グローバルに利用可能なデータソースを制御します。

**リソースモニター (resource monitor)** (n.) リソースタイプ実装のオプション部分。リソースで障害精査を定期的に行って、リソースが正常に動作しているかどうか、またどの程度のパフォーマンスを示しているかについて判断します。

**リソース呼び出し (resource invocation)** (n.) 1 つのノードで動作してい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概念です。

**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、後続のフ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。

**リバース DNS 検索 (reverse DNS lookup)** (n.) 「[DNS \(ドメインネームシステム\) \(DNS domain name system\)](#)」のクエリを行なって、数値「[IP アドレス \(IP address\)](#)」を同等の「[完全修飾ドメイン名 \(fully qualified domain name\)](#)」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。

**リファレンス配備アーキテクチャ (reference deployment architecture)** (n.) パフォーマンスが設計、実装、およびテストされている「[配備アーキテクチャ \(deployment architecture\)](#)」。基準配備アーキテクチャは、カスタムソリューションの配備アーキテクチャを設計する場合の開始点として使用されます。

**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 へのポイント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。

**リモートインタフェース (remote interface)** (n.) Enterprise JavaBeans™ コンポーネントの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。

**リライタ (Rewriter)** (n.) リライタは、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。リライタは、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。またリライタには、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。

**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。

**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。「[完全修飾ドメイン名 \(fully qualified domain name\)](#)」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。

**ルートエントリ (root entry)** (n.) 「[DIT \(ディレクトリ情報ツリー\) \(DIT \(directory information tree\)\)](#)」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。

**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。

**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。

**列 (column)** (n.) データベーステーブル内のフィールド。

**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。

**レプリカ (replica)** (n.) サーバーの「**レプリケーション (replication)**」領域のインスタンス。「**コンシューマレプリカ (consumer replica)**」と「**サプライヤレプリカ (supplier replica)**」も参照してください。

**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 (識別名) (DN (distinguished name))**」と資格、および接続をセキュリティ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 (識別名) (DN (distinguished name))**」。

**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 台のサーバー間で開始されるセッション。

**レベル (level)** (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、Emergency レベルでは、ログに記録されるイベントはほとんどありません。情報レベルでは、多くのイベントがログに記録されます。

**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。J2EE™ 仕様では、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。

**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。

**連携アイデンティティ (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** とも呼ばれます。連携サーバーそれぞれにはシンボリック名が与えられます。シンボリック名は、文字と数字で構成される **coservern** などの文字列で、*n* は番号を表します。

**連携の解除 (defederation)** (n.) 「[連携の終了 \(federation termination\)](#)」を参照してください。

**連携の終了 (federation termination)** (n.) ユーザーのアイデンティティ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。

**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。

**連鎖 (chaining)** (n.) 別のサーバーに要求を中継する手段。要求の結果は、収集およびコンパイルされてからクライアントに返されます。レプリケーションにおいて、コンシューマレプリカが更新要求を受け取り、対応するマスターレプリカを保持するサーバーにその要求を転送するときに、連鎖が発生します。このプロセスはリフェラルとは異なります。「[連鎖サフィックス \(chained suffix\)](#)」も参照してください。

**連鎖サフィックス (chained suffix)** (n.) 連載の実装。連鎖サフィックスは、通常のサフィックスのように振舞いますが、持続的記憶領域はありません。代わりに、連鎖サフィックスはリモートに保存されたデータをポイントします。「[連鎖 \(chaining\)](#)」も参照してください。

**レンダリング (rendering)** (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。

**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。

**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。

**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。

**ローカルディスク (local disk)** (n.) 所定のクラスタノードに対して物理的にプライベートなディスク。

**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 を使って区別されます。「[グローバルトランザクション \(global transaction\)](#)」、「[JDBC™ テクノロジ \(Java™ DataBase Connectivity ソフトウェア\) \(JDBC™ technology \(Java™ DataBase Connectivity software\)\)](#)」も参照してください。

**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 \(ローカルメール転送プロトコル\) \(LMTP \(Local Mail Transfer Protocol\)\)](#)」を参照してください。

**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。



**ロードバランシング (load balancing)** (n.) クラスタ内のノード全体にアプリケーションの負荷を分散し、クライアントの要求が適宜処理されるようにするプロセス。スケーラブルなサービスにのみ適用されます。

**ロードバランスポリシー (load balancing policy)** (n.) アプリケーションの要求負荷をノード全体で分散するのに好ましい方法。スケーラブルなサービスにのみ適用されます。

**ロール (role)** (1) (n.) 配備環境で、1 つ以上のグループによって表されるアプリケーション内のサブジェクトの機能的グループ化。「ユーザー (user)」、「グループ (group)」も参照してください。

(2) (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) (ACI (access control instruction))」) を使用して定義します。

(3) (n.) エントリのグループ化のメカニズム。各ロールには、そのロールを所有するエントリであるメンバーが存在します。

**ロールに基づく属性 (role-based attribute)** (n.) エントリが関連付けられた「CoS (サービスクラス) (CoS (class of service))」 テンプレート内に特定のロールを所有しているために、エントリに表示される属性。

**ロールバック (rollback)** (n.) 「トランザクション (transaction)」の取り消し。

**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。

**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。

**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) (RAF (robot application function))**」を参照してください。

**論理アーキテクチャ (logical architecture)** (n.) 分散アプリケーションの論理的構成要素およびそれらの構成要素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サービスのコンポーネントの両方が含まれます。

**論理ネットワークインタフェース (logical network interface)** (n.) インターネットのアーキテクチャでは、ホストが1つ以上のIPアドレスを持つことができます。**Messaging Server** は、追加の論理ネットワークインタフェースを設定して、数個の論理ネットワークインタフェースと1つの物理ネットワークインタフェースの間でマッピングを確立します。各論理ネットワークインタフェースには、IPアドレスが1つ割り当てられます。このマッピングにより、1つの物理ネットワークインタフェースで、複数のIPアドレスに応答することができます。また、引継ぎや切り替えのイベント時には、追加のハードウェアインタフェースを必要とせず、IPアドレスを1つのクラスタメンバーからほかのメンバーに移すこともできます。

**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。

**論理ホスト名 (logical host name)** (n.) ネットワークアドレスを表す論理ホスト名の集合を収めたリソース。論理ホスト名リソースは、一度に1つのノードによってのみ制御できます。「**論理ホスト (logical host)**」も参照してください。

**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。

**ワイヤレスデスクトップディスパッチャ (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。

**ワイルドカード (wildcard)** (n.) 1つまたは複数のほかの文字または文字範囲を表すことができる検索文字列内の特殊文字。

