



Sun Java™ System

Sun Java Enterprise System

术语表

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

文件号码 819-1935

版权所有 © 2005 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. 保留所有权利。

对于本档中介绍的产品，Sun Microsystems, Inc. 对其所涉及的技术拥有相关的知识产权。需特别指出的是（但不局限于此），这些知识产权可能包含在 <http://www.sun.com/patents> 中列出的一项或多项美国专利，以及在美国和其他国家/地区申请的一项或多项其他专利或待批专利。

本产品含有 SUN MICROSYSTEMS, INC. 的机密信息和商业秘密。未经 SUN MICROSYSTEMS, INC. 的事先明确书面许可，不得使用、泄露或复制。

美国政府权利 - 商业软件。政府用户应遵循 Sun Microsystems, Inc. 的标准许可协议，以及 FAR（Federal Acquisition Regulations，即“联邦政府采购法规”）的适用条款及其补充条款。

本发行版可能包含由第三方开发的内容。

本产品的某些部分可能是从 Berkeley BSD 系统衍生出来的，并获得了加利福尼亚大学的许可。UNIX 是 X/Open Company, Ltd. 在美国和其他国家/地区独家许可的注册商标。

Sun、Sun Microsystems、Sun 徽标、Java、Solaris、JDK、Java Naming and Directory Interface、JavaMail、JavaHelp、J2SE、iPlanet、Duke 徽标、Java 咖啡杯徽标、Solaris 徽标、SunTone Certified 徽标以及 Sun ONE 徽标是 Sun Microsystems, Inc. 在美国和其他国家/地区的商标或注册商标。

所有 SPARC 商标的使用均已获得许可，它们是 SPARC International, Inc. 在美国和其他国家/地区的商标或注册商标。标有 SPARC 商标的产品均基于由 Sun Microsystems, Inc. 开发的体系结构。

Legato 和 Legato 徽标是 Legato Systems, Inc. 的注册商标，Legato NetWorker 是其商标或注册商标。Netscape Communications Corp 徽标是 Netscape Communications Corporation 的商标或注册商标。

OPEN LOOK 和 Sun(TM) 图形用户界面是 Sun Microsystems, Inc. 为其用户和许可证持有者开发的。Sun 感谢 Xerox 在研究和开发可视或图形用户界面的概念方面为计算机行业所做的开拓性贡献。Sun 已从 Xerox 获得了对 Xerox 图形用户界面的非独占性许可证，该许可证还适用于实现 OPEN LOOK GUI 和在其他方面遵守 Sun 书面许可协议的 Sun 许可证持有者。

本服务手册所介绍的产品以及所包含的信息受美国出口控制法制约，并应遵守其他国家/地区的进出口法律。严禁将本产品直接或间接地用于核设施、导弹、生化武器或海上核设施，也不能直接或间接地出口给核设施、导弹、生化武器或海上核设施的最终用户。严禁出口或转口到美国禁运的国家/地区以及美国禁止出口清单中所包含的实体，包括但不限于被禁止的个人以及特别指定的国家/地区的公民。

本文档按“原样”提供，对于所有明示或默示的条件、陈述和担保，包括对适销性、适用性和非侵权性的默示保证，均不承担任何责任，除非此免责声明的适用范围在法律上无效。

Sun Java Enterprise System 术语表

本术语表确定 Java Enterprise System 术语及定义。

access control (访问控制) 通过对服务器的访问加以控制来保证服务器安全的方法。

access control entry (访问控制条目) 参见 [ACE](#)。

access control instruction (访问控制指令) 参见 [ACI](#)。

access control list (访问控制列表) 参见 [ACL](#)。

access control rules (访问控制规则) 为给定的一组目录条目或目录属性指定用户权限的规则。

access domain (访问域) 限制从某一指定域内访问特定 Messaging Server 操作的域。例如，访问域可用于限制可在何处收集帐户的电子邮件。

access rights (访问权限) 访问权限指定授予或拒绝的访问控制级别。访问权限与可在目录上执行的操作类型相关。可授予或拒绝下列权限：读、写、添加、删除、搜索、比较、自写、代理和全部。

account (帐户) 定义特定用户或用户组的信息。此信息包括用户名或组名、一个或多个有效电子邮件地址以及传送电子邮件的方式和位置。

account inactivation (帐户去活) 禁用单个用户帐户或一组帐户，这样便可自动拒绝所有验证尝试。

ACE（**access control entry**，访问控制条目）(1) 访问控制列表中的一个信息项。也称为访问控制信息。

(2) Web 服务器用于评估收到的访问请求的规则层级结构。

(3) 为日历、日历属性和日历组件（如事件和任务）提供访问控制的字符串。

ACI（**access control instruction**，访问控制指令）该指令将准许或拒绝访问目录中的条目。

ACL（**access control list**，访问控制列表）(1) 控制对目录的访问的机制。在 Directory Server 中，ACL 为目录条目中的一个 ACI 属性。

(2) ACE 集合。ACL 是定义哪些用户可访问您的服务器的一种机制。您可以定义特定于某个特殊文件或目录的 ACL 规则，准许或拒绝访问一个或多个用户和组。

(3) 共同为日历、日历属性和日历组件（如事件和任务）提供访问控制的一组 ACE 字符串。

(4) 与目录相关的一组数据，它们定义用户、组或用户以及组具有的访问该目录的权限。ACL 由一个或多个 ACE 字符串组成。

account federation（**帐户联合**） 用户选择将不同服务提供者帐户与身份认证提供者帐户联合起来时所发生的过程。用户保留在每个提供者处的个人帐户信息，同时建立一个允许在提供者帐户之间交换验证信息的链接。也称为身份联合。

accumulated patch（**累积修补程序**） 一个修补程序，该程序结合了先前的一个或多个修补程序中的补丁、相同修补程序的任意先前版本中的补丁和将要发布的当前补丁集。

activation（**激活**） 将企业 bean 的状态从辅助存储器转移到内存的过程。

active boot environment（**活动引导环境**） 当前已开启并运行的环境。

active node（**活动节点**） 包含会话数据的 HADB 节点。如果活动节点出现故障，备用节点会从镜像节点复制数据并转为活动状态。另参见 [HADB node](#)（**HADB 节点**）、[spare node](#)（**备用节点**）、[mirror node](#)（**镜像节点**）和 [data redundancy unit](#)（**数据冗余单元**）。

address (地址) (1) 电子邮件消息中确定邮件发送方式和位置的信息。消息头和消息信封中都包含地址。信封地址确定如何路由和传送消息。头地址仅用于显示。

(2) 在联网技术领域内标识网络中某一节点的唯一代码。类似 `example.corp.com` 这样的名称被 DNS 转换为“点分”形式的地址 (168.124.0.0)。

address handling (地址处理) 由 MTA 执行的操作，其目的是检测寻址过程中的错误、必要时重写地址以及将地址与收件人对应。

addressing protocol (寻址协议) 使电子邮件可以传送的寻址规则。RFC 822 是在 Internet 上应用最广的协议，也是 Messaging Server 支持的协议。其他协议还有 X.400 和 UUCP。

address token (地址令牌) 重写规则模式的地址元素。

admin console (管理控制台) 管理员与 Portal Server 6.0 连接的 Directory Server Access Management Edition GUI 界面。

administered object (受管对象) 由管理员创建的预配置 Java Enterprise System 对象（连接工厂或目标），由一个或多个 JMS 客户机使用。

使用受管对象可将 Java™ 消息服务 (Java™ Message Service, JMS) 客户机从提供者的专有状态中分离出来。这些对象由管理员放置在一个 Java Naming and Directory Interface™ (JNDI) 名称空间中，由 JMS 客户机使用 JNDI 查找进行访问。

administration console (管理控制台) 参见 [console \(控制台\)](#)。

administration domain (管理域) 管理控制区域。另参见 [document type definition \(文档类型定义\)](#)。

administration interface (管理界面) 基于浏览器的表单集，用于配置和管理 Java™ Enterprise System 服务器。另参见 [CLI](#)。

administration privileges (管理权限) 定义用户管理角色的一组权限。

administration server (管理服务器) 专用于提供 Java Enterprise System Directory Server 管理功能的应用服务器实例。

administration server administrator（**管理服务器管理员**） 具有启动或停止服务器（即使没有 Java Enterprise System Directory Server 连接）的管理权限的用户。管理服务器管理员承担本地服务器组中所有服务器的有限服务器任务（通常只是“重新启动服务器”和“停止服务器”）。安装管理服务器时，此管理员的条目自动在本地创建。此管理员不是用户目录中的用户。

administrative console（**管理控制台**） 用于运行群集管理软件的工作站。

administrative domain（**管理域**） Java Enterprise System Application Server 的一个功能，允许不同的管理用户创建并管理他们自己的域。域是在单个系统中使用已安装的一组公用二进制文件创建的一组实例。另参见 **domain**（**域**）。

administrator（**管理员**） 具有定义的一组管理权限的用户。另参见 **configuration administrator**（**配置管理员**）、**Directory Manager**（**目录管理员**）、**administration server**（**管理服务器**）、**server administrator**（**服务器管理员**）、**top-level administrator**（**顶级管理员**）、**domain administrator**（**域管理员**）、**organization administrator**（**组织管理员**）、**family group administrator**（**系列组管理员**）、**mail list owner**（**邮寄列表所有者**）。

admpw Sun Enterprise™ Administrator Server 超级用户的用户名和密码文件。

adoption scenario（**应用情况**） 部署 Java Enterprise System 软件的总体因素，描述您的初始软件系统的特征以及试图达到的目标。Java Enterprise System 有四种基本的应用情况：新系统、更改、扩展和升级。

affiliation（**联合提供者**） 联合提供者是一组提供者，当他们组成联合提供者时，不考虑他们的特殊验证域。联合提供者由联合提供者属主形成并维护。联合提供者文档描述由各提供者的 **providerID** 共同标识的一组提供者。联合提供者的成员可以通过成员身份（利用其“联合提供者 ID”）调用服务，也可以单独（利用其“提供者 ID”）调用服务。

agent（**代理**） 在网络设备（如路由器、主机或 X 终端）中运行网络管理软件的软件。另参见 **intelligent agent**（**智能代理**）。

alarm event（**报警事件**） 由 Calendar Server ENS 生成并发送的事件。发生报警事件时，会向特定收件人发送一个消息提示。

alert（警报） 用户在弹出窗口即时收到的时间关键消息。只要使用“Show message status”（显示消息状态）选项，发件人便可在关闭或单击警报时知道谁接收了消息并得到消息已被阅读的通知。如果该警报消息要求响应，可右键单击警报以调用一个上下文菜单，其中有“Chat with Sender”（与发件人聊天）选项。

alias（别名） 电子邮件地址的一个备用名称。

alias file（别名文件） 用于设置未在目录中设置的别名（如邮寄主管别名）的文件。

aliasing（别名设置） 在“Java Enterprise System Portal Server 搜索引擎”中用一个项目替代另一项目，从具有不同模式的其他“搜索引擎”导入资源描述时，该搜索引擎使用别名设置。

All IDs threshold（全部 ID 阈值） 全局应用于 Java Enterprise System Directory Server 管理的每个索引的大小限制。当某个条目 ID 列表达到此限制值时，服务器用一个“全部 ID”令牌替换该条目 ID 列表。

All IDs token（全部 ID 令牌） 致使服务器假定所有目录条目都与索引键匹配的一种机制。实际上，“全部 ID”令牌会使 Java Enterprise System Directory Server 执行无索引搜索来匹配索引键。

allowed attributes（允许的属性） 可存在于使用特定对象类的条目中的可选属性。另参见 [attribute（属性）](#)、[required attributes（必需属性）](#)。

Allow filter（允许过滤器） Java Enterprise System Messaging Server 访问控制规则，用于确定允许访问一个或多个 POP、IMAP 或 HTTP 服务的客户机。另参见 [deny filter（拒绝过滤器）](#)。

alternate address（备用地址） 某帐户的辅助地址，通常是主地址的变型。在某些情况下，一个帐户有多个地址会更方便一些。

alternate root（备用根） 客户机上 root 文件系统中安装某一软件包的位置。备用根一般是使用 `pkgadd -R` 提供的。

AML（abstract markup language，抽象标记语言） 与具体供应商或型号无关的移动设备标记语言。

amnesia (失忆) 含有陈旧 **category (类别)** 的群集在关闭之后重新启动时出现的情况。例如，在只有节点 1 运行的两节点群集上，如果节点 1 的群集配置更改，则节点 2 的 CCR 变为陈旧。如果关闭该群集然后在节点 2 上重新启动，则由于节点 2 的 CCR 已陈旧，导致出现失忆的情况。

anonymous access (匿名访问) 此种访问经授予后允许任何人不提供凭证即可访问目录信息，无论绑定情况如何。

API (application programming interface, 应用编程接口) (1) 一组指令，计算机程序可使用这些指令与专用于解释该 API 的其他软件或硬件通信。

(2) 一组调用约定或指令，定义程序如何调用现有软件包中的服务。

APOP (authenticated post office protocol, 验证邮局协议) 与 POP 类似，但 APOP 不使用纯文本密码进行验证，而是将密码编码与一个质询字符串一起使用。

applet 使用 Java™ 编程语言编写的、在 Web 浏览器上运行的小型应用程序。applet 通常由 Web 页面调用或者嵌入 Web 页面以提供特殊功能。Java applet 是使用 Java™ 编程语言编写的一种小型应用程序，可以与 Web 页面一同发送到用户的浏览器。Java applet 不必将用户请求发送回服务器便可执行任务。Instant Messaging 客户机即是一种 Java applet。另参见 *serolet*。

application client container (应用程序客户机容器) 参见 **container (容器)**。

application component (应用程序组件) 自定义开发的执行某些特定计算功能的软件 **component (组件)**，为 **end user (最终用户)** 或其他应用程序组件提供 **business service (业务服务)**。应用程序组件通常符合分布式组件模型（如 CORBA 和 J2EE™ 平台）。这些组件（无论单独还是组合在一起）都可被封装为 **web service (web 服务)**。

application server (应用服务器) 运行业务应用程序的一种软件平台。应用服务器通常为应用程序提供高级别服务，如组件生命周期、位置以及分发和事务资源访问。

application service (应用程序服务) 组件或组件的集合体，代表多个客户机执行业务逻辑，因而必须是一个多线程进程。应用程序服务也可以是封装为 Web 服务或独立的内容服务器的组件或组件集合体。

application tier (应用程序层次) 对 J2EE 应用程序在概念上的划分：

客户层： 用户界面。最终用户与客户机软件（如 Web 浏览器）交互，以使用应用程序。

服务器层： 组成您的应用程序的业务逻辑和表示逻辑，在应用程序的组件中定义。

数据层： 使您的应用程序能够与数据源交互的数据访问逻辑。

approximate index (近似索引) 此种索引允许在目录信息树中进行有效的近似或“音近”搜索。

architecture (体系结构) 一种设计，展示了分布式应用程序（或其他某个软件系统）的逻辑和物理构件及其相互关系。对于 **distributed enterprise application (分布式企业应用程序)** 而言，体系结构设计通常同时包括应用程序的 **logical architecture (逻辑体系结构)** 和 **deployment architecture (部署体系结构)**。

A record (A 记录) 一种包含主机名及其相关 IP 地址的 DNS 记录。A 记录由 Internet 上的讯息传递服务器用于为电子邮件选择路由。另参见 **domain name system (域名系统)**、**MX record (MX 记录)**。

assembly (装配) 将某一应用程序的分散组件组合到可以部署的单个单元中的过程。另参见 **deployment (部署)**。

asynchronous communication (异步通信) 一种通信模式，在该模式下，消息的发送者不需要等待发送方法返回即可继续其他工作。

attribute (属性) (1) 请求对象中可由 servlet 设置的名称/值对。也可以是修改 XML 文件中某一元素的名称/值对。另参见 **parameter (参数)**。属性更通常指一个元数据单位。

(2) 保存有关某个条目的描述信息的名称/值对。属性具有类型（名称）和一组值。属性类型还为可以存储为该类属性值的信息种类进行了语法规定。

(3) 定义 Java Enterprise System Directory Server Access Management Edition 服务为组织提供的参数。组成 Java Enterprise System Directory Server Access Management Edition 服务的属性可以归为以下任一类别：动态、策略、用户、组织或全局。使用这些类型对每种服务中的属性进行细分，可以使服务模式管理更一致并可简化服务参数的管理。

attribute provider (属性提供者) 属性提供者是托管属性数据的一种 Web 服务。

attribute list (属性列表) 参见 [optional attribute list \(可选属性列表\)](#) 和 [request object \(请求对象\)](#)。

auditing (审计) 记录重要事件以备后续检查（通常在出错或安全性被破坏的情况下进行）的方法。

AUTH 一个 SMTP 命令，使 SMTP 客户机能够指定对服务器的验证方法、执行验证协议交换、在必要时对后续协议交互的安全层进行协商。

authenticating Directory Server (验证 Directory Server) 在 PTA 中，验证 Java Enterprise System Directory Server 中包含请求方客户机的验证凭证。启用 PTA 的用户目录将绑定请求传递给验证目录，该目录检验请求方客户机的绑定凭证。

authentication (验证) (1) 确定某人或某事是否与其声明身份或情况相符的过程。在专用或公共计算机网络（包括 Internet）中，验证通常使用登录密码完成。知道密码的用户即被视为可信用户。

(2) 为 Java Enterprise System Directory Server 验证客户机用户身份的过程。用户必须提供一个绑定 DN 和相应的密码或证书，才能被授权访问目录。Java Enterprise System Directory Server 允许用户在经目录管理员授予权限后执行功能或访问文件和目录。另参见 [server authentication \(服务器验证\)](#)。

authentication certificate (验证证书) 从服务器发送到客户机或从客户机发送到服务器以对另一方进行检验或验证的数字文件。证书确保其持有者（客户机或服务器）的真实性。证书不可转让。

authentication domain (验证域) 一组服务提供者，他们至少具有一个同意使用“自由联盟计划”(Liberty Alliance Project, LAP) 交换用户验证信息的身份认证提供者。一旦建立了 **circle of trust (信任环)**，即在所有提供者之间启用了单点登录验证。也称为信任环。

authorization (授权) 为用户授予特定访问权限的过程。授权基于验证，而且由访问控制强制执行。

automatic failback (自动故障恢复) 在主节点发生故障而后又作为群集成员重新启动之后，将资源组或设备组返回其主节点的过程。另参见 **switchback (切换回原状态)**。

autoreply option file (自动回复选项文件) 用于设置电子邮件自动回复选项（如休假通知）的文件。

AutoReply utility (AutoReply 实用程序) 对发送给激活了 AutoReply 功能的帐户的消息自动响应的实用程序。Java Enterprise System Messaging Server 中的每个帐户都可配置为自动回复收到的消息。

availability service (可用性服务) Sun Java System Application Server 的一种功能，用于在服务器实例、Web 容器、EJB 容器上启用高可用性，也用于为 RMI/IIOP 请求启用高可用性。

backbone (主干) 分布式系统的主要连通机制。与主干上的一个中间系统连接的所有系统都彼此连接。主干不会妨碍您出于成本、性能或安全原因而将系统设置为绕过主干。

back-end server (后端服务器) 指 Java Enterprise System Messaging Server 环境中只具备存储和检索电子邮件消息功能的电子邮件服务器。也称为消息存储服务器。

backout (回退) 将软件更改（如修补程序）删除，从而使系统返回先前状态。

back up (备份) 将消息存储服务器中的文件夹内容复制到备份设备。另参见 **restore (恢复)**。

backup group (备份组) 参见 **network adapter failover group (网络适配器故障转移组)**。

backup store (备份存储) 数据的信息库，通常为文件系统或数据库。可由后台线程 (sweeper 线程) 监视备份存储以便移除不需要的条目。

banner (标题) 客户机首次连接到某一服务 (如 IMAP) 时该服务显示的文本字符串。

base DN (基 DN) (base distinguished name, 基标识名) DIT 中的条目。对于由基 DN 标识的条目、直接下属于该基 DN 或该条目的条目以及 *DIT* 中该基 DN 之下的所有条目，都可执行搜索操作。

bean-managed persistence (bean 管理持久性) 实体 bean 的变量与数据存储之间的数据传送。数据访问逻辑通常由开发者使用 Java™ 数据库连接 (Java™ Database Connectivity, JDBC™) 软件或其他数据访问技术提供。另参见 [container-managed persistence \(容器管理持久性\)](#)。

bean-managed transaction (bean 管理事务) 其中，企业 bean 的事务划分由开发者通过程序控制。另参见 [container-managed transaction \(容器管理事务\)](#)。

Berkeley DB (Berkeley database, Berkeley 数据库) 适用于高并发性读写工作负载以及需要事务和可恢复性的应用程序的事务数据库存储。Java Enterprise System Messaging Server 将 Berkeley 数据库用于许多用途。

bind DN (绑定 DN) (bind distinguished name, 绑定标识名) 绑定请求中用于对 Java Enterprise System Directory Server 进行验证的标识名。

bind rule (绑定规则) 在访问控制环境中，绑定规则指定特定用户或客户机要访问目录信息所必须具备的凭证和满足的条件。

BLOB (binary large object, 二进制大对象) 用于存储和检索复杂对象字段的一种数据类型。BLOB 是二进制或可序列化对象 (如图片)，转换后为大字节数组，随后序列化为容器管理持久性字段。

BMP 参见 [bean-managed persistence \(bean 管理持久性\)](#)。

BMT 参见 [bean-managed transaction \(bean 管理事务\)](#)。

body (正文) 电子邮件消息的一部分。尽管报头和信封必须遵循标准格式，但消息正文的内容可由发件人决定。正文可包含文本、图形或多媒体。结构化正文遵循 MIME 标准。

boot environment (引导环境) 可引导的 Solaris 环境，由一组磁盘分片、相关的安装点和文件系统组成。各磁盘分片可在同一磁盘上，也可分布于多个磁盘。

broker (代理) 管理 Java™ 消息服务 (Java™ Message Service, JMS) API 消息路由选择、传送、持久性、安全性和登录的 Message Queue 实体。代理提供了一个接口，管理员可使用该接口监视并调优性能和资源的使用。

browsing (浏览) 在 Java Enterprise System Portal Server 内，指在某一“搜索”数据库中查找按类别划分的资源。

browsing index (浏览索引) 参见 [virtual list view index \(虚拟列表视图索引\)](#)。

building module (构建模块) 不依赖或部分依赖共享服务的硬件或软件结构。是一种可提供最佳性能和水平可伸缩性的特定配置。

business logic (业务逻辑) 实现某一应用程序（而非数据集成或表示逻辑）的基本业务规则的代码。

business service (业务服务) [application component \(应用程序组件\)](#) 或组件集合体，代表多个客户机执行业务逻辑（因而是一个多线程进程）。业务服务也可以作为 [web service \(web 服务\)](#) 封装起来的分布式组件的集合体，还可以是独立的 [server \(服务器\)](#)。

CA (certificate authority, 证书授权机构) (1) 签发用于加密事务的数字文件的内部组织或第三方组织。

(2) 网络中签发和管理安全凭证和用于消息加密和解密的公共密钥的授权机构。作为 PKI 的一部分，CA 与注册授权机构核对数字证书申请者提供的信息，以进行检验。如果注册授权机构验证了申请者的信息，CA 则可以签发证书。另参见 [PKI](#)。

cache (高速缓存) 本地存储的原始数据的副本。请求使用缓存的数据时，不必再从远程服务器检索。

Cache Control Directive (高速缓存控制指令) Java Enterprise System Application Server 控制代理服务器缓存哪些信息的一种方法。您可使用缓存控制指令覆盖代理的默认缓存, 避免缓存敏感信息, 从而防止以后被检索。要使这些指令起作用, 代理服务器必须与 HTTP 1.1 兼容。

cached rowset (缓存行集) 一种对象, 允许您从某数据源检索数据, 然后在检查和修改数据时与该数据源分离。缓存行集既跟踪检索的原始数据, 也跟踪应用程序对原始数据所做的任何更改。如果应用程序尝试更新原始数据源, 则此行集被重新连接到该数据源, 而且只有那些发生更改的行被合并回数据库中。

calendar access protocol (日历访问协议) 参见 [CAP](#)。

Calendar Express 一种基于 Web 的日历客户机程序, 为最终用户提供对 Calendar Server 的访问权。

calendar group (日历组) 若干日历的集合, 可帮助用户管理多个日历。

calendar ID (日历 ID) 与 Java Enterprise System Calendar Server 数据库中的某一日历关联的唯一标识符。也称为 `calid`。

calendar lookup database (日历查找数据库) 参见 [CLD](#)。

Calendar Server application programming interface (Calendar Server 应用程序接口) 参见 [CSAPI](#)。

calendar user agent (日历用户代理) 参见 [CUA](#)。

callable statement (可调用语句) 一个类, 对于支持从存储过程返回结果集的数据库, 该类可封装数据库的过程和函数调用。

CAP (calendar access protocol, 日历访问协议) 一种标准的 Internet 日历协议, 基于 “Internet 工程任务组” (Internet Engineering Task Force, IETF) 确定的要求。

capability (能力) 为客户机提供的字符串, 它定义某一给定 IMAP 服务中的可用功能。

cascading replication (级联复制) 在级联复制方案中，一台服务器（通常称为集线服务供应者）既是特定副本的使用者同时又是供应者。该服务器保留一份只读副本并维护更改日志。它从持有原版数据的供应者服务器接收更新内容，然后再将这些更新提供给使用者。

catalog (目录) 参见 [index \(索引\)](#)。

cataloging (编目) 参见 [indexing \(建立索引\)](#)。

category (类别) “搜索”数据库内资源的逻辑分组。概括而言，一组类别有时也称为“分类系统”。

CCPP (composite capability and preference profiles, 功能及首选项综合配置文件) 对于 Portal Server Mobile Access 软件来说，它是用于“用户代理配置文件”的规范以及用于客户机检测的预配置数据。CCPP 规范描述了设备的功能和用户首选项。

CCR (cluster configuration repository, 群集配置系统信息库) Sun™ Cluster 软件用来永久性存储群集配置信息的、具有高可用性和重复性的数据存储。

cHTML 适用于移动设备的简化版 HTML。

certificate (证书) (1) 用于标识 Instant Messaging Server 的电子文档，与公共密钥关联在一起。Java Enterprise System Instant Messaging Server 支持 Instant Messaging 服务器之间交换证书。证书交换对于单个用户来说是透明的。

(2) 一种数字数据，指定个人、公司或其他实体的名称并证明证书内所含公共密钥属于该实体。客户机和服务器都可拥有证书。

(3) 证书将用户或 CA 的公共密钥与该用户或 CA 的身份（通常为标识名）紧密关联。证书由 CA 以数字化方式签署，并可在建立 SSL 连接期间验证，以获得连接另一端的公共密钥。X.509 证书存储在 `caCertificate;binary` 或 `userCertificate;binary` 属性中的目录中。

(certificate authority) (证书授权机构) 参见 [CA](#)。

certificate-based authentication (基于证书的验证) 根据客户机提交的数字证书确认用户的身份。另参见 [password authentication \(密码验证\)](#)。

certificate database（**证书数据库**） 包含服务器的数字证书或证书的文件。也叫做证书文件。

certificate name（**证书名称**） 标识证书及其所有者的名称。

certificate revocation list（**证书撤销列表**） 参见 [CRL](#)。

CGI（**common gateway interface**，公共网关接口）外部程序与 HTTP 服务器通信所用接口。使用 CGI 的程序被称为 CGI 程序或 CGI 脚本。CGI 程序处理的表单或解析的输出通常是服务器无法处理或解析的。

chained suffix（**链接后缀**） 一种实现链接的方式。链接后缀的特性与常规后缀相似，但没有永久存储区。相反，链后缀指向远程存储的数据。另参见 [chaining](#)（[链接](#)）。

chaining（**链接**） 将请求中转到另一服务器上的一种方法。请求的结果将被收集、编译，然后返回客户机。在复制条件下，链接发生于使用者副本收到更新请求并将此请求转发给持有相应主副本的服务器时。注意，此过程与引用不相同。另参见 [chained suffix](#)（[链接后缀](#)）。

change log（**更改日志**） 更改日志是对副本所作修改的记录。然后供应者服务器将在存储于使用者服务器上的副本上回放这些修改，或者在多主副本复制情况下，在其他主副本上回放这些修改。注意，它与逆向更改日志不同，后者不用于复制。

channel（**通道或频道**） (1) 处理消息的基本 MTA 组件。通道代表与另一计算机系统或一组系统的连接。每个通道由一个或多个通道程序和一个外出的消息队列组成，该队列用于存储将要发送给与该通道相关联的一个或多个系统的消息。另参见 [channel block](#)（[通道块](#)）、[channel host table](#)（[通道主机表](#)）、[channel program](#)（[通道程序](#)）。

(2) 在 Java Enterprise System Portal Server Desktop 中，频道由提供者和配置组成。频道生成的内容可包括标记片段、框架集、HTML 页面等。频道内容常常与其他频道内容汇集，形成“Portal 桌面”。

channel block（**通道块**） 单一通道定义。另参见 [channel host table](#)（[通道主机表](#)）。

channel host table（**通道主机表**） 通道定义的集合。另参见 [channel block](#)（[通道块](#)）。

channel program (通道程序) 通道的一部分，负责将消息发送到远程系统，并在发送后从队列中删除消息，以及从远程系统接收消息并将它们放置到适当的通道队列中。另参见 [master channel program \(主通道程序\)](#)、[slave channel program \(从通道程序\)](#)。

character type (字符类型) 用于区分字母型字符与数值或其他类型字符的属性，以及大小写字母映射。

chat (聊天) Instant Messaging 的即时讯息传递版本。“聊天”是一项实时会话功能。“聊天”会话可以在根据需要随时创建的聊天室或预先建好的会议室内进行。

checkpoint (检查点) 主节点发送给辅助节点的通知，以使他们之间的软件状态保持同步。另参见 [primary \(主节点\)](#) 和 [secondary \(辅助节点\)](#)。

child (子项) 其他类别的子类别。另参见 [category \(类别\)](#)。

chroot 可以创建的附加根目录，用于将服务器限制在特定目录内。可以使用此功能来保护未受保护的服务器。

cipher (加密器) 加密器是用于加密或解密的加密算法（数学函数）。

ciphertext (密文) 没有正确密钥解密信息时任何人都无法读取的加密信息。

circle of trust (信任环) 参见 [authentication domain \(验证域\)](#)。

CKL (compromised key list, 已泄密密钥列表) 拥有已泄密密钥的用户的密钥信息列表。CA 也提供此列表。“已泄密密钥列表”一词的附加定义。另参见 [CRL](#)。

class definition (类定义) 指定创建特定对象实例所需信息的定义。

classic CoS (传统 CoS) 用模板条目的 DN 和目标条目的属性值之一来标识模板条目。

classification rules (分类规则) 用于将资源指定给某一类别或几个类别的一组规则。

classloader (类加载器) 基于 Java™ 技术的组件，负责按照特定规则加载 Java 类。另参见 [classpath \(类路径\)](#)。

class of service (服务类) 参见 [CoS](#)。

classpath (类路径) 用于标识存储 Java 类的目录和 Java™ 归档 (Java™ Archive, JAR) 文件的路径。另参见 [classloader \(类加载器\)](#)。

CLD (Calendar Lookup Database, 日历查找数据库) 一个插件, 用于在日历数据库分布于两个或多个后端服务器时, 确定日历的物理位置。Calendar Server 提供 LDAP CLD 插件和算法 CLD 插件。

cleartext (明文) 未加密文本。

CLI (command-line interface, 命令行界面) 可在用户提示符下键入可执行指令的界面。另参见 [administration interface \(管理界面\)](#)。

client (客户机) 请求软件 [service \(服务\)](#) 的软件。(注: 此术语并非指人—参见 [end user \(最终用户\)](#)。) 客户机可以是请求另一服务的某项服务, 或者是最终用户所访问的某个 GUI 组件。

client authentication (客户机验证) 客户机证书验证过程, 方法为用密码验证证书签名及通往信任 CA 列表内 CA 的证书链。另参见 [authentication \(验证\)](#)、[\(certificate authority\) \(证书授权机构\)](#)。

client contract (客户机订约) 一种订约, 它确定客户机和 EJB™ 容器之间的通信规则, 为使用企业 bean 的应用程序建立统一开发模型, 并通过标准化与客户机之间的关系来确保 bean 的更大再使用率。

client conditional properties (客户机条件属性) Portal Server Mobile Access 客户机类型的属性, 它允许管理员为频道指定属性或为给定客户机指定容器频道。

client database (客户机数据库) 对于 Portal Server Mobile Access 来说, 它是由内部和外部库组成的数据库。内部库包含所有默认移动设备数据定义。外部库包含自定义的客户机数据定义, 这些定义会覆盖内部库中的定义。

client detection (客户机检测) 用于确定访问门户的每个移动设备的功能和特性的 Access Manager 进程。

Client Editor (客户机编辑器) 允许用户创建客户机类型并管理客户机属性的 Access Manager 界面。Client Editor 界面可从 Access Manager 控制台访问。

client identifier (客户机标识符) 将连接及其对象与某状态关联在一起的标识符，它由 Java Enterprise System 消息服务器代表客户机来维护。

Client Manager (客户机管理器) 可从控制台访问的 Access Manager 界面，利用它可管理客户机类型和属性。

client profile (客户机配置文件) 标识每个客户机的 Access Manager 配置文件。

***client runtime (客户机运行时环境)** 参见 Java Enterprise System 客户机运行时环境。

client-server model (客户机/服务器模型) 一种计算模型，其中联网计算机向其他客户机计算机提供特定的服务。例如，DNS 的名称服务器和名称解析器范例，以及文件服务器/文件客户机之间的关系（如 NFS 和无硬盘主机）。

***client type (客户机类型)** Access Manager 客户机数据库内的条目。

clientType 一种属性，指为 Access Manager 客户机数据提供唯一性索引的名称。

cluster (群集) (1) 由高速网络连接在一起的一组计算机，它们像一台多 CPU 计算机一样协同工作。如果群集内一台服务器出现故障，则其工作可转移至另一正常运行的服务器。另参见 [failover \(故障转移\)](#)。

(2) 一组添加到并由主管理服务器控制的远程从属管理服务器。群集内所有服务器使用的平台都必须相同且拥有相同的用户 ID 和密码。

(3) 两个或多个互连节点或域，它们共享一个群集文件系统且被配置为共同执行故障转移、并行或可伸缩资源。

(4) 两个或多个互连代理，它们一起串联工作以提供讯息传递服务。

cluster configuration repository (群集配置系统信息库) 参见 [category \(类别\)](#)。

cluster file system (群集文件系统) 在全群集范围提供对现有本地文件系统的高可访问性的群集服务。

cluster interconnect (群集互连) 硬件联网的基础结构，包括电缆、群集传输结点及群集传输适配器。Messaging Server 和数据服务软件使用此基础结构进行群集内部的通信。

cluster member (群集成员) 当前具体群集的活动成员。此成员能够与其他群集成员共享资源，并且能够为其他群集成员和群集的客户机提供服务。另参见 [cluster node \(群集节点\)](#)。

cluster membership monitor (群集成员监视器) 参见 [CMM](#)。

cluster node (群集节点) 被配置为群集成员的节点。群集节点可能是或不是当前成员。另参见 [cluster member \(群集成员\)](#)。

cluster transport adapter (群集传输适配器) 节点上的网络适配器，它将该节点连接到群集互连之上。另参见 [cluster interconnect \(群集互连\)](#)。

cluster transport cables (群集传输电缆) 连接到各端点的网络连接。它连接群集传输适配器与群集传输结点或两个群集传输适配器。另参见 [cluster interconnect \(群集互连\)](#)。

cluster transport junction (群集传输结点) 用作群集互连一部分的硬件交换机。另参见 [cluster interconnect \(群集互连\)](#)。

CMM (cluster membership monitor, 群集成员监视器) 用来维护连贯一致的群集成员名单的软件。其他群集软件将使用此成员信息来确定于何处查找高可用性的服务。CMM 确保非群集成员不能损坏数据也不能将已损坏或不一致的数据传输到客户机。

CMP 参见 [container-managed persistence \(容器管理持久性\)](#)。

CMR 参见 [container-managed relationship \(容器管理关系\)](#)。

CMT 参见 [container-managed transaction \(容器管理事务\)](#)。

cn 参见 [common name attribute \(通用名称属性\)](#)。

CNAME record (CNAME 记录) 一种将域名别名映射到域名的 DNS 记录。

collation order (整理顺序) 一种与语言及文化相关的信息，说明如何排序给定语言的字符。此信息可能包括字母表中字母的顺序或如何比较有重音字母与无重音字母。

collection (集合) 包含文档信息的数据库，如词汇表或文件属性。搜索功能将使用集合来检索符合指定搜索条件的文档。

collocation (并置) 一种属性，说明位于同一节点上。此概念在群集配置期间使用，以改善性能。

colocate (协同定位) 将某组件和与其相关组件置于同一内存空间处，以避免远程过程调用从而改善性能。

column (列) 数据库表内的字段。

command line interface (命令行界面) 参见 [CLI](#)。

comm_dssetup.pl 一种 Directory Server 准备工具，它可以使现有的 Directory Server 做好供 Messaging Server 使用的准备。

comment character (注释字符) 位于行首、将该行转变为非可执行注释的字符。

commit (提交) 通过向数据库发送所需命令来完成某项事务。另参见 [rollback \(回滚\)](#)、[transaction \(事务\)](#)。

common domain (通用域) 在有多个身份认证提供者的 [circle of trust \(信任环\)](#) 中，服务提供者需要一个能够确定主体将使用哪一身份认证提供者的方法。由于此功能必须跨任一数量的域名系统 (domain name system, DNS) 域工作，所以 Liberty 采用了为环内所有身份和服务提供者创建一个通用域的方法。此预定域称为通用域。在通用域内，当主体通过服务提供者的验证后，身份认证提供者将编写一个存储主体的身份认证提供者的通用域 cookie。然后，当主体尝试访问环内其他服务提供者时，该服务提供者将读取此通用域 cookie，请求便可转发给正确的身份认证提供者。

common log file format (通用日志文件格式) 服务器将信息输入访问日志时使用的格式。所有主要服务器（包括 Web Server）之间的格式均相同。

common name attribute (通用名称属性) 标识由 LDAP 目录内条目定义的人员或对象的 cn 属性。

Communication Services (通信服务) 一种综合讯息传递解决方案，可将集成电子邮件、日历、即时消息以及在线状况信息传送给企业客户。“通信服务”核心解决方案包括 Messaging Server、Calendar Server 和 Instant Messaging Server。

Compass 一种为 Portal Server 3.0 提供搜索功能的搜索引擎服务。该搜索引擎已集成到 Portal Server 6.0 的内核中。参见 [Search Engine \(搜索引擎\)](#)。

component (组件) (1) 一种软件逻辑单元，它是分布式应用程序的建立基础。组件可以是包含在 Java Enterprise System 内的 [system component \(系统组件\)](#) 之一，或者是自定义开发的 [application component \(应用程序组件\)](#)。应用程序组件通常符合分布式组件模型（如 CORBA 和 J2EE™ 平台）且执行某些特定计算功能。这些组件单独或组合起来提供 [business service \(业务服务\)](#)，并可以封装为 [web service \(web 服务\)](#)。

(2) Web 应用程序、企业 bean、消息驱动 bean、应用程序客户机或连接器。另参见 [module \(模块\)](#)。

component contract (组件订约) 建立企业 bean 及其容器间关系的订约。

component product descriptor file (组件产品描述符文件) 含有给定组件产品的元数据（通常为 XML 格式）的文件。

component state (组件状态) 描述日历事件（如会议）的属性集。在 WCAP 中，compstate 参数允许 fetch 命令按组件状态返回事件。例如，compstate 可能为 REPLY-DECLINED（参加者拒绝出席会议）或 REQUEST_NEEDS-ACTION（参加者在会议中尚未有任何行动）。

compromised key list (已泄密密钥列表) 参见 [CKL](#)。

computed attribute (已计算属性) 一种属性，它不与条目本身一起存储而与操作结果中的常规属性一起返回到客户机应用程序。

conference room (会议室) 一种预先建立的聊天室，由管理员或其他拥有 `sysRoomsAdd` 权限的用户配置。管理员或其他拥有 `sysRoomsAdd` 权限的用户可以确定哪些用户可以查看和进入会议室。

configuration (配置) 调优服务器或为组件提供元数据的过程。通常，特定组件的配置保存在该组件的部署描述符文件中。另参见 [administrative console \(管理控制台\)](#)、[deployment descriptor \(部署描述符\)](#)。

configuration administrator (配置管理员) 拥有管理权限的人员，可管理服务器以及整个服务器软件拓扑中的配置目录数据。对于整个服务器软件拓扑中的所有资源，配置管理员拥有不受限的访问权。只有此管理员可向其他管理员分配服务器访问权限。配置管理员在管理员组及其成员就位之前，负责管理性配置的初始管理。

Configuration Directory Server (配置目录服务器) 为服务器或一组服务器维护配置信息的 Java Enterprise System Directory Server。

configuration file (配置文件) 包含信息传递系统内特定组件的配置参数的文件。

conflict (冲突) 在通过复制使两个服务器之间的数据保持同步之前，更改不同目录服务器上的相同目录数据时出现的情况。服务器同步时，将检测他们的副本是否一致，然后解决冲突或记录错误。

conflict resolution (冲突解决方案) 用于解决更改信息的决定性程序。详细信息参见“Java Enterprise System Directory Server 管理指南”。

congestion thresholds (拥塞阈值) 由系统管理员设置的磁盘空间限制，以便当系统资源不足时通过限制新操作来防止数据库超载。

connection (连接) (1) 与 Java Enterprise System 消息服务器的活动连接。此连接可为队列连接或主题连接。

connection factory (连接工厂) 一种对象，产生使 J2EE™ 组件能够访问资源的连接对象。它用于创建 Java™ 消息服务 (Java™ Message Service, JMS) 连接 (TopicConnection 或 QueueConnection)，应用程序代码通过这些连接可以使用所提供的 JMS 实现。应用程序代码使用 Java Naming and Directory Interface™ (JNDI) 服务查找采用 JNDI 名称的连接工厂对象。

connection pool (连接池) 一组连接。通过缓存和重新使用物理连接来避免连接开销，以及减少大量线程之间共享的连接数量来实现数据库的高效访问。另参见 [JDBC™ connection pool \(JDBC™ 连接池\)](#)。

connector (连接器) 容器的一种标准扩展机制，用于提供到 EIS 的连通性。连接器具有 EIS 特异性，由资源适配器和获得 EIS 连通性所需的应用程序开发工具组成。资源适配器通过支持连接器体系结构内定义的系统级订约而插入容器。

connector architecture (连接器体系结构) 用于集成 J2EE™ 应用程序与 EIS 的体系结构。此体系结构由两部分构成：EIS 供应商提供的资源适配器和允许此资源适配器插入的 J2EE 服务器。此体系结构定义了资源适配器为了能够插入 J2EE 服务器而必须支持的一组订约，例如事务、安全性和资源管理。

console (控制台) 一种 GUI，使用户能配置、监视、维护和故障排除许多服务器软件组件。

consume (使用) 接收消息使用者从目标设备带来的消息。

consumer (使用者) (1) 包含从供应者服务器复制而得的目录树或子树的服务器。
(2) 由用于从目标设备接收消息的会话创建的对象 (MessageConsumer)。在点对点传送模型下，使用者就是一个接收器或浏览器 (QueueReceiver 或 QueueBrowser)。在发布/订阅传送模型下，使用者就是一个订阅者 (TopicSubscriber)。

consumer replica (使用者副本) 引用对主副本进行的所有添加、修改和删除操作的副本。服务器可持有任何数量的不同命名上下文的使用者副本。

contact (联系人) 接收或发送即时消息的用户或 LDAP 组的 userID (名称)。可将联系人添加到个性化联系人组中，以便监视其在线状态。在其他即时讯息传递环境中也称作“好友”。

contact group (联系人组) 用户维护的联系人列表。实际列表存储在 Instant Messaging Server 上。您可以创建联系人组以便合理地管理联系人。

contact list (联系人列表) 在 Java Enterprise System Instant Messaging 中，它是所有联系人组的列表。

container (容器) (1) 为特定类型的 J2EE™ 组件提供生命周期管理、安全性、部署和运行时服务的实体。Java Enterprise System Application Server 提供 web 和 EJB™ 容器且支持应用程序客户机容器。另参见 [component \(组件\)](#)。

(2) 在 Java Enterprise System Portal Server 6.0 中，容器是指主要通过汇集其子频道内容的方法来生成自身内容的频道。在 Java Enterprise System Directory Server Access Management Edition 中，容器定义着获取其他 Directory Server Access Management Edition 对象的组织对象的类型。

container entry (容器条目) 代表目录中子树顶层的条目。

container-managed persistence (容器管理持久性) EJB™ 容器负责实体 bean 的持久性。实体的 bean 变量和数据存储之间的数据传输，其中数据访问逻辑由 Java Enterprise System Application Server 提供。另参见 [bean-managed persistence \(bean 管理持久性\)](#)。

container-managed relationship (容器管理关系) 一对类内字段间的关系，对此关系的一方进行的操作将会影响到另一方。

container-managed transaction (容器管理事务) 企业 bean 的事务划分由 EJB™ 容器明确指定并自动控制。另参见 [bean-managed persistence \(bean 管理持久性\)](#)。

control descriptor (控制描述符) 一组企业 bean 配置条目，使您能够指定可选单个属性覆盖 bean 方法，以及企业 bean 事务和安全属性。

conversational state (会话状态) 对象状态由于与同一客户机的重复交互活动而更改。另参见 [persistent state \(持久状态\)](#)。

cookie 信息的小集合，可被发送到来访浏览器并在该浏览器后续的每一次访问时被检索，以便服务器可识别来自同一客户机的访问。**Cookies** 具有域特异性，并可利用与在应用程序和服务器之间进行其他数据交换相同的 web 服务器安全功能。接受 cookie 可使 web 页面加载更加快速，同时不会对计算机带来安全威胁。

cooperating server (协作服务器) 想与您的服务器进行通信以及您的服务器想与之通信的服务器。又称为“协同服务器”。每个协作服务器都有一个给定的符号名，它是由字母和数字组成的字符串，例如 `coservern`，其中 n 为数字。

CORBA (common object request broker architecture, 公共对象请求代理体系结构) 面向对象的分布式计算的标准体系结构定义。

core service (核心服务) 定义 Java™ Enterprise System 服务器提供的基础功能的一个或多个密钥服务，与支持服务或附属服务相对。

CoS (class of service, 服务类) 在条目间共享属性的方法。

CoS definition entry (CoS 定义条目) 标识所用 CoS 类型的条目。此条目作为 LDAP 子条目存储在其所影响的分支下。

coserver (协同服务器) 参见 [cooperating server \(协作服务器\)](#)。

CoSNaming provider (CoSNaming 提供者) 为了支持全局 JNDI 名称空间 (IIOP 应用程序客户机可访问)，Java Enterprise System Application Server 包括基于 J2EE 的 CosNaming 提供者，它支持对 CORBA 引用的绑定 (远程 EJB™ 引用)。

CoSNaming Service (CoSNaming 服务) 基于 IIOP 的命名服务。

CoS template entry (CoS 模板条目) 包含共享属性值的列表的条目。

CRAM-MD5 RFC 2195 规范中记载的一种轻量标准跟踪验证机制。当仅需要防止用户登录密码遭到网络窃听时，此机制可提供替代 TLS (SSL) 的快速 (虽然不太稳定) 备选措施。

crawler 参见 [robot](#)。

create method (创建方法) 在创建企业 bean 时对其进行自定义的方法。

CRL (certificate revocation list, 证书撤销列表) 由证书授权机构发布的列表, 此列表用于指明客户机用户或服务器用户不应再信任的所有证书。在此情况下, 该证书已被撤销。另参见 [CKL](#)。

cronjob (仅限 UNIX) 由时钟守护进程在配置过程中自动执行的任务。另参见 [crontab file \(crontab 文件\)](#)。

crontab file (crontab 文件) (仅限 UNIX) 在指定时间自动执行的命令的列表 (每行一个命令)。

CSAPI (Calendar Server application programming interface, Calendar Server 应用编程接口) 一种编程接口, 提供修改或增强 Calendar Server 的功能集的能力。CSAPI 模块是在启动 Calendar Server 时从 cal/bin/plugins 目录加载的插件。

CUA (Calendar user agent, 日历用户代理) 日历客户机用于访问 Calendar Server 的应用程序。

daemon (守护进程) (仅限 UNIX) 在后台运行的、独立于终端的一种程序, 在必要时随时执行某项功能。守护进程程序的常见示例有邮件处理器、许可证服务器和打印守护进程。

DAP (directory access protocol, 目录访问协议) ISO/ITU-T X.500 协议, LDAP 的基础。

data access logic (数据访问逻辑) 涉及与某数据源的交互的业务逻辑。

database (数据库) 指代关系数据管理系统 (relational database management system, RDBMS) 的通用术语。它是一种软件数据包, 用来创建和处理大量相关的、经过组织的数据。

database connection (数据库连接) 与数据库或其他数据源之间的通信链接。各组件可同时创建和处理若干数据库连接, 以访问数据。

database wire protocol (数据库有线通信协议) 参见 [DWP](#)。

data redundancy unit (数据冗余单元) (DRU) 一组 HADB 节点，其中含有半数的活动和备用节点以及一个完整的数据副本。HADB 被组织为两个 DRU，这两个 DRU 互相镜像。为确保容错，支持一个 DRU 的计算机在电源、处理单元和存储单元方面必须完全自支持。另参见 [HADB node \(HADB 节点\)](#)、[active node \(活动节点\)](#)、[spare node \(备用节点\)](#) 和 [mirror node \(镜像节点\)](#)。

data service (数据服务) 被装备为在 RGM 控制下作为高可用资源运行的应用程序。

data source (数据源) 数据源（例如数据库）的句柄。数据源在 Application Server 注册，然后以编程方式检索，以与数据源建立连接和接口。数据源定义指定连接数据源的方式。

data source object (数据源对象) 数据源对象具备一组属性，用于标识和描述此对象代表的实际数据源。

data store (数据存储) (1) 包含目录信息的存储区，通常用于整个 *DIT*。

(2) 永久存储 Message Queue 代理所需信息（持久订阅、关于目标的数据、持久性消息、审计数据）的数据库。

DC tree (DC 树) (domain component tree, 域组件树) 镜像 DNS 网络语法的一种 *DIT*。“DC 树”中的标识名示例为
cn=billbob,dc=bridge,dc=net,o=internet。

declarative security (声明性安全) 组件配置文件中的声明性安全属性，允许组件的容器（例如，bean 的容器或 servlet 引擎）隐式管理安全性。这种安全性不需要程序控制。它与 [programmatic security \(程序安全性\)](#) 相对。另参见 [container-managed persistence \(容器管理持久性\)](#)。

declarative transaction (声明性事务) 参见 [container-managed transaction \(容器管理事务\)](#)。

decryption (解密) 破解加密信息的过程。另参见 [encryption \(加密\)](#)。

default calendar (默认日历) 用户登录到 Calendar Express 之后看到的第一个日历。默认日历的日历 ID 通常与用户的用户 ID 相同。例如，jdoe@example.com 的默认日历名称为 jdoe。

default index (默认索引) 安装 Directory Server 时为每个数据库实例创建的一组索引。安装 Java Enterprise System Directory Server 之后，即为每个数据库实例创建了一组默认索引。详细信息参见“Java Enterprise System Directory Server 管理指南”。

default master (默认主站) 默认群集成员，对于这些成员，故障转移资源类型进入联机状态。

defederation (取消联合) 参见 [federation termination \(联合终止\)](#)。

definition entry (定义条目) 参见 [CoS definition entry \(CoS 定义条目\)](#)。

defragmentation (取消分段) MIME 功能，可使分为若干短消息或片段的长消息重新组合。每一片段中显示的“消息部分内容类型”头字段所含的信息有助于将各片段重新组合为一条消息。另参见 [fragmentation \(分段\)](#)。

delegated administrator console (委托管理员控制台) 一种基于 Web 浏览器的软件控制台，使域管理员可以对托管域添加和修改用户及用户组。还允许最终用户更改其密码、设置消息转发规则、设置休假规则和列出邮件列表订阅。

delegated administrator for messaging and collaboration (讯息传递和协作委托管理员) 使域管理员可以对托管域添加和修改用户及用户组的一组界面 (GUI 和实用程序)。

delegated administrator server (委托管理员服务器) 采用托管域处理对目录的访问控制的守护程序。

delegation (委托) 一种面向对象的技术，将若干对象的组合作为一种实现策略使用。负责操作结果的一个对象将实现过程委托给另一对象。例如，类加载器经常将某些类的加载委托给其父项。

delete a message (删除消息) 标记某消息以进行删除。删除的消息不会从消息存储中实际移除，除非用户通过单独操作将其擦除或清除。另参见 [purge a message \(清除消息\)](#)、[expunge a message \(抹去消息\)](#)。

delivery (传送) 参见 [message delivery \(消息传送\)](#)。

delivery mode (传送模式) 表示讯息传递可靠性的一种模式：消息可被保证传送并成功使用一次且仅使用一次（持久性传送模式），或者被保证至少传送一次（非持久性传送模式）。

delivery mode (传送模型) 传送消息时利用的模型。模型可为点对点式或发布/订阅式。在 Java™ 消息服务 (Java™ Message Service, JMS) 中，每一模型都有单独的编程域，这些域使用特定的客户机运行时对象和目标类型（队列或主题）以及一个统一的编程域。

delivery policy (传送策略) 一种规范，详细说明注册多个消息使用者后队列如何路由消息。此类策略有单一、故障转移和循环。

delivery status notification (传送状态通知) 此类消息提供向某收件人发送的消息的状态信息，例如指明由于网络故障而延迟传送的消息。

denial of service attack (拒绝服务攻击) 个人通过向邮件服务器发送大量消息有意或无意对其进行攻击的一种情形。服务器的总处理能力可能会受到严重影响，服务器本身也可能超载和失效。

deny filter (拒绝过滤器) Java Enterprise System Messaging Server 的一种访问控制规则，用于确定被拒绝访问以下一项或多项服务的客户机：POP、IMAP 或 HTTP。另参见 [allowed attributes \(允许的属性\)](#)。

deployment (部署) Java Enterprise System 解决方案生命周期过程中的一个阶段，在此阶段，部署方案被转化成部署设计，后者被实现、原型化而后在生产环境中展开。此过程的最终产品也被称为部署（或已部署解决方案）。

deployment architecture (部署体系结构) 一种高层次设计，描述了 [logical architecture \(逻辑体系结构\)](#) 到实际计算环境的映射。物理环境包括内联网或 Internet 环境中的计算机、这些计算机之间的网络链接以及为软件提供支持所需的任何其他物理设备。

deployment descriptor (部署描述符) 随各模块和应用程序提供的一种 XML 文件，用于描述应如何部署各应用程序。部署描述符指示部署工具通过特定的容器选项部署某模块或应用程序，并说明了部署程序必须解决的特定配置要求。

deployment scenario (部署方案) 用于 Java Enterprise System 解决方案的一种 **logical architecture (逻辑体系结构)** 以及解决方案要满足业务需求而必须达到的服务质量要求。服务质量要求包括下列各方面的要求：性能、可用性、安全性、可维护性以及可伸缩性/潜在容量。部署方案是部署设计的起点。

depth (深度) 从“搜索引擎”中某站点的起点开始追踪的链接的数量。在您定义站点时，即定义 robot 从该点开始可追踪的链接的数量，从而限定搜索的深度。

dereference an alias (解除别名引用) 在绑定或搜索操作中，指定目录服务将某个条目的别名标识名转换为实际标识名。

Desktop (桌面) 参见 *Java™ System Portal Server Desktop*。

destination (目标) Java Enterprise System 消息服务器中的物理位置，生成的消息被传送到此位置进行路由选择，随后传送给使用者。该物理目标由受管理的对象标识和封装。客户机使用此受管对象指定目标，客户机为此目标生成消息并从此目标使用消息。

destination resource (目标资源) 代表“主题”目标或“队列”目标的对象。由各应用程序用来对“队列”进行读和写操作或对“主题”进行发布和订阅。应用程序代码使用 Java Naming and Directory Interface™ (JNDI) 服务查找使用“JNDI 名称”的 Java™ 消息服务 (Java™ Message Service, JMS) 资源对象。

development (开发) Java Enterprise System 解决方案部署过程中的一项任务，通过此任务对某 **deployment architecture (部署体系结构)** 的自定义组件进行编程和测试。

device detection (设备检测) 参见 **client detection (客户机检测)**。

device group (设备组) 可从群集 HA 配置中的不同节点进行控制的一组用户定义设备资源，例如磁盘。该设备组可包括磁盘、Solstice DiskSuite™ 软件磁盘集和“VERITAS 卷管理器”磁盘组设备资源。

device ID (设备 ID) (1) 用于识别可通过“Solaris™ 操作系统”访问的设备的一种机制。在 `devid_get(3DEVID)` 手册页中有“设备 ID”的说明。

(2) Messaging Server DID 驱动程序使用设备 ID 确定不同群集节点上各 Solaris 逻辑名称之间的相关性。DID 驱动程序可探测每个设备以获取其设备 ID。如果该设备 ID 与群集中其他位置的另一设备 ID 一致，则这两个设备被命名为相同的 DID 名称。如果以前群集中没有此设备 ID，则会指定一个新的 DID 名称。另参见 **Solaris™ logical name (Solaris™ 逻辑名称)** 和 **DID**。

device information（设备信息） 用于 Portal Server Mobile Access 的设备特定客户机数据。

DHCP（dynamic host configuration protocol, 动态主机配置协议）Internet 提出的一种标准协议，该协议允许系统将 IP 地址动态分配给网络上的各个计算机。另参见 [IP address](#)（IP 地址）。

DID 参见 [device ID](#)（设备 ID）。

DID driver（DID 驱动程序） 由 Java Enterprise System Messaging Server 软件实现的一种设备 ID 驱动程序，用以提供跨群集的一致设备名称空间。另参见 [DID name](#)（DID 名称）。

DID name（DID 名称） 标识 SunPlex™ 系统中的全局设备的设备 ID 名称。DID 名称是一种群集式标识符，并与“Solaris™操作系统”逻辑名称具有一对一或一对多的关系（此类逻辑名称采用 dXsY 格式，其中 X 为整数，Y 为分片名称）。另参见 [Solaris™ logical name](#)（Solaris™ 逻辑名称）。

digest authentication（摘要验证） 一种验证类型，允许用户无需以明文方式发送用户名和密码即可进行验证。浏览器使用 MD5 算法创建摘要值。服务器使用“摘要验证”插件比较客户机提供的摘要值。

DIGEST-MD5 一种比 CRAM-MD5 更安全的轻量标准跟踪验证机制。记载于 RFC 2831 规范中，同时提供了无需额外的 TLS (SSL) 设置开销即可保护整个连接的选择。

digital signature（数字签名） 用于验证消息和签名者的一种电子安全机制。

directive（指令） 一种“搜索引擎”语句，它使用特殊格式调用函数（例如 robot 应用函数）并将参数传递给参数块中的函数。例如，以下指令调用 `enumerate-urls` 函数，并为 `max` 和 `type` 传递参数：

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

directory (目录) 一种经过优化用于读数据而非写数据的特殊数据库。大多数目录都基于行业标准协议 LDAP (轻量目录访问协议)。

directory access protocol (目录访问协议) 参见 *CUA*。

directory context (目录上下文) 目录信息树中的点, 搜索用于验证消息存储访问的用户名和密码的条目时, 该点为一个起点。另参见 *base DN (基 DN)*。

directory entry (目录条目) 由标识名标识的一组目录属性及它们的值。每个条目都包含一个对象类属性, 说明条目描述的对象种类并定义条目包含的一组属性。

directory information tree (目录信息树) 参见 *DIT*。

directory lookup (目录查找) 根据指定用户或资源的名称或其他特征搜索目录, 以查找有关该用户或资源的信息的过程。

Directory Manager (目录管理员) (1) 对目录服务器数据库具备管理权限的用户。该用户不具备访问控制权限 (将目录管理员视为目录的超级用户)。

(2) 具备特权的数据库管理员, 类似于 UNIX 系统中的超级用户。目录管理员不具备访问控制权限。

directory schema (目录模式) 定义可存储在目录中的数据的一组规则。

directory server (目录服务器) (1) 通过逻辑集中式信息库提供某组织内人员和资源的相关信息的服务器。另参见 *LDAP*、*Java™ System Directory Server* 和 *Java™ System Directory Server Access Management Edition*。

(2) 基于 LDAP 的 Java Enterprise System 目录服务。

Directory Server Console (Directory Server 控制台) LDAP 客户机应用程序, 可提供用于浏览、配置和管理目录内容的图形用户界面。“Directory Server 控制台”是 Java Enterprise System Directory Server 产品的一个组件。

directory service (目录服务) 一种数据库应用程序, 用于管理组织内人员和资源的基于属性的描述性信息。

directory synchronization (目录同步) 将 MTA 目录高速缓存与存储在目录服务中的当前目录信息同步的过程。另参见 [MTA directory cache \(MTA 目录高速缓存\)](#)。

disconnected state (断开状态) 在此状态下，邮件客户机连接至服务器，创建选定消息的高速缓存副本，然后断开与服务器的连接。

disk device group (磁盘设备组) 参见 [device group \(设备组\)](#)。

disk group (磁盘组) 参见 [device group \(设备组\)](#)。

diskset (磁盘集) 参见 [device group \(设备组\)](#)。

Dispatcher (分发程序) 为已定义的 TCP 端口处理连接请求的 MTA 组件。“分发程序”是一种多线程连接分发代理程序，允许多个多线程服务器共同负责分担某项指定服务。使用“分发程序”时，您可以使若干多线程 SMTP 服务器进程同时运行。

display profile (显示配置文件) 一组 XML 文档，用于在 Java Enterprise System Portal Server 中定义和配置提供者和频道。

distinguished name (标识名) 参见 [DN](#)。

distributable session (可分布会话) 可在某个群集的所有服务器中分布的用户会话。

distributed enterprise application (分布式企业应用程序) 一种应用程序，其逻辑遍布于整个网络或 Internet 环境（分布式方面），并且其范围和规模满足了生产环境或服务提供者的需要（企业方面）。

distributed indexing (分布式索引建立) 在“搜索引擎”中分配不同 robot，从而为网络的不同部分建立索引的过程。采用分布式索引建立可减少每个 robot 的负载。这样，一个单独的“搜索引擎”便可从所有不同的 robot 收集所有资源说明，方法是导入每个 robot 的资源说明。

Distributed Lock Manager (分布式锁管理器) 在共享磁盘 Oracle Parallel Server 环境中使用的加锁软件。“分布式锁管理器”使 Oracle 进程可在不同节点上运行，以同步数据库访问。“分布式锁管理器”专为高可用性而设计。如果某进程或节点崩溃，不需要关闭或重新启动余下的节点。此时会对“分布式锁管理器”进行快速重新配置，以便从此故障中恢复。

distributed transaction (分布式事务) 可应用于多个可能驻留在各个单独服务器上的异构数据库的单独事务。

distribution (分发) 以各种媒体形式和包装方法展现自己的一个二进制位集合。

distribution list (分发列表) 参见 [mailing list \(邮递列表\)](#)。

distribution list owner (分发列表所有者) 参见 [mail list owner \(邮寄列表所有者\)](#)。

DIT (directory information tree, 目录信息树) 存储在目录中的信息的逻辑表示法。DIT 镜像多数文件系统使用的树模型，树根点显示在分层结构的顶部。

DN (distinguished name, 标识名) 目录中条目名称和位置的字符串表示法。

DN attribute (DN 属性) 包含关联用户、用户组或对象的标识信息的文本字符串。

DNS (domain name system, 域名系统) 网络中的计算机用来将 IP 地址（例如 00.120.000.168）与主机名（例如 [www.example.com](#)）相关联的系统。客户机通常使用 DNS 查找它们要联系的服务器的 IP 地址。DNS 中的数据通常在本地表中扩充，例如从 NIS 或 UNIX 系统中的 `/etc/hosts` 文件。另参见 [IP address \(IP 地址\)](#)。

DNS alias (DNS 别名) DNS 服务器了解的指向另一主机的主机名。DNS 别名作为一个 DNS CNAME 记录执行。计算机始终具备一个实名，但它们可有一个或多个别名。例如，别名 [www.example.com](#) 可能指向目前充当服务器的名为 [realthing.example.com](#) 的实际计算机。

DNS database (DNS 数据库) 域名（主机名）及其相应的 IP 地址的数据库。

DNS domain (DNS 域) 一组计算机，它们的主机名共用一个公用后缀（域名）。从语句构成上说，一个 Internet 域名由一系列句点（圆点）分隔的名称（标签）构成，例如 corp.mktng.example.com。另参见 [document type definition \(文档类型定义\)](#)。

DNS spoofing (DNS 电子欺骗) 一种网络攻击形式，在这种攻击下，DNS 服务器遭到破坏，因而提供虚假的信息。

document (文档) 网络中的一种文件，多数是网页或字处理文档，但也可能是文本文件、电子表格等等。按“搜索引擎”建立索引的资源的通用术语。

document root (文档根目录) (1) 服务器上的目录，此目录包含将为访问 Java Enterprise System Web Server 的用户显示的文件、图像和数据。

(2) 服务器上的目录，此目录包含您要显示给访问该服务器的用户的文件、图像和数据。

(3) 文档根目录（有时称为文档主目录）是包含您要提供给远程客户机的所有虚拟服务器文件的中心目录。

document type definition (文档类型定义) 参见 [DTD](#)。

domain (域) (1) 全限定域名的最后一部分，用于标识拥有此域名的公司或组织（例如，example.com、host.example.com）。

(2) 受控于某单独计算机系统的资源。另参见 [administration domain \(管理域\)](#)、[DNS domain \(DNS 域\)](#)、[hosted domain \(托管域\)](#)、[virtual domain \(虚拟域\)](#)。

(3) Java™ 消息服务 (Java™ Message, JMS) 客户机用来为 JMS 消息传递操作编程的一组对象。有两个编程域：一个用于点对点式传送模型，一个用于发布/订阅式传送模型。

Domain Administration Server (域管理服务器) “域管理服务器”是一个专门指定的“应用服务器”实例，用于为 Sun Java System Application Server 处理所有管理任务。它可维护和更新 Application Server 配置信息的中心信息库。如果 Domain Application Server 未在运行，则无法执行管理任务。

domain administrator (域管理员) 拥有以下管理权限的用户: 可以使用讯息传递和协作 GUI 或 CLI 的“委托管理员”创建、修改、删除托管域中的邮件用户、邮件列表、家庭帐户。默认情况下, 此用户可以担任拓扑中所有讯息传递服务器的消息存储管理员。

domain alias (域别名) 指向另一域的域条目。使用别名使托管域可拥有若干个域名。

domain hosting (域托管) 托管域的过程。在共享讯息传递服务器上托管一个或多个域的功能。例如, 域 `example.com` 和 `example.org` 均可托管在 `example.com` 邮件服务器上。用户向此托管域发送邮件, 并从那里接收邮件。邮件服务器的名称不出现在邮件地址中。

domain name (域名) (1) 在邮件地址中使用的主机名。

(2) 定义管理组织的唯一名称。域可包含其他域。域名的解释顺序为从右至左。例如, `example.com` 既是 Example 公司的域名, 又是顶级 `com` 域的子域。`example.com` 域可进一步分为子域, 例如 `corp.example.com` 等等。另参见 [host name \(主机名\)](#)、[fully qualified domain name \(全限定域名\)](#)。

domain name system (域名系统) 参见 [DNS](#)。

domain organization (域组织) 组织树中托管域下方的子域。对于要按部门编排用户和用户组条目的公司, 域组织很有用。

domain part (域部分) 邮件地址中“at”符号 (@) 右侧的部分。例如, `example.com` 是邮件地址 `jdoe@example.com` 的域部分。

domain quota (域配额) 分配给邮件消息的域的空间数量。空间数量由系统管理员配置。

domain registry (域注册表) 一种单独的数据结构, 其中包含安装服务器时创建和配置的所有域的域特定信息 (例如域名、域位置、域端口、域主机)。

domain rewrite rules (域重写规则) 参见 [rewrite rule \(重写规则\)](#)。

domain template (域模板) 重写规则的一部分，定义如何重写地址的主机和域部分。此模板可包括一个完全静态的主机和域地址或一个单独的字段替代字符串，或两者都包括。

double failure (双故障) HADB 中一个或多个镜像节点对同时发生故障。参见 [HADB](#)、[HADB node \(HADB 节点\)](#)、[active node \(活动节点\)](#)、[spare node \(备用节点\)](#)、[mirror node \(镜像节点\)](#) 和 [data redundancy unit \(数据冗余单元\)](#)。

drop word (丢弃词) 参见 [stop word \(止用词\)](#)。

DRU 参见 [data redundancy unit \(数据冗余单元\)](#)。

DSA (directory system agent, 目录系统代理) Directory Server 的 X.500 术语。

DSE (directory server entry, 目录服务器条目) 一种条目或特定于 DSA 的条目，含有与其相关的特定于服务器的附加信息。DSE (如根 DSE 或模式 DSE) 在每个服务器上具有不同的属性。

DSP (digital signal processing, 数字信号处理) 信号从模拟到数字的转换。使用电话以语音方式访问“门户服务器”软件需要 DSP cvar。

DSML (directory services markup language, 目录服务标记语言) 一族表示 XML 标记语言的文档格式，通过它们可在 XML 中表示目录服务。Java Enterprise System Directory Server 5.2 符合 DSML 标准的版本 2 (DSMLv2)。

DSN 参见 [delivery status notification \(传送状态通知\)](#)。

dservd 一种守护程序，它访问存放目录信息的数据库文件，并使用 LDAP 协议与目录客户机通信。

dssetup 一种 Java Enterprise System Directory Server 准备工具，它可以使现有的 Directory Server 做好供 Java Enterprise System Messaging Server 使用的准备。

DTD (document type definition, 文档类型定义) 关于一类 XML 文件的结构和属性的说明。

DWP (database wire protocol, 数据库有线通信协议) 一种 Calendar Server 专有协议，它允许多个服务器在同一 Calendar Server 系统内链接在一起，以形成分布式日历存储。Calendar Server 使用 DWP 检索存储在日历数据库中的远程数据。

dynamic group (动态组) 由 LDAP 搜索 URL 定义的邮件组。用户通常通过在其目录条目中设置 LDAP 属性加入组。

dynamic redeployment (动态重新部署) 参见 [dynamic reloading \(动态重新加载\)](#)。

dynamic reloading (动态重新加载) 在不重新启动服务器的情况下更新并重新加载组件的过程。默认情况下，servlet、使用 JavaServer Pages™ 技术 ([JSP™ technology \(JSP™ 技术\)](#)) 创建的页面以及企业 bean 组件可以动态地重新加载。又称为动态重新部署和版本化。

dynamic web application (动态 web 应用程序) 指 servlet、JSP™ 页面、内容提供者或任何其他需要由 Java™ web 容器处理的程序，该容器通过用户浏览器访问。对于 Java Enterprise System Portal Server，该应用程序安装在 web 服务器中。

EAR file (EAR 文件) (enterprise archive file, 企业归档文件) 含有 J2EE™ 应用程序的归档文件。EAR 文件带有 .ear 扩展名。另参见 [JAR file \(JAR 文件\)](#)。

e-commerce (electronic commerce, 电子商务) 表示通过 Internet 开展业务的术语。

EHLO command (EHLO 命令) SMTP 命令。该命令对服务器进行查询以确定该服务器是否支持 SMTP 扩展命令。在 RFC 1869 中定义。

EIS (enterprise information system, 企业信息系统) 解释为包式企业应用程序、事务系统或用户应用程序。EIS 的示例有 R/3、PeopleSoft、Tuxedo 和 CICS。

EJB container (EJB 容器) 参见 [container \(容器\)](#)。

ejbc utility (ejbc 实用程序) 企业 bean 的编译程序。此实用程序检查所有 EJB 类和接口是否符合 EJB™ 规范，并生成存根和框架。

EJB™ QL (EJB™ Query Language, EJB™ 查询语言) 一种查询语言，用于在容器管理关系定义的实体 bean 的网络中导航。

EJB™ technology (EJB™ 技术) (Enterprise JavaBeans™ technology, 企业 JavaBeans™ 技术) 企业 bean 是封装应用程序业务逻辑的服务器端组件。业务逻辑是实现应用程序目的的代码。例如，在一个清单控制应用程序中，企业 bean 可以用称为 `checkInventoryLevel` 和 `orderProduct` 的方法实现业务逻辑。通过调用这些方法，远程客户机可以访问由该应用程序提供的清单服务。另参见 [container \(容器\)](#)、[entity bean \(实体 bean\)](#)、[message-driven bean \(消息驱动 bean\)](#) 和 [session bean \(会话 bean\)](#)。

element (元素) 一个较大集合的成员，例如一个数组中的数据单元或一个逻辑元素。在 XML 文件中，元素是基本的结构单位。XML 元素含有子元素或数据，并且可能包含属性。

encapsulate (封装) 将知识固定在一个模块内。因为对象会封装数据和实现，因此对象的用户可以将它视为提供服务的黑箱。可以添加、删除或更改实例变量和方法，但如果对象提供的服务保持不变，则使用该对象的代码可以继续使用该对象，而无需重新编写。

encryption (加密) 通过使信息无法识读而保护信息免受未授权使用的过程。有些加密方法使用对信息进行加密的代码，称为密钥。另参见 [decryption \(解密\)](#)。

endpoint (端点) 群集传输适配器或群集传输结点上的物理端口。

end user (最终用户) 常常通过图形用户界面（如 Internet 浏览器或移动设备 GUI）使用分布式应用程序的人员。应用程序同时支持的最终用户数是决定应用程序的 [deployment architecture \(部署体系结构\)](#) 的重要因素。

ENS 参见 [event notification service \(事件通知服务\)](#)。

enterprise network (企业网络) 由相互连接的网络集合构成的网络，这些网络集合分布在一个地理上分散的区域中。企业网络满足了分布广泛的公司的需要，并用于公司的关键任务应用程序。

entity bean (实体 bean) 与物理数据相关的企业 bean，如数据库的一行。实体 bean 的生命较长，因为它们绑定到永久性数据。实体 bean 总是事务性的并可被多用户识别。参见 [message-driven bean \(消息驱动 bean\)](#)、[read-only bean \(只读 bean\)](#)、[session bean \(会话 bean\)](#)。

entropy (熵) 封闭系统中随机性的量度。特别是在 SSL 上下文中，为了在随机数生成中引入熵（确保随机性），使用了多个种子。

entry (条目) 一组属性和一个独有的标识名。

entry distribution (条目分发) 跨多个服务器分发目录条目的方法，以便扩展为可支持大量条目。

entry ID list (条目 ID 列表) 条目 ID 的列表。目录使用的每个索引由一个索引键值表和相应的条目 ID 列表组成。目录使用条目 ID 列表构建可能符合客户机应用程序搜索请求的备选条目的列表。

enumeration (枚举) robot 操作的一个阶段。在此阶段中，robot 寻找资源，包括提取和执行超文本链接。

envelope (信封) 含有关于电子邮件消息的发件人和收件人传输信息的容器。此信息并非消息头的组成部分。在消息从一个地点传输到另一个地点时，各种电子邮件程序都要使用信封。用户只能看到消息头和消息正文。

envelope field (信封字段) 消息信封中已命名的信息项，如 RCPT TO。

equality index (等同索引) 一种索引，利用它可以有效地搜索包含特定属性值的条目。

ERP (enterprise resource planning, 企业资源计划) 一种多模块软件系统。该系统通常包括一个关系数据库和多个应用程序，用以管理采购、库存、人员、客户服务、运输、财务计划和业务的其他重要方面。

error handler (错误处理程序) 处理错误的程序。在 Messaging Server 中，错误处理程序发出出错消息，并在邮寄主管填好错误处理程序操作表单后，对其进行处理。

error handler action form (错误处理程序操作表单) 发送到邮寄主管帐户的表单。该表单伴有 Messaging Server 不能处理的收到消息。邮寄主管通过填写表单指示服务器如何处理消息。

error message (错误消息) 报告错误或其他情况的消息。Java Enterprise System Messaging Server 在多种情况下会生成消息，特别是当其获得不能处理的电子邮件消息时。其他消息（称为通知错误）仅作为提供信息之用。

ESMTP 参见 [extended simple mail transfer protocol \(扩展的简单邮件传输协议\)](#)。

ESP 企业服务提供者。

ETRN command (ETRN 命令) SMTP 命令，该命令可以使客户机请求服务器启动其邮件队列的处理，以便获得在服务器等待客户机的消息。在 RFC 1985 中定义。

event (事件) (1) 在日历中具有关联日期和时间的条目。例如，事件可以是日历中的一个新会议或日程。

(2) 一个已命名的操作。该操作触发模块或外部 Java Naming and Directory Interface™ (JNDI) 资源的响应。

(3) 受管理对象在状态、控制权、严重性或说明方面发生的变化。

event notification service (事件通知服务) 一种通用服务。该服务接受可分类的服务器级别事件的报告，然后通知其他已登记对某些类别事件感兴趣的服务器。允许 Java Naming and Directory Interface™ (JNDI) Service 作为到远程 JNDI 服务器的桥梁。

expander (扩散程序) 电子邮件传送系统的组成部分。它允许消息传送到地址列表。邮件扩散程序用于实现邮件列表。用户将消息发送到单个地址（例如，users@example.com），然后邮件扩散程序负责传送到列表中的邮箱。又称为邮件分发器。另参见 [EXPN command \(EXPN 命令\)](#)。

expansion (扩散) 将发送地址为邮件列表的消息转换为足够满足邮件列表中每个成员的副本数的操作。适用于邮件列表的 MTA 处理。

expires header (到期标题) 远程服务器指定的返回文档的到期时间。

EXPN command (EXPN 命令) 用于扩展邮件列表的 SMTP 命令。在 RFC 821 中定义。

expunge a message (抹去消息) 永久清除已从“收件箱”中删除的消息。另参见 [delete a message \(删除消息\)](#)、[purge a message \(清除消息\)](#)。

extended simple mail transfer protocol (扩展的简单邮件传输协议) 一种 Internet 消息传输协议。ESMTP 将可选命令添加到 SMTP 命令集以获得增强功能，包括 ESMTP 服务器搜索远程站点所实现命令的功能。

extensible markup language (可扩展标记语言) 参见 [XML](#)。

extensible style language (可扩展样式语言) 参见 [XSL](#)。

extensible style language transformation (可扩展样式语言变换) 参见 [XSLT](#)。

extracting (提取) 查找文档中超文本链接的过程。每个被提取的链接将添加到 URL 池中进一步处理。

extranet (外联网) 公司内联网向 Internet 的延伸，以允许客户、供应商和远程工作人员访问数据。

facade (门面) 使用特定于应用程序的有状态会话 bean 管理各种 Enterprise JavaBeans™ 组件的位置。

facility (工具) 在 Messaging Server 日志文件条目中，用于表示生成日志条目的软件子系统（如“网络”或“帐户”）的名称。

factory class (工厂类) 创建持久性管理器的类。另参见 [congestion thresholds \(拥塞阈值\)](#)。

failback (故障恢复) 参见 [authorization \(授权\)](#)。

failfast (故障快除) 在故障节点的不当操作造成破坏以前，从其群集中有序地终止和移除。

failover (故障转移) (1) 一种恢复过程。通过该过程，可以将 bean 透明地从服务器崩溃中拯救出来。

(2) 计算机服务从一个系统向另一个系统的自动转移，以提供冗余备份。

(3) 发生故障后，资源组或设备组从当前主节点自动重新定位到新的主节点。

failover resource (故障转移资源) 其资源一次只能由一个节点正确控制的资源。另参见 [single-instance resource \(单一实例资源\)](#) 和 [scalable resource \(可伸缩资源\)](#)。

family group administrator (系列组管理员) 拥有在系列组中添加和删除系列成员的管理权限的用户。此用户可以授予访问组中其他成员的系列组管理权限。

fancy indexing (复杂索引) 提供比简单索引更多信息的索引方法。复杂索引按名称显示目录列表以及文件大小、上次修改日期和指示文件类型的图标。因此，客户机加载复杂索引所需的时间可能比简单索引更长。

fault monitor (故障监视器) 一个故障守护程序和用于探测数据服务各部分情况并采取措施的多个程序。另参见 [resource monitor \(资源监视器\)](#)。

federated identity (联合身份) 一个用户访问的所有服务提供者中的帐户信息（如个人数据、验证信息、消费习惯和历史记录、消费偏好等）的合并。该信息由用户管理，并且可以在用户的同意下，安全地与用户选择的提供者共享。

federation cookie (联合 cookie) 联合 cookie 是由名为 fedCookie 的 Access Manager 实现的 cookie。根据主体的联合状态，其值可为 yes 或 no。它不是 LAP 规范的已定义部分。

federation termination (联合终止) 一个过程，用户通过该过程取消在用户的身份认证提供者和联合服务提供者帐户间建立的关联关系。又称为取消联合。

file cache (文件高速缓存) 文件高速缓存含有关于文件和静态文件内容的信息。默认情况下，文件高速缓存处于启用状态。

file extension (文件扩展名) 文件名称的最后部分，通常定义文件的类型。例如，在文件名 index.html 中，文件扩展名为 html。

file transfer protocol (文件传输协议) 参见 [FTP](#)。

file type (文件类型) 给定文件的格式。例如，图形文件与文本文件具有不同的文件类型。文件类型通常由其文件扩展名标识。另参见 [fault monitor \(故障监视器\)](#)。

filter (过滤器) (1) 在搜索请求中，搜索范围内的条目要在搜索响应中返回所必须符合的模式。过滤器还用于角色构建和访问控制定义中。

(2) 定义特定资源类型的一组规则。站点定义使用这些过滤器定义 robot 应该接受或忽略的资源类型。

filtered role (过滤角色) 将角色分配给条目的方法。允许根据每个条目所含的属性，将条目分配给角色。这通过指定 LDAP 过滤器完成。符合过滤器要求的条目被视为拥有角色。

filtering (过滤) 确定文档是否为某些站点的组成部分的过程，这些站点将纳入索引。

finder method (查找方法) 使客户机可以在全局目录中查找一个 bean 或一组 bean 的方法。

firewall (防火墙) 保护组织内的联网计算机免受外部访问的网络配置，通常包括硬件和软件。防火墙一般用于保护物理建筑或组织站点内部的信息，如网络的电子邮件和数据文件等。

flexible log format (灵活日志格式) 服务器用于将信息输入访问日志的格式。

folder (文件夹) 已命名的消息集合。文件夹中可以含有其他文件夹。又称为邮箱。另参见 [personal folder \(个人文件夹\)](#)、[public folder \(公共文件夹\)](#)、[shared folder \(共享文件夹\)](#)、[INBOX](#)。

form action handler (表单操作处理程序) 在 servlet 或应用程序逻辑中特别定义的方法。该方法基于表单上的命名按钮执行操作。

FORTEZZA 美国政府机构使用的一种加密系统，用于管理敏感但未分类的信息。

forwarding (转发) 参见 [message forwarding \(消息转发\)](#)。

foundation profile (基础配置文件) 一组 API 以及 CDC。它们共同提供面向下一代应用程序、消费电子产品和嵌入式设备的 J2ME™ 应用程序运行时环境。

fractional replication (部分复制) 经过过滤的属性子集的复制。

fragmentation (分段) 一种 MIME 功能，可将较大的消息拆分成多个较小的消息。另参见 [default calendar \(默认日历\)](#)。

fresh start (全新启动) 从 robot 的起始点开始启动。全新启动将删除 robot 的状态信息，从而使 robot 从其初始状态开始其下一次运行。与重新启动相对。

FTP (file transfer protocol, 文件传输协议) 一种 Internet 协议。该协议允许文件通过网络从一台计算机传输到另一台计算机。

fully qualified domain name (全限定域名) 系统的全名，包含其主机名及其域名。例如：[example.sun.com](#)，其中的 [example](#) 是（服务器的）主机名，[sun.com](#) 是域名。

gateway (网关) 从一种本机格式转换成另一种本机格式的系统。示例如：往返于 X.400 和 RFC 822 之间的电子邮件网关。连接两个或更多电子邮件系统（特别是两个不同网络中的不同类型的邮件系统）并在其间传输消息的机器。有时，映射和转换会比较复杂。通常需要一个存储/转发方案，消息根据此方案从一个系统完全接收后，才经过适当转换，然后传输到下一个系统。

general access (通用访问) 一种用户访问级别。授予此访问权限后，即表示所有通过验证的用户均可访问目录信息。

general ACL (通用 ACL) Java Enterprise System Directory Server 中的已命名访问控制列表，它将用户或用户组与一个或多个权限相关联。此列表可随意定义和访问，以记录任何权限设置。

generation (生成) robot 运行的一个阶段。在该阶段中，robot 为在枚举阶段搜索的每个资源生成一个资源说明。

generic resource (通用资源) 置于“资源组管理器”控制下，作为通用资源类型组成部分的应用程序守护程序及其子进程。

generic resource type (通用资源类型) 数据服务的模板。通用资源类型可用于使简单的应用程序成为故障转移数据服务（在一个节点停止，在另一个节点开始）。此类型不需要通过 SunPlex™ API 编程。

generic servlet (通用 servlet) 扩展 javax.servlet.GenericServlet 的 servlet。通用 servlet 独立于协议：它们本身不含对 HTTP 或任何其他传输协议的支持。另参见 [HTTP servlet](#)。

GIF (graphics interchange format, 图形交换格式) 最初由 CompuServe 创建的一种跨平台图像格式。GIF 文件通常比其他图形文件类型（例如 TIFF）小很多。GIF 是最常用的交换格式之一。GIF 图像可直接在 UNIX、Microsoft Windows 和 Apple Macintosh 系统上显示。

global database connection (全局数据库连接) 对多个组件可用的数据库连接。需要资源管理器。

global device (全局设备) 从所有群集成员（如磁盘、CD-ROM 和磁带等）均可访问的设备。

global device namespace (全局设备名称空间) 含有全局设备的逻辑、群集范围名称的名称空间。Solaris™ Operating System 中的本地设备在 /dev/dsk, /dev/rdisk 和 /dev/rmt 目录中定义。全局设备名称空间在 /dev/global/dsk、/dev/global/rdisk 和 /dev/global/rmt 目录中定义全局设备。

global interface (全局接口) 物理托管共享地址的网络接口。

global interface node (全局接口节点) 托管全局接口的机器或域。

global resource (全局资源) 在 Java Enterprise System Messaging Server 软件内核层提供的高可用性资源。全局资源可包括磁盘 (HA 设备组)、群集文件系统以及全局网络。

global transaction (全局事务) 由事务管理器管理和协调, 并可跨多个数据库和进程的事务。事务管理器通常使用 XA 协议与数据库后端交互。另参见 [local transaction \(本地事务\)](#)。

GMT (Greenwich Mean Time, 格林威治标准时间) 英格兰格林威治子午线的标准太阳时间, 也是世界上所有其他时区参考的时间标准。GMT 不受“日光节约时间”或“夏时制”的影响。

granularity level (粒度级别) 将应用程序分成多个部分的方法。较高的粒度级别意味着应用程序被划分成许多更小的、定义更具体的 Enterprise JavaBeans™ 组件。较低的粒度级别意味着应用程序被划分成较少的部分, 从而生成较大的程序。

greeting form (问候表单) 通常在为用户创建帐户后向其发送的消息。此表单用于确认新帐户并验证其内容。

group (组) (1) 以某种方式联系在一起的多个用户。组成员资格通常由本地系统管理员维护。另参见 [user \(用户\)](#)、[role \(角色\)](#)。

(2) 排列在一个 [distinguished name \(标识名\)](#) 下面的多个 LDAP 邮件条目。通常用作邮件列表, 但也可用于将某些管理权限授予组内某些成员。另参见 [dynamic group \(动态组\)](#)、[static group \(静态组\)](#)。

group folders (组文件夹) 这些文件夹含有共享文件夹和组文件夹。另参见 [public folder \(公共文件夹\)](#)、[shared folder \(共享文件夹\)](#)。

group ID (组 ID) Calendar Server 文件 (如计数器和日志) 的组。组 ID 存储在 `ics.conf` 文件的 `local.servergid` 参数中。又称为 **GID**。

group scheduling engine (组调度引擎) 处理组调度的 Calendar Server 进程。此引擎使用户可与位于相同服务器或不同服务器上的其他日历用户一起安排事件。然后, 其他用户可以修改、取消或回复该事件。

GUI 图形用户界面。

HA 参见 [high availability \(高可用性\)](#)。

HA data service (HA 数据服务) 参见 [data service \(数据服务\)](#)。

HADB 参见 [high availability database \(高可用性数据库\)](#)。

HADB node (HADB 节点) 一组 HADB 进程、共享内存的专用区域以及用于存储和更新会话数据的一个或多个辅助存储设备。每个活动的 (数据存储) 节点必须拥有一个镜像节点, 因此节点都是成对出现的。另外, 可加入两个或更多备用节点以获得最大可用性。如果一个活动节点发生故障, 并且无法在超时之前恢复正常, 则备用节点会从镜像节点复制数据, 然后成为活动节点。另参见 [high availability database \(高可用性数据库\)](#)、[active node \(活动节点\)](#)、[spare node \(备用节点\)](#)、[mirror node \(镜像节点\)](#) 和 [data redundancy unit \(数据冗余单元\)](#)。

handle (句柄) 标识企业 bean 的对象。客户机可以使句柄序列化, 以后再取消它的序列化, 以获得对相应 bean 的引用。

hard restart (硬重启) 进程或服务的终止及其随后的重新启动。另参见 [SOAP](#)。

hashdir 一种命令行实用程序, 用于确定哪个目录含有某特定用户的消息存储。

HDML (Handheld Device Markup Language, 手持设备标记语言) Openwave 的专有语言, 用于为使用 Openwave 浏览器的移动设备编程。

header (报头) 电子邮件消息中正文前面的部分。报头由字段名及值组成，字段名与值以冒号隔开。报头包含电子邮件程序以及想了解消息的用户所需的信息。例如，报头包括传送信息、内容摘要、跟踪和 MIME 信息。报头表明消息的接收者、发送者、发送时间及主题。报头必须按照 RFC 822 编写，以便电子邮件程序可以读懂。

header field (头字段) 消息头中已命名的信息项，如 “From:” 或 “To:”。又称为头行。

heartbeat (心跳) 在所有可用群集互连传输路径上周期性发送的消息。如果在指定的间隔和重试次数之后未收到心跳信息，可能会触发传输通信的内部 **failover (故障转移)**，使其转到另一条路径。到某一群集成员的所有路径都发生故障将使 **CMM** 重新评估群集法定数目。

heuristic decision (启发式决策) 特定事务使用的事务模式。事务必须 “提交” 或 “回滚”。

high availability (高可用性) 在系统失败或出现处理故障时能够检测到服务中断，并提供恢复机制。另外，高可用性使备份系统在主系统出现故障时能够接管服务。又称为 HA。

high availability database (高可用性数据库) (HADB) 一种高度可伸缩、高度可用的会话状态持久性基础结构。Application Server 使用 HADB 存储 HTTP 会话状态和有状态会话 bean 状态。另参见 **HADB node (HADB 节点)**、**active node (活动节点)**、**spare node (备用节点)**、**mirror node (镜像节点)** 和 **data redundancy unit (数据冗余单元)**。

home interface (主接口) 一种机制，它定义使得客户机能够创建和移除企业 bean 的方法。

home page (主页) 服务器上的一个文档，充当服务器内容的目录或入口点。此文档的位置在服务器的配置文件中定义。

hop (中继站) 两台计算机之间的传输。

horizontal scalability (水平可伸缩性) Calendar Server 既可以在单个服务器上运行，又可以作为一组进程，使用各种可能配置选项在多个服务器上运行的能力。

host (主机) 驻留有一个或多个服务器的计算机。

hosted domain (托管域) 一种外包给 ISP 的电子邮件域。即，ISP 通过为某一组织运行和维护电子邮件服务为该公司提供电子邮件域托管功能。托管域与其他托管域共享同一 Java Enterprise System Messaging Server 主机。在早期基于 LDAP 的电子邮件系统中，域由一个或多个电子邮件服务器主机支持。使用 Messaging Server，单个服务器可以托管许多域。对于每一个托管域，都有一个指向该域的用户和组容器的 LDAP 条目。又称为虚拟托管域或 **virtual domain (虚拟域)**。另参见 **document type definition (文档类型定义)**。

host-IP authentication (主机 IP 验证) 用于限制对 Java Enterprise System Administration Server 或位于 Web 站点上的文件和目录的访问的安全机制，具体实现方法是使它们只对使用特定计算机的客户机可用。

host name (主机名) 域内某一特定计算机的名称。主机名是 IP 主机名，它可以是“简写”的主机名（例如，mail）或全限定主机名。全限定主机名由主机名和 **domain name (域名)** 组成。例如，mail.example.com 是域 example.com 中的主机名 mail。在其域内主机名必须是唯一的。您的组织中可以有名为 mail 的多个计算机，只要这些计算机位于不同的子域中，例如，mail.corp.example.com 和 mail.field.example.com。主机名总是映射到特定的 IP 地址。另参见 **fully qualified domain name (全限定域名)**、**IP address (IP 地址)**。

host-name hiding (主机名隐藏) 使用基于域的电子邮件地址的做法，这些地址不包含特定内部主机的名称。

HTML (hypertext markup language, 超文本标记语言) 一种用于创建可由 Web 浏览器显示的文档的编码标记语言。每一文本块都由表明文本性质的代码包围。

HTML page (HTML 页面) 用 HTML 编码、适于 Web 浏览器显示的页面。

HTTP (hypertext transfer protocol, 超文本传输协议) 基于 **TCP/IP** 的 Internet 协议，用于从远程主机取得超文本对象。

HTTPD (hypertext transfer protocol daemon, 超文本传输协议守护进程) HTTP 守护进程或服务的缩写，它是一个使用 HTTP 协议供应信息的程序。

HTTP-NG (hypertext transfer protocol-next generation, 下一代超文本传输协议) 下一代超文本传输协议。

HTTPS (hypertext transfer protocol secure, 超文本安全传输协议) 使用 [SQL](#) 实现的 HTTP 的安全版本。

HTTP servlet 扩展 `javax.servlet.HttpServlet` 的 servlet。这些 servlet 具有对 HTTP 协议的内置支持。另参见 [generic servlet \(通用 servlet\)](#)。

hub (集线器) 充当系统的单个接触点的主机。例如，两个网络由防火墙分隔时，防火墙计算机通常充当邮件集线器。

hub supplier (集线器供应者) 在 [replication \(复制\)](#) 条件下，指存放从其他服务器复制的 [replica \(副本\)](#) 并将它复制到另一台服务器的服务器。另参见 [cascading replication \(级联复制\)](#)。

hypertext transfer protocol secure (超文本安全传输协议) 参见 [HTTP-NG](#)。

iCalendar Message-Based Interoperability Protocol (基于 iCalendar 消息的互操作性协议) 此协议指定从 [iCalendar Transport-Independent Interoperability Protocol \(独立于 iCalendar 传输的互操作性协议\)](#) 到基于 Internet 电子邮件传输的绑定。此协议又称为 iMIP。iMIP 在 RFC 2447 中定义。

iCalendar Transport-Independent Interoperability Protocol (独立于 iCalendar 传输的互操作性协议) 一种基于 iCalendar 对象规范的 Internet 协议，它提供不同日历系统之间的调度互操作性。此协议又称为 iTIP。iTIP 在 RFC 2446 中定义。

IDE (integrated development environment, 集成开发环境) 允许从单个图形用户界面创建、组合、部署和调试代码的软件。

IDENT 参见 [Identification Protocol \(标识协议\)](#)。

Identification Protocol (标识协议) 这种协议提供了确定远程进程的身份的方法，适用于负责特定 TCP 连接的远端的远程进程。又称为 IDENT。在 RFC 1413 中定义。

identity federation (身份联合) 参见 [account federation \(帐户联合\)](#)。

identity provider (身份认证提供者) 专门提供验证服务的提供者。作为验证的管理服务，身份认证提供者维护并管理身份验证信息。对于身份认证提供者提供的验证，其所有联合服务提供者都认可。

identity service (身份认证服务) 身份认证服务是一种作用于资源的 Web 服务，可以检索、更新与“主体”(身份)有关的数据属性或者对这些属性执行某些操作。身份认证服务的示例有公司电话簿或日历服务。

IDL (interface definition language, 接口定义语言) 用于定义到远程 **CORBA** 对象的接口的语言。这些接口独立于操作系统和编程语言。它描述远程过程调用 (**RPC**) 的功能接口，以便编译程序能够生成在计算机之间整理参数的代理与存根代码。

idle state (闲置状态) 一种状态。在此状态下，robot 仍然在运行，但是已经处理了其 URL 池中的所有 URL。此时，robot 仍然可以对状态请求做出响应。

iHTML (i-mode hypertext markup language, i-mode 超文本标记语言) NTT DoCoMo 的日语 i-mode 服务使用的语言。

IIOP (Internet Inter-ORB Protocol, Internet ORB 互联协议) **RMI over IIOP** (IIOP 上的远程方法调用) 和 **CORBA** (公共对象请求代理体系结构) 均使用的一种传输层协议。

IIOP cluster (IIOP 群集) 为获得 RMI/IIOP 请求的高可用性而配置的 IIOP 群集。

IIOP endpoint (IIOP 端点) 为 IIOP 群集配置的 IIOP 侦听器，其目的是实现 RMI/IIOP 请求的高可用性。

IIOP listener (IIOP 侦听器) 侦听指定端口并接受来自基于 CORBA 的客户机应用程序的外来连接的侦听套接口。

imagemap (热点图) (1) 一种激活图像某些区域的进程，使用户可通过用鼠标单击图像的不同区域来浏览并获取信息。

(2) 一种 **CGI** 程序，用于执行其他 **HTTPD** 实现中的热点图功能。

IMAP4 (Internet Message Access Protocol Version 4, Internet 消息访问协议第 4 版) 一种标准协议, 使用户可从主讯息传递系统断开连接而仍然能够处理其邮件。IMAP 规范允许对这些断开连接的用户进行管理控制, 并且在他们重新连接到讯息传递系统时同步他们的消息存储。

iMIP 参见 [iCalendar Message-Based Interoperability Protocol](#) (基于 iCalendar 消息的互操作性协议)。

immediate subordinate (直接下属) 在 *DIT* 中, 如果某个条目的 **distinguished name (标识名)** 是通过将其 *RDM* 附加到另一条目 (父条目) 的标识名得到的, 则此条目是该父条目的直接下属。

immediate superior (直接上级) 在 *DIT* 中, 如果某个条目的 **distinguished name (标识名)** 附加另一条目 (子条目) 的 *RDM* 后即构成该子条目的标识名, 则此条目是该子条目的直接上级。

import agent (导入代理) **importing (导入)** 期间使用的进程。

importing (导入) 将新的或更新的资源描述从另一数据库中引入“搜索引擎”的过程。

imsadmin command (imsadmin 命令) 用于管理域管理员、用户和组的一组命令行实用程序。

imsimta command (imsimta 命令) 用于执行各种 *MTA* 维护、测试和管理任务的一组命令行实用程序。

inactive boot environment (非活动引导环境) 当前未引导或未被指定为在下次重新引导时激活的环境。另参见 [active boot environment \(活动引导环境\)](#)。

INBOX 用户默认邮箱的保留名称。用于邮件传送。INBOX 是唯一的不区分大小写的文件夹名, 即 INBOX、Inbox 和 inbox 都是用户默认邮箱的有效名称。

index (索引) 资源或文档的集中化的并可搜索的数据库。又称为目录。

indexing (建立索引) 为资源提供集中化并可搜索的数据库的过程。又称为编目。

indexing (编排索引) 提供一个内容集中的、可搜索的资源数据库的过程。又称为编目录。

indirect CoS (间接 CoS) 用一个目标条目属性的值标识模板条目。

inittab file (inittab 文件) (仅限 UNIX) 列出由于任何原因停止后需要重新启动的程序的文件。该文件确保程序连续运行。由于其位置的关系, 此文件又称为 `/etc/inittab`。并非所有 UNIX 系统上都有此文件。

installation directory (安装目录) 安装服务器的二进制 (可执行) 文件的目录。对于 Messaging Server, 安装目录是 **server root (服务器根目录)** 的子目录: `server-root/bin/msg/`。另参见 **instance directory (实例目录)**。

instance directory (实例目录) 包含定义服务器的特定实例的文件的目录。对于 Messaging Server, 实例目录是 **server root (服务器根目录)** 的子目录: `server-root/msg-instance/`, 其中 *instance* 是在安装时指定的服务器名称。另参见 **installation directory (安装目录)**。

Instant Messaging multiplexor (Instant Messaging 多路复用器) 一个客户机连接管理器。它使大量并发客户机连接只需使用少量到后端 Instant Messaging 服务器的连接, 从而提高 Instant Messaging Server 的可伸缩性。Instant Messaging 客户机连接到此多路复用器, 而不是连接到 Instant Messaging 服务器本身。安装在防火墙的公共侧时, 此多路复用器保护用户数据库不受入侵者的侵犯, 将 Instant Messaging Server 保护在防火墙之后。

Instant Messaging Server (1) 指 Java Enterprise System Messaging Server 产品自身, 包括所有组件 (服务器、多路复用器和 Java Enterprise System Instant Messaging Server)。

(2) 该产品的后端服务器进程, 用于处理来自 Instant Messaging (通过 Instant Messaging Server 多路复用器) 的外来命令。验证 Instant Messaging 用户时, Instant Messaging Server 还与 LDAP 服务器通信。另参见 **Instant Messaging multiplexor (Instant Messaging 多路复用器)**。

intelligent agent (智能代理) 服务器中代表用户执行各种请求 (如 HTTP、NNTP、SMTP 和 FTP 请求) 的对象。在某种意义上, 智能代理充当服务器的客户机, 发出服务器完成的请求。

international index (国际索引) 一种搜索索引。可快速搜索 *DIT* (其中的属性有语言标记) 中的信息。

Internet Message Access Protocol Version 4 (Internet 消息访问协议第 4 版)

参见 *IMAP4*。

Internet Protocol (Internet 协议) 参见 [IP](#)。

intranet (内联网) 某一公司或组织内多个 [TCP/IP](#) 网络连成的网络。对于分布在公司 LAN 上的内部应用程序，内联网使公司能够使用相同类型的万维网服务器和客户机软件。内联网上与 Internet 通信的敏感信息通常用防火墙加以保护。另参见 [firewall \(防火墙\)](#)、[extranet \(外联网\)](#)。

invalid user (无效用户) 消息处理过程中出现的错误情况。出现此错误情况时，消息存储向 MTA 发送通信，然后删除它拥有的消息副本。MTA 将消息弹回发送者并删除其拥有的消息副本。

IP (Internet Protocol, Internet 协议) [TCP/IP](#) 族中用于在世界范围内链接网络的协议。由美国国防部开发，并在 Internet 中采用。此协议族的突出点是 IP 协议。

IP address (IP 地址) 由点分隔的一组数字，如 192.168.255.255，指定内联网或 Internet 上某台计算机的实际位置。是分配给使用 [TCP/IP](#) 的主机的 32 位地址。

ISDN 综合业务数字网。

ISINDEX 在客户机中启用搜索的 HTML 标记。文档可以使用网络导航器的功能来接受搜索字符串，然后将其发送到服务器以访问可搜索索引，无需使用表单。要使用 ISINDEX HTML 标记，必须创建一个查询处理程序。

ISMAP HTML 文档中 `IMG SRC` 标记的扩展，用于告知服务器指定的图像为 [imagemap \(热点图\)](#)。

ISO 8601 一个“国际标准化组织”标准，用于指定日期与时间的数字表示。Calendar Server 使用 ISO 8601 标准表示法表示日期、时间和持续时间字符串。

isolation level (隔离级别) 参见 [transaction isolation level \(事务隔离级别\)](#)。

issued certificate (已颁发证书) 由 (certificate authority) ([证书授权机构](#)) 颁发的证书。另参见 [self-generated certificate \(自生成证书\)](#)。

iTIP 参见 [iCalendar Transport-Independent Interoperability Protocol \(独立于 iCalendar 传输的互操作性协议\)](#)。

J2EE™ platform (J2EE™ 平台) (Java™ 2 Platform, Enterprise Edition) 用于开发和部署多层、基于 Web 的企业应用程序的环境。J2EE 平台由提供这些应用程序开发功能的一组服务、[API](#) 和协议组成。

J2ME™ platform (J2ME™ 平台) (Java™ 2 Platform, Micro Edition) 适用于移动设备的小型应用程序环境。

JAF (JavaBeans™ Activation Framework, JavaBeans™ 激活框架) 将 MIME 数据类型支持集成到 Java 平台中。另参见 [MIME data type \(MIME 数据类型\)](#)。

JAR file (JAR 文件) (Java™ Archive file, Java™ 归档文件) 用于将许多文件聚合到一个文件的文件。JAR 文件具有扩展名 .jar。

JAR file contract (JAR 文件订约) Java™ 归档文件订约, 指定企业 bean 软件包中必须包含哪些信息。

JAR file format (JAR 文件格式) 一种独立于平台的 Java™ 归档文件格式, 它将许多文件聚合到一个文件中。多个 applet 及其必备组件 (类文件、图像、声音以及其他资源文件) 可以捆绑到一个 JAR 文件中, 然后通过单个 HTTP 事务下载到浏览器。JAR 文件格式还支持文件压缩和数字签名。

JATO 用于转换使用 Java 编程语言编写的代码和使用 [XML](#) 编写的代码的库。又称为 Sun Java System Web Application Framework 和 Application Framework。JATO 适合于企业 Web 应用程序开发。JATO 结合了多种概念, 如显示字段、应用程序事件、组件分层结构以及以页面为中心的开发方法。

Java™ 由 Sun Microsystems, Inc. 开发的面向对象、独立于平台的编程语言, 其目的是解决现代编程实践中存在的许多问题。

Java™ 2 Platform, Enterprise Edition 参见 [J2EE™ platform \(J2EE™ 平台\)](#)。

Java™ 2 Platform, Micro Edition 参见 [J2ME™ platform \(J2ME™ 平台\)](#)。

JavaBean™ namespace (JavaBean™ 名称空间) 一种标准，允许为由某一软件包定义的元素名称集指定唯一的标签。使用该软件包的文档可以加入任何其他文件，而不会造成元素名称间的冲突。在软件包中定义的元素将被唯一地标识，以便解析程序可以根据您的软件包而不是其他软件包确定何时解释一个元素。

JavaBeans™ Activation Framework (JavaBeans™ 激活框架) 参见 [JAF](#)。

JavaBeans™ component architecture (JavaBeans™ 组件体系结构) 一种可移植的、独立于平台的可重用组件模型。

Java™ Development Kit (Java™ 开发工具包) 参见 [JDK™ software \(JDK™ 软件\)](#)。

Java™ Enterprise System 由单个 Sun 软件产品集成的软件系统，它支持分布式企业应用程序。

Java™ IDL (API, 扩展) (Java Interface Definition Language, Java 接口定义语言) 用 Java 编程语言编写的 *API*，它提供与 *CORBA* 的、基于标准的兼容性和连通性。

JavaMail™ (API, 扩展) 应用程序用于与邮件存储交互的对象。应用程序代码使用 Java Naming and Directory Interface™ (JNDI) 服务查找采用 JNDI 名称的 JavaMail 会话资源对象。

JavaScript™ programming language (JavaScript™ 编程语言) 一种简洁的、基于对象的脚本语言，用于开发客户机和服务器 Internet 应用程序。

JavaServer Pages™ technology (JavaServer Pages™ 技术) 参见 [JSP™ technology \(JSP™ 技术\)](#)。

Java™ System Compass Server 用于方便用户访问网络资源的服务器技术，通常与 Portal Server 3.0 一同使用。Portal Server 6.0 包含一个紧密集成的搜索引擎，该引擎具有 Compass Server 对 Portal Server 3.0 提供的功能。

Java™ System Delegated Administrator 一组 GUI 和 CLI 界面，它们使管理员能够向托管域添加用户以及修改托管域中目录的用户和组。

Java™ System Directory Server *LDAP* 的 Java Enterprise System 版本。Application Server 的每一个实例都使用 Directory Server 来存储共享的服务器信息，包括用户和组的有关信息。

Java™ System Directory Server Access Management Edition 一组界面，为 Portal Server 提供用户和服务管理、验证和单点登录服务、策略管理、日志记录服务、调试实用程序和客户机支持。

Java™ System Instant Messaging Client 使用户能够发送和接收即时消息和警报的客户机。

Java™ System Message Queue 实现 Java™ 消息服务 (Java™ Message Service, JMS) 开放标准的讯息传递系统。Message Queue 是一种 JMS 提供者。

Java™ System Portal Server 使远程用户能够通过 Internet 安全地访问其组织的网络和网络服务的软件产品。它创建一个安全的 Internet 门户，使所有目标受众（包括员工、业务伙伴或一般公众）都能访问相应内容、应用程序和数据。被认为是整个 Portal Server 产品解决方案的核心部分，由所有 Portal Server 包共享。

Java™ System Portal Server Desktop 它提供主终端用户界面，以及通过内容提供者接口 (PAPI) 实现可扩展内容聚合的机制。通常又称为“桌面”。“桌面”包括许多提供者，它们提供构建某些类型频道所需的容器分层结构和基本构件。“桌面”在 Directory Server Access Management Edition 服务之上实现一种显示配置数据存储机制，用于存储内容提供者和频道数据。“桌面”还包括一个管理控制台模块，用于编辑显示配置及其他“桌面”服务数据。

Java™ System Portal Server Instant Collaboration Pack 一种服务器即时讯息传递产品，包括服务器、multiplexor（多路复用器）和 Instant Messaging 组件。又称为 Java System Instant Messaging Server。

Java™ System Portal Server Pack 指代 Portal Server 附加产品的一个通用术语。

Java™ System Web Server Portal Server 中的一种 Web 服务器，用作 Portal Server 和 Portal Server 包 Web 应用程序的 Web 容器。Web Server 随附在 Directory Server Access Management Edition 产品中。

Java™ Web Start software (Java™ Web Start 软件) 一种 Web 应用程序启动程序。使用 Java Web Start 软件，单击 Web 链接即可启动应用程序。如果应用程序不在计算机上，Java Web Start 会自动下载该应用程序并将它缓存到计算机中。应用程序下载到其缓存区后，即可从桌面图标或浏览器链接启动。无论使用哪种方法启动应用程序，始终都会显示应用程序的最新版本。

JAXM (Java™ API for XML Messaging, 用于 XML 讯息传递的 Java™ API) 一种使用 SOAP 标准的使应用程序能够发送和接收面向文档的 XML 消息的 Java API。这些消息可以有附件，也可以没有附件。

JAXP (Java™ API for XML Processing, 用于 XML 处理的 Java™ API) 一种使用 DOM、SAX 和 XSLT 来支持 XML 文档处理的 Java API。它使应用程序能够独立于特定的 XML 处理实现方式对 XML 文档进行解析和转换。

JAXR (Java™ API for XML Registries, 用于 XML 注册表的 Java™ API) 一种统一的标准 Java API，用于访问不同种类的 XML 注册表。它使用户能够构建、部署和搜索 Web 服务。

JAX-RPC (Java™ API for XML-based RPC, 用于基于 XML 的 RPC 的 Java™ API) 一种 Java API，使开发者能够根据基于 XML 的 *RPC* 协议，构建可交互使用的 Web 应用程序和 Web 服务。

JDBC™ connection pool (JDBC™ 连接池) 将用于指定数据库连接的 JDBC 数据源属性与连接池属性结合在一起的池。

JDBC™ resource (JDBC™ 资源) 一种资源，用于通过现有 JDBC 连接池，将应用程序服务器中运行的应用程序连接到数据库。由 Java Naming and Directory Interface™ (JNDI) 名称（供应用程序使用）和一个现有 JDBC 连接池的名称组成。

JDBC™ technology (JDBC™ 技术) (Java™ DataBase Connectivity software, Java™ 数据库连接软件) 一组基于标准的类和接口，使开发者能够创建数据感知组件。JDBC API 以一种独立于平台和供应商的方式，实现了连接到数据源以及与数据源交互的方法。

JDK™ software (JDK™ 软件) (Java™ Development Kit, Java™ 开发工具包) 用于编写 Java applet 或应用程序的软件工具。

JHTML (J-Sky hypertext markup language, J-Sky 超文本标记语言) Vodafone 专有的语言, 用于为日语 J-Sky 设备编程。

JMS (Java™ Message Service, Java™ 消息服务) 定义 Java 客户机如何访问消息服务设备的一套标准接口和语义。这些接口为使用 Java 编程语言编写的程序提供了一种创建、发送、接收和读取消息的标准方式。

JMS-administered object (JMS 管理对象) (Java™ Message Service administered object, Java™ 消息服务管理对象) 由管理员创建的预配置 Java™ 消息服务对象 **JMS connection factory (JMS 连接工厂)** 或 **JMS destination (JMS 目标)**, 供一个或多个 JMS 客户机使用。使用受管理对象可以使 JMS 客户机从提供者的拥有权中脱离出来, 从而使客户机独立于提供者。这些对象由管理员置入一个 Java Naming and Directory Interface™ (JNDI) 名称空间中, 并由 JMS 客户机使用 JNDI 查找进行访问。

JMS API (Java™ Message Service API, Java™ 消息服务 API) 定义 JMS 客户机如何访问 JMS 消息服务设备的一套标准接口和语义。这些接口为使用 Java 编程语言编写的程序提供了一种创建、发送、接收和读取消息的标准方式。

JMS client (JMS 客户机) (Java™ Message Service client, Java™ 消息服务客户机) 一种应用程序或软件组件, 使用 JMS 消息服务与其他 JMS 客户机进行交互, 以交换消息。

JMS connection factory (JMS 连接工厂) (Java™ Message Service connection factory, Java™ 消息服务连接工厂) 由 Java™ 消息服务管理的对象, JMS 客户机使用该对象创建与 JMS 消息服务的连接。

JMS destination (JMS 目标) (Java™ Message Service destination, Java™ 消息服务目标) JMS 消息服务中的物理目标, 生成的消息被传送到此目标, 以进行路由选择和继续传送给使用者。此物理目标由一个 JMS 管理对象标识和封装, JMS 客户机使用该对象指定收到的消息和外出的消息的目标。

JMS messages (JMS 消息) (Java™ Message Service messages, Java™ 消息服务消息) 由 Java™ 消息服务客户机使用的异步请求、报告或事件。消息包含报头 (可以向其中添加附加字段) 和正文。消息头指定标准字段和可选属性。消息正文含有要发送的数据。

JMS provider (JMS 提供者) (Java™ Message Service provider, Java™ 消息服务提供者) 一种为消息传递系统提供 JMS 接口, 并添加完整产品所需的管理和控制功能的产品。

JMS service (JMS 服务) (Java™ Message Service service, Java™ 消息服务的服务) 为 Java™ 消息服务消息传递系统提供传送服务 (包括与 JMS 客户机的连接、消息路由选择和传送、持久性、安全性和登录) 的软件。消息服务维护的物理目标接收 JMS 客户机发送的消息, 而该目标的消息被发送至使用该消息的客户机。

JNDI extension (JNDI 扩展) (Java Naming and Directory Interface™ extension, Java Naming and Directory Interface™ 扩展) Java 平台的一种标准扩展, 它为支持 Java 技术的应用程序提供使用企业中多个命名和目录服务的统一接口。作为 Java™ Enterprise API 集的组成部分, JNDI 可使各种不同的企业命名和目录服务相互连接。

JNDI name (JNDI 名称) (Java Naming and Directory Interface™ name, Java Naming and Directory Interface™ 名称) 用于访问已在 JNDI 命名服务中注册的资源名称。

job controller (作业控制器) 负责在接到各种其他 MTA 组件请求时调度和执行任务的 MTA 组件。

JRE (Java™ runtime environment, Java™ 运行时环境) **JDK™ software (JDK™ 软件)** 的一个子集, 该子集由 Java™ 虚拟机、Java 核心类和支持文件组成。为使用 Java 编程语言编写的应用程序提供运行时支持。

jspc utility (jspc 实用程序) 使用 **JSP™ technology (JSP™ 技术)** 创建的页面的编译程序。该实用程序检查所有 JSP 页面是否符合 JSP 规范。

JSP™ technology (JSP™ 技术) (1) 将 HTML 或 XML 标记、JSP 标记以及用 Java™ 编程语言编写的代码结合使用而编写的文本页。使用 JSP 技术创建的页面将标准浏览器页面的布局功能和编程语言的强大功能结合在一起。

(2) 支持所有 JavaServer Pages 技术元功能 (包括实例化、初始化、解构、从其他组件访问以及配置管理) 的扩展。在 Web 服务器而非 Web 浏览器上运行的可再度使用的 Java 应用程序。

JSS 参见 [Network Security Services for Java \(面向 Java 的网络安全服务\) \(JSS\)](#)。

JTA (Java transaction API, Java 事务 API) 允许应用程序和 J2EE™ 服务器访问事务的 API。

JTS (Java transaction service, Java 事务服务) 用于处理事务的 Java 服务。

key database (密钥数据库) 含有服务器的一个或多个证书的一个或多个密钥对的文件。也称为密钥文件。

key-pair file (密钥对文件) 参见 **trust database (信任数据库)**。

knowledge information (知识信息) 目录服务基础结构信息的组成部分。目录服务器使用知识信息将对信息的请求传递给其他服务器。

knowledge reference (知识引用) 指向存储在不同数据库中的目录信息的指针。

last-modified header (上次修改报头) 在来自服务器的 HTTP 响应中返回的文档文件的最后修改时间。

LDAP (Lightweight Directory Access Protocol, 轻量目录访问协议) 专用于通过 TCP/IP 并跨多个平台运行的目录服务协议。X.500 目录访问协议 (Directory Access Protocol, DAP) 的简化形式, 允许对信息 (包括跨多个 Java System 服务器的用户概要文件、分发列表以及配置数据) 的存储、检索、分发进行单点管理。Directory Server 使用 LDAP 协议。

LDAP database (LDAP 数据库) 存储在验证中使用的用户和组列表的数据库。

LDAP data interchange format (LDAP 数据交换格式) 参见 **LDIF**。

LDAP filter (LDAP 过滤器) 根据存在的特定属性或属性值指定条目集的方法。

LDAP referrals (LDAP 引用) 含指向另一 LDAP 条目的符号链接 (引用) 的 LDAP 条目。LDAP 引用由一个 LDAP 主机和一个标识名组成。LDAP 引用通常用于引用现有 LDAP 数据, 从而不必复制该数据。LDAP 引用还用于当程序所依赖的特定条目已经移动时, 保持该程序的兼容性。

LDAP search string (LDAP 搜索字符串) 含有可替换参数的字符串, 其中的参数定义用于目录搜索的属性。例如, LDAP 搜索字符串 “uid=%s” 表示搜索基于用户 ID 属性。

LDAP server (LDAP 服务器) 一种软件服务器，可维护 LDAP 目录并为目录查询提供服务。“Sun 目录服务”和“Netscape 目录服务”是“LDAP 服务器”的实现。

LDAP server failover (LDAP 服务器故障转移) LDAP 服务器的一种备份功能。如果一台 LDAP 服务器发生故障，系统可以切换到另一台 LDAP 服务器。

LDAP URL 一种 [URL](#)，它提供使用 [DNS](#) 查找目录服务器，然后通过 LDAP 完成查询的方法。ldap://ldap.example.com 是一个 LDAP URL 样例。

LDAPv3 第 3 版的 [LDAP](#) 协议。

LDBM LDAP 数据库管理器。

LDBM database (LDBM 数据库) 一种基于磁盘的高性能数据库，该数据库由包含 Directory Server 中所有数据的一组大文件组成。

LDIF (LDAP Data Interchange Format, LDAP 数据交换格式) 该格式使用 *type:value* 对以文本形式表示 Directory Server 条目。

leaf entry (叶条目) 下面没有其他条目的条目。叶条目不能是目录树中的分支点。

Legato NetWorker® software (Legato NetWorker® 软件) 由 Legato Systems, Inc. 分发的第三方备份实用程序。

level (级别) 日志冗余的名称，表示日志文件中记录的事件类型的相对数量。例如，“紧急”级别将记录很少的事件。而“信息”级别将记录很多事件。

Liberty-enabled client (启用 Liberty 的客户机) 启用 Liberty 的客户机是拥有（或知道如何获得）有关身份认证提供者信息的客户机，主体将使用这些信息对服务提供者身份进行验证。

Liberty-enabled proxy (启用 Liberty 的代理) 启用 Liberty 的代理是模拟启用 Liberty 的客户机的 HTTP 代理。

life-cycle event (生命周期事件) 服务器生命周期的一个阶段，如启动或关闭。

life-cycle module (生命周期模块) 侦听并执行其任务以响应服务器生命周期中的事件的事件的模块。

Lightweight Directory Access Protocol (轻量目录访问协议) 参见 [LDAP](#)。

listener (侦听器) 在发布对象中注册的类，说明事件发生时执行的操作。

listen port (侦听端口) 服务器用于与客户机和其他服务器通信的端口。

listen socket (侦听套接字) **port (端口)** 号与 **IP address (IP 地址)** 的组合。服务器和客户机通过侦听套接字连接。

LMTP (Local Mail Transfer Protocol, 本地邮件传输协议) 与 **SMB protocol (SMB 协议)** 类似，但不需要邮件传送队列管理。另外，LMTP 为消息的每个收件人都提供一个状态代码，而 SMTP 仅为消息提供一个状态代码。在 RFC 2033 中定义。

load balancer (负载均衡器) 一种软件，可控制与多台网关机器的连接，使每个可用系统上的负载基本相等。

load balancing (负载均衡) 跨群集中多个节点分配应用程序负载从而使客户机请求及时得到服务的过程。仅适用于可伸缩服务。

load-balancing policy (负载均衡策略) 跨节点分配应用程序请求负载的首选方式。仅适用于可伸缩服务。

local database connection (本地数据库连接) 本地连接中的事务上下文对当前进程以及当前数据源是本地关系，而不是跨进程或跨数据源分布。

local disk (本地磁盘) 特定群集节点在物理上专有的磁盘。

locale (语言环境) 确定整理顺序、字符类型、货币格式以及日期和时间格式的一种设置，用于表示特定地域、文化或习俗的用户的数据。语言环境包括如何解释、存储或整理特定语言的数据的信息。语言环境还指示应使用哪个代码页表示一种特定语言。

local interface (本地接口) 一种接口，该接口通过会话或实体 bean 为位于相同 Java™ 虚拟机 (JVM™ 机器) 中的客户机提供访问该 bean 的机制。

Local Mail Transfer Protocol (本地邮件传输协议) 参见 [LMTP](#)。

local part (本地部分) 电子邮件地址中标识收件人的部分。另参见 [domain part \(域部分\)](#)。

local session (本地会话) 仅对一台服务器可见的用户会话。

local transaction (本地事务) 源于一个数据库并且仅限于单个进程内的事务。本地事务仅对单一后端有效。本地事务通常使用 JDBC™ API 界定。另参见 [global transaction \(全局事务\)](#)、[JDBC™ technology \(JDBC™ 技术\)](#)。

log directory (日志目录) 保存一个服务的所有日志文件的目录。

log expiration (日志失效期) 当日志文件达到允许的最长使用期限后，将从日志目录中删除。

logical architecture (逻辑体系结构) 一种描述分布式应用程序的逻辑构建块以及这些构建块之间的关系（或接口）的设计。逻辑体系结构包括分布式 [application component \(应用程序组件\)](#) 以及支持这些组件所需的基础结构服务组件。

logical host (逻辑主机) Messaging Server 2.0 (最低) 的一个概念，它包括一个应用程序、存放该应用程序数据的磁盘集或磁盘组以及用于访问群集的网络地址。SunPlex™ 系统中不再有此概念。

logical host name (逻辑主机名) 含有代表网络地址的逻辑主机名集合的资源。逻辑主机名资源一次只能由一个节点控制。另参见 [logical host \(逻辑主机\)](#)。

logical network interface (逻辑网络接口) 在 Internet 体系结构中，一个主机可以有多个 IP 地址。Messaging Server 可配置更多的逻辑网络接口，用以在多个逻辑网络接口和单个物理网络接口之间建立映射。每个逻辑网络接口拥有一个单独的 IP 地址。此映射可以使单个物理网络接口响应多个 IP 地址。此映射还可以在接管或转切时使 IP 地址从一个群集成员移动到另一个成员，而无需更多的硬件接口。

log rotation (日志轮转) 创建新日志文件作为当前日志文件的过程。所有后续记录的事件都将写入新的当前文件中。以前的日志文件不再记录事件，但仍然保留在日志目录中。

lookup (查找) 与搜索相同，使用指定的参数筛选数据。

mailbox (邮箱) 存储和查看消息的位置。另参见 **folder (文件夹)**。

mail client (邮件客户机) 帮助用户发送和接收电子邮件的程序。邮件客户机是各种网络和邮件程序的组成部分，用户与其有很多联系。邮件客户机创建和提交要传送的消息，检查新的收到的邮件，并接受和组织收到的邮件。

mail exchange record (邮件交换记录) 参见 **MX record (MX 记录)**。

mailing list (邮递列表) 参见 **mailing list (邮递列表)**。

mailing list owner (邮递列表所有者) 参见 **mail list owner (邮寄列表所有者)**。

mail list (邮寄列表) 电子邮件地址列表，可利用邮寄列表地址向这些地址发送消息。有时称为组。

mail list owner (邮寄列表所有者) 拥有向邮寄列表添加成员或从其中删除成员的管理权限的用户。

mail relay (邮件转发) 从一个 **user agent (用户代理)** 或 **MTA** 接收邮件，然后将其转发给邮件收件人的消息存储或其他路由器的邮件服务器。

mail router (邮件路由器) 参见 **mail relay (邮件转发)**。

managed object (受管理对象) 构成 **MIB** 组成部分的 **SNMP** 数据元素。在 **Directory Server** 中，受管理对象存放在 **cn=monitor** 中，并且 **SNMP** 代理向网络管理站提供对象。与 **LDAP** 属性相同，每个受管理对象都有一个点号形式的名称和对象标识符。

managed role (受管理角色) 允许创建显式的成员枚举列表。

management information base (管理信息库) 参见 **MIB**。

mapping (映射) (1) 将面向对象的模型与数据关系模型（通常是关系型数据库的模式）关联的能力。将一个模式转换成不同结构的过程。

(2) 用户到安全角色的映射应用。

mapping tree (映射树) 将后缀名称（子树）与数据库关联的数据结构。

master (主站) 参见 [primary \(主节点\)](#)。

master agent (主代理) 参见 [SNMP master agent \(SNMP 主代理\)](#)。

master channel program (主通道程序) 通常启动向远程系统传输的通道程序。另参见 [slave channel program \(从通道程序\)](#)。

master directory server (主目录服务器) 含有要复制的数据的目录服务器。

matching category (匹配类别) 与作为搜索提交的结果而返回的搜索查询相匹配的类别。

matching document (匹配文档) 与作为搜索提交的结果而返回的搜索查询相匹配的文档。

matching rule (匹配规则) 关于服务器在搜索操作期间如何比较字符串的指导原则。在国际搜索中，匹配规则使服务器知道使用何种整理顺序和运算符。

MD5 由 RSA Data Security 开发的消息摘要算法。MD5 可用于生成极有可能唯一的简短数据摘要。

MD5 signature (MD5 签名) 由 [MD5](#) 算法生成的消息摘要。

MDB (message-driven bean, 消息驱动的 bean) 作为异步消息使用者的企业 bean。消息驱动的 bean 没有特定客户机的状态，但其实例变量可能含有跨客户机消息处理状态，包括开放式数据库连接和对基于 EJB™ 体系结构的对象的对象引用。客户机为将消息驱动的 bean 作为消息侦听器的目标发送消息，以此方式来访问该消息驱动的 bean。

member (成员) 接收以邮件列表作为地址的电子邮件副本的用户或组。另参见 [mailing list \(邮递列表\)](#)、[expansion \(扩散\)](#)、[moderator \(仲裁人\)](#)。

message (消息) 电子邮件的基本组成部分，它由一个 [header \(报头\)](#) 和一个 [body \(正文\)](#) 组成，并且通常在从发件人向收件人传输的过程中包含在一个 [envelope \(信封\)](#) 中。

message access services (消息访问服务) 支持客户机访问 Messaging Server 消息存储的协议服务器、软件驱动程序和库。

message delivery (消息传送) 当 *MTA* 将消息传送到本地收件人 (邮件文件夹或程序) 时发生的操作。

message-driven bean (消息驱动 bean) 参见 *MDB*。

message forwarding (消息转发) 当 *MTA* 将传送到特定帐户的消息发送到该帐户属性指定的一个或多个新目标时发生的操作。转发可由用户配置。另参见 [message delivery \(消息传送\)](#)、[message routing \(消息路由选择\)](#)。

message handling system (消息处理系统) 参见 *MHS*。

message queue (消息队列) 一种目录, 从客户机或其他邮件服务器接收的消息在该目录中排队, 等待立即或延迟传送。

Message Queue client runtime (Message Queue 客户机运行时环境) 为 JMS 客户机提供与 Java Enterprise System 消息服务器的接口的软件。客户机运行时环境支持客户机向目标发送消息以及从这些目标接收消息所需的全部操作。

Message Queue message server (Message Queue 消息服务器) 为 Message Queue 消息传递系统提供传送服务 (包括与 JMS 客户机的连接、消息路由选择和传送、持久性、安全性和登录) 的软件。消息服务器维护的物理目标接收 JMS 客户机发送的消息, 而该目标的消息被发送至使用该消息的客户机。

message quota (消息配额) 该限制定义特定文件夹可使用的磁盘空间大小。

message routing (消息路由选择) 当第一个 *MTA* 确定收件人不是本地帐户但可能存在于其他位置时, 将消息从一个 *MTA* 传输到另一个 *MTA* 的操作。路由选择通常只能由网络管理员配置。另参见 [message forwarding \(消息转发\)](#)。

messages (消息) JMS 客户机使用的异步请求、报告或事件。消息包含报头 (可以向其中添加附加字段) 和正文。消息头指定标准字段和可选属性。消息正文含有要发送的数据。

message selector (消息选择器) 使用者根据 JMS 消息头中的属性值 (选择器) 选择消息的方法。消息服务根据消息选择器中置入的条件执行消息过滤和路由选择。

message service (消息服务) 参见 [Message Queue message server \(Message Queue 消息服务器\)](#)。

message store (消息存储) Messaging Server 实例的所有本地传送消息的数据库。消息可以存储于单个物理磁盘上，也可以跨多个物理磁盘存储。

message store administrator (消息存储管理员) 拥有管理某个 Messaging Server 安装的消息存储的管理权限的用户。该用户可以查看和监视邮箱，并指定对该存储的访问控制。该用户可以使用代理授权权限运行某些实用程序，以管理消息存储。

message store partition (消息存储分区) 驻留在单个物理文件系统分区中的消息存储或消息存储的子集。

message submission (消息提交) 客户机 [user agent \(用户代理\)](#) 将消息传输给邮件服务器，并请求传送。

message transfer agent (消息传输代理) 参见 [MTA](#)。

messaging (讯息传递) 由企业应用程序使用的一种异步请求、报告或事件系统，该系统可以使松散耦合应用程序可靠而安全地传输信息。

Messaging Multiplexor (讯息传递多路复用器) 参见 [MMP](#)。

Messaging Server administrator (Messaging Server 管理员) 拥有 Messaging Server 实例的安装和管理等权限的管理员。

messaging server base directory (讯息传递服务器基目录) 用于安装所有与特定主机上的特定 Administration Server 关联的服务器的目录。通常命名为 *msg_svr_base*。另参见 [installation directory \(安装目录\)](#)。

Messenger Express 一种邮件客户机，使用户可以通过基于浏览器 (HTTP) 的界面访问他们的邮箱。消息、文件夹和其他邮箱信息以 HTML 格式在浏览器窗口中显示。另参见 [webmail \(web 邮件\)](#)。

Messenger Express Multiplexor 起到“多路复用器”作用的代理讯息传递服务器。该服务器允许连接到 Messaging Server (Messenger Express) 的 HTTP 服务。Messenger Express Multiplexor 可帮助跨多个服务器机器分配邮件用户。

metadata (元数据) 关于组件的信息，如组件名称、组件行为规范等。

metadevice state database replica (元设备状态数据库副本) 存储在磁盘上、记录配置和所有元设备状态以及错误条件的数据库。该信息对 Solstice DiskSuite™ 软件磁盘集的正确运行很重要。

metainformation (元信息) 关于资源的信息，如作者名称、文档标题、创建日期等。“搜索引擎” robot 在创建资源说明时使用元信息和文档内容。

MHS (message handling system, 消息处理系统) 一组连接的 [MTA](#)、其用户代理及消息存储。

MIB (management information base, 管理信息库) 一种树形结构，定义 [SNMP master agent \(SNMP 主代理\)](#) 可以访问的变量。MIB 提供对 HTTP 服务器的网络配置、状态和统计信息的访问。使用 SNMP 可以从 [NMS](#) 查看此信息。另参见 [AUTH](#)。

migration (迁移) 将数据文件（如数据配置或自定义）从产品的一个版本迁移到另一个版本的过程。

MIME (multipurpose internet mail extensions, 多用途 internet 邮件扩展) 一种多媒体邮件和讯息传递的新标准。一种可以通过将多媒体文件附加到消息中从而将多媒体加入电子邮件消息的协议。

MIME data type (MIME 数据类型) MIME 类型控制系统支持的多媒体文件类型。

mime.types file (mime.types 文件) MIME 类型配置文件。此文件将文件扩展名映射到 MIME 类型，从而使服务器可以确定请求内容的类型。例如，带扩展名 .html 的资源请求表示客户机正在请求 HTML 文件，而带扩展名 .gif 的资源请求表示客户机正在请求 GIF 格式的图形文件。

mirror node (镜像节点) 含有与另一个活动节点相同的数据但驻留在其他数据冗余单元的活动 HADB 节点。每个活动节点必须有一个镜像节点，因此节点都是成对出现的。当一个节点检测到其镜像节点发生故障时，会接管该故障节点的角色，然后继续提供服务。另参见 [HADB node \(HADB 节点\)](#)、[active node \(活动节点\)](#)、[spare node \(备用节点\)](#) 和 [data redundancy unit \(数据冗余单元\)](#) (DRU)。

MMP (Messaging Multiplexor, 讯息传递多路复用器) 一种的专用 Messaging Server，它可与多个邮件服务器单点连接，从而帮助跨多个邮箱主机分配大的用户库。

mobile application configuration (移动应用程序配置) 一种 Access Manager 服务，该服务允许设置用于向移动设备传送的通讯录、日历和邮件应用程序。

mobile client type (移动客户机类型) 参见 [*client type \(客户机类型\)](#)。

mobile device (移动设备) 便携式无线设备，如移动电话或个人数字助理。

mobile devices link (移动设备链接) 在“Portal 桌面”上显示的超文本链接。

mobile devices page (移动设备页) 允许用户管理移动设备选项的 Web 页面。

Mobile Portal Desktop (移动 Portal 桌面) 在移动设备上显示的“Portal 桌面”。

moderator (仲裁人) 首先接收所有目标地址包含在邮递列表中的电子邮件，以决定是否应该将该消息转发到邮递列表中的地址的人。将消息转发给邮递列表之前，仲裁人可以对消息进行编辑。另参见 [mailing list \(邮递列表\)](#)、[expansion \(扩散\)](#)、[member \(成员\)](#)。

module (模块) (1) 在应用程序外部单独部署的 Web 应用程序、企业 bean、消息驱动 bean、应用程序客户机或连接器。另参见 [component \(组件\)](#)、[life-cycle module \(生命周期模块\)](#)。

(2) 一组 Java Enterprise System 服务器，它们彼此相互依赖，或者密切相关到足以部署为一个单元，以提供特定的一种或一组服务。服务模块是经过预先测试的多服务器集合体，适于在部署体系结构中使用。

modutil 为外部加密设备或硬件加速器设备安装 PKCS#11 模块所需的软件实用程序。

MTA (message transfer agent, 消息传输代理) 用于路由选择和传送消息的专用程序。各 MTA 之间协同工作以传输消息并将消息最终传送到目标收件人。MTA 确定是将消息传送到本地消息存储器，还是发送到另一个 MTA 进行远程传送。

MTA configuration file (MTA 配置文件) 包含 Messaging Server 的所有 [channel \(通道或频道\)](#) 定义以及确定如何为路由选择重写地址的 [rewrite rule \(重写规则\)](#) 的 imta.cnf 文件。

MTA directory cache (MTA 目录高速缓存) MTA 处理消息时需要的用户和组的目录服务信息快照。另参见 [directory synchronization \(目录同步\)](#)。

MTA hop (MTA 中继站) 在 *MTA* 之间路由消息的操作。

MUA 参见 [user agent \(用户代理\)](#)。

multihomed host (多原位址主机) 位于多于一个公共网络上的主机。

multihost disk (多主机磁盘) 在物理上连接到多个节点的磁盘。

multimaster replication (多主复制) 一种复制模型，可以在若干主复制副本中的任一副本中写入和更新条目，而且在执行写入或更新之前不必与其他主副本通信。每个服务器都保存副本的更改日志。在一台服务器上所做的修改会自动复制到其他服务器上。出现冲突时，使用时间戳确定哪台服务器拥有最新版本。

multiplexor (多路复用器) 包含与远程服务器通信的数据库链接的服务器。

multipurpose internet mail extensions (多用途 Internet 邮件扩展) 参见 [MIME](#)。

MX record (MX 记录) (mail exchange record, 邮件交换记录) 一种 [DNS](#) 记录，它将一个主机名映射到另一个主机名。

n + 1 directory problem (n + 1 目录问题) 在不同类型的目录和数据库中管理同一信息的多个实例时存在的问题，导致硬件和人力成本增加。

NAFO 参见 [network adapter failover group \(网络适配器故障转移组\)](#)。

name collision (名称冲突) 复制期间添加或重命名多个条目并尝试使用相同的 [DN](#) 时出现的冲突。目录服务器自动重命名冲突的条目以确保 DN 的唯一性。

name identifier (名称标识符) 用于在许多服务和身份认证提供者组织之间映射用户的帐户信息的假名，其目的是保持匿名。通过使用此标识符，身份认证提供者和服务提供者都不知道用户的真实身份。

name resolution (名称解析) 将 **IP address (IP 地址)** 映射到对应名称的过程。另参见 [DNS](#)。

namespace (名称空间) LDAP 目录的树结构。另参见 [DIT](#)。

naming attribute (命名属性) [DIT](#) 标识名中的最终属性。另参见 [relative distinguished name \(相对标识名\)](#)。

naming context (命名上下文) 某一 [DIT](#) 的特定后缀，由其 [DN](#) 确定。在 Directory Server 中，特定类型的目录信息存储在命名上下文中。例如，存储 Example Corporation Boston 办事处市场部员工所有条目的命名上下文可能称为 `ou=marketing, ou=Boston, o=example, c=US`。

native channel (本机频道) 显示本机内容的 Portal Server 频道。

native content (本机内容) 使用本机标记语言（例如 HTML）编写的内容，无需转换即可发送到客户机。

native desktop (本机桌面) 显示本机内容的 Portal Server Desktop。

NDN (nondelivery notification, 无法传送通知) 消息传输期间 MTA 在 [address \(地址\)](#) 与 [rewrite rule \(重写规则\)](#) 之间找不到匹配项时由 [MTA](#) 发回发件人的无法传送报告（附原始消息）。

nested role (嵌套角色) 命名其他角色定义的 [role \(角色\)](#)。某嵌套角色的成员集是其所包含角色的所有成员的并集。嵌套角色还可以定义扩展范围，以便将其他子树中角色的成员包括在内。

NetFile 一种基于 Java™ 技术的文件服务器应用程序，使用户可以远程访问文件系统，从而允许对文件和目录进行远程操作。

Netlet Java Enterprise System Portal Server 中使用的一种 Java [applet](#)，允许任何基于 [TCP/IP](#) 的应用程序通过已验证的 Portal Server 连接安全地连接到服务器。

NetMail NetMail 组件实现 NetMail（基于 Java 技术的客户机）和 NetMail Lite 电子邮件客户机。这些客户机与标准 IMAP 和 SMTP 服务器一起工作。

Netscape™ Console (Netscape™ 控制台) 使用 Java™ 编程语言编写的一个应用程序，它为服务器管理员提供一个图形界面，用于在企业网络内任一中心位置管理所有 Netscape™ 服务器。通过安装的任何 Netscape Console 实例，可以查看和访问企业网络上已授予您访问权限的所有 Netscape 服务器。

network adapter failover group (网络适配器故障转移组) 在同一子网同一节点配置的由一个或多个网络适配器组成的集合，出现适配器故障时它们互相进行备份。又称为 NAFO 组。

network address resource (网络地址资源) 参见 [network resource \(网络资源\)](#)。

network management station (网络管理站) 参见 [NMS](#)。

network manager (网络管理器) 用于读取、格式化和显示 [SNMP](#) 数据的程序。又称为 SNMP 客户机。

network resource (网络资源) 包含一个 (或多个) [logical host name \(逻辑主机名\)](#) 或 [shared address \(共享地址\)](#) 的资源。

Network Security Services for Java (面向 Java 的网络安全服务) (JSS) 提供到 Network Security Services SSL 库的 Java 绑定的类库。Portal Server 使用此类库启动来自 servlet 的 [SQL](#) 连接和接受 Portal Server Secure Remote Access Pack 网关中的 SSL 连接。

news channel (新闻频道) 用于张贴和共享信息的论坛。用户订阅新闻频道的目的是查看更新。新闻频道中的信息通常使用 URL 自动发布，或者由具有适当权限的用户发布。管理员可以控制新闻频道访问，方式是将用户分配到他们需要的频道并且决定谁可以查看新闻频道中的信息或向新闻频道发布信息。

news channel list (新闻频道列表) 显示您当前订阅的所有新闻频道的一个窗口。每个新闻频道都用单独的选项卡表示。

next-hop list (下一中继站列表) 相邻系统的列表，邮件路由使用它确定向何处传输消息。下一中继站列表中系统的顺序决定邮件路由向这些系统传输消息的顺序。

NIS (network information service, 网络信息服务) (仅限 UNIX) 一种关于程序和文件数据的系统, UNIX 系统使用它在某一计算机网络中收集、整理和共享有关计算机、用户、文件系统和网络参数的特定信息。

NMS (network management station, 网络管理站) 一个安装了一或多个网络管理应用程序的功能强大的工作站。NMS 是用于远程管理您的网络的计算机。

NNTP (Network News Transfer Protocol, 网络新闻传输协议) 一种新闻组协议。要在服务器上使用代理服务, 必须定义新闻服务器主机。

node (节点) (1) *DIT* 中的一个条目。

(2) 可以作为 SunPlex™ 系统一部分的物理计算机或域 (在 Sun™ Enterprise E10000 服务器中)。又称为 **host (主机)**。

(3) 计算节点。网络或 Internet 环境中的诸多计算机之一。分布式应用程序部署在此环境中, 其中不同的分布式组件、业务服务和服务器运行于各种计算节点上。

node agent (节点代理) 节点代理是一种轻量代理, 托管 Application Server 实例的每台计算机 (包括安装 Domain Administration Server 的计算机) 都需要此代理。节点代理按照 Domain Administration Server 的指示执行任务, 包括启动、停止、创建和删除 Application Server 实例。

non-cluster mode (非群集模式) 使用 -x 引导选项引导某一群集成员时产生的状态。在此状态下, 节点已不再是群集成员, 但仍然是群集节点。另参见 **cluster member (群集成员)**、**cluster node (群集节点)**。

nondelivery notification (无法传送通知) 参见 *NDN*。

NoPassword authentication (无密码验证) 一种验证类型, 允许用户登录 Access Manager 而不提示其输入密码。

NOTARY messages (NOTARY 消息) 符合 NOTARY 规范 RFC 1892 的无法传送通知 (Nondelivery notification, NDN) 和传送状态通知。

notification message (通知消息) 一种由 Messaging Server 发送的消息，它提供消息传送处理的状态及任何传送问题或完全失败的原因。这些消息只是为了提供信息，不需要邮寄主管执行操作。另参见 [delivery status notification \(传送状态通知\)](#)。

notification service (通知服务) 一种从其他服务器接收订阅和通知，然后将通知转发给特定订户的服务。Calendar Server csnotifyd 服务使用“事件通知服务”(Event Notification Service, ENS) 作为事件代理发送事件和待办任务的通知。

NSAPI 参见 [server plug-in API \(服务器插件 API\)](#)。

ns-slapd (仅限 UNIX) 负责 Directory Server 所有操作的一个进程或服务。Windows 系统中与之相当的是 [slapd.exe](#)。

ns-slapd.exe (仅限 Windows) Windows 系统中的进程监视器。

obj.conf file (obj.conf 文件) 服务器的目标配置文件。此文件包含附加的初始化信息、服务器自定义设置，以及服务器处理来自客户机（例如浏览器）的请求时使用的指令。Web Server 每次处理客户机请求时都读取此文件。

object class (对象类) 一种模板，用于指定条目所描述的对象种类和条目所包含的属性集。例如，Directory Server 指定一个 emailPerson 对象类，它具有诸如 commonname、mail（电子邮件地址）、mailHost 和 mailQuota 这些属性。

object identifier (对象标识符) 参见 [OID](#)。

object persistence (对象持久性) 参见 [persistence \(持久性\)](#)。

obsolete patch (过时修补程序) 不再视为有效或最新的修补程序。某修补程序被视为过时的三种情况：该修补程序的后续版本修复相同的问题、另一个修补程序包含原修补程序的修复补丁，或者该修补程序不再视为相关。

offline state (脱机状态) 一种状态，在此状态下，邮件客户机从服务器系统将消息下载到客户机系统，可以在客户机系统中查看和回复消息。这些消息可能从服务器删除，也可能不删除。

OID (**object identifier**, 对象标识符) 对象标识符的字符串表示, 由使用句点分隔的十进制数字组成 (例如, 1.3.6.1.4.1)。在 *LDAP* 中, 对象标识符用于唯一标识模式元素, 包括对象类和属性类型。对象标识符分层结构的顶层由标准主体管理, 并委托给希望构造自己的模式定义的组织。

online state (**联机状态**) 一种状态, 在此状态下, 消息保留在服务器上并由邮件客户机远程响应。

operational attribute (**操作属性**) 操作属性包含目录内部用于跟踪修改和子树 *properties* 的信息。操作属性不会在对搜索的响应中返回, 除非显式请求。

optional attribute list (**可选属性列表**) 某一指定对象类的可选属性列表。可选属性有前导关键字 *MAY*。

organization (**组织**) 在 *Directory Server Access Management Edition* 中, 代表某层次结构顶层的对象, 企业使用它来管理部门和资源。安装 *Directory Server Access Management Edition* 时, 它会自动创建一个顶层组织 (*default o=isp*) 来管理 *Directory Server Access Management Edition* 企业配置。安装之后, 可以创建其他组织来管理单独的企业。创建的所有组织都位列顶层组织之下。另参见 *subordinate reference* (*次级参考*)。

organization administrator (**组织管理员**) 拥有以下管理权限的用户: 可通过“用于讯息传递和协作的委托管理员” GUI 或 CLI, 在组织或子组织中创建、修改和删除邮件用户和邮件列表。

O/R mapping tool (**O/R 映射工具**) (*object-to-relational database tool*, 对象关系数据库工具) *Application Server* 的 *Administrative* (管理) 界面内的一个映射工具, 它为实体 bean 创建 XML 部署描述符。

OSI tree (**OSI 树**) (*Open Systems Interconnect tree*, “开放系统互连”树) 镜像“开放系统互连”网络语法的一个 *DIT*。OSI 树中的一个标识名 (*DN*) 示例是 *cn=billt,o=bridge,c=us*。

package (**软件包**) 文件与目录的集合。封装软件包是分发安装软件的一种方法。另参见 *A record* (*A 记录*)、*deployment* (*部署*)。

parallel resource type (并行资源类型) 一种资源类型，如并行数据库，它被装备为在群集环境中运行，以便可以同时从两个或多个节点控制该环境。

parallel service instance (并行服务实例) 在单独节点上运行的一个并行资源类型实例。

parameter (参数) (1) 自 Java Enterprise System Application Server 客户机发送的名称值对，包括表单字段数据、HTTP 头信息等，它封装在一个请求对象中。另参见 [attribute \(属性\)](#)。

(2) Java 方法或数据库制备命令的参数。

parent access (父访问权限) 如果授予，表示当 [bind DN \(绑定 DN\)](#) 是目标条目的父项时，用户具有对目录树中自身位置之下的条目的访问权限。

partition (分区) 参见 [message store partition \(消息存储分区\)](#)。

passivation (钝化) 从内存释放某一 bean 的资源而不损坏该 bean 的一种方法。这样，可令 bean 持久，无需实例化系统开销即可重新调用。

pass-through authentication (让渡验证) 参见 [PTA](#)。

pass-through subtree (让渡子树) 在让渡验证中，PTA Directory Server 将来自其 [DN](#) 包含在此子树中的所有客户机的绑定请求传递到验证 Directory Server。

password authentication (密码验证) 通过用户名和密码确认用户身份。另参见 [certificate-based authentication \(基于证书的验证\)](#)。

password file (密码文件) (仅限 UNIX) 存储 UNIX 用户登录名、密码和用户 ID 号的文件。此密码文件因其所在位置又称为 `/etc/passwd`。

password policy (密码策略) 控制如何在给定目录中使用密码的一组规则。

patch (修补程序) 对某一例程或目标程序的快速修改。另参见 [accumulated patch \(累积修补程序\)](#)、[obsolete patch \(过时修补程序\)](#)、[patch version number \(修补程序版本号\)](#)。

patch version number (修补程序版本号) 修补程序标识符的后两位数字，例如“nnnnnn-03”。每次发布修补程序新版本时，此编号都加一。

pattern (模式) 用于匹配目的的字符串表达式，如“允许”和“拒绝”过滤器中的字符串表达式。

PDC (personal digital certificate, 个人数字证书) 附加到消息的、用于验证用户身份的电子证书。可以通过正确输入用户 ID 和密码创建个人数字证书，也可以使用 **SSL certificate (SSL 证书)** 请求创建个人数字证书，该请求又将使用户通过其连接的服务器安全性证书。

peer (同级类别) 具有与另一类别相同父类别的子类别。

permanent failure (永久性故障) 消息处理过程中出现的错误情况。出现永久性故障时，消息存储器会删除其电子邮件消息副本。**MTA** 将消息发回发件人并删除其消息副本。

permissions (权限) (1) 授予或拒绝给某一用户或组的一组权限。此信息包括用户名或组名、一个或多个有效电子邮件地址，以及如何和向何处传送电子邮件。

(2) 在访问控制上下文中，权限声明是授予还是拒绝对目录信息的访问权限，以及授予或拒绝的访问权限级别。另参见 **access rights (访问权限)**。

(3) 控制对日历的访问权限的设置。例如，在 **Calendar Express** 中，权限包括 **Availability (可用性)**、**Invite (邀请)**、**Read (读取)**、**Delete (删除)** 和 **Modify (修改)**。**Calendar Server** 管理员使用命令行实用程序将权限设置为 **ACE** 字符串。另参见 **ACL**。

persistence (持久性) (1) 对于企业 bean 而言，指用于在实体 bean 的实例变量与某一基础数据库之间传输实体 bean 的状态的协议。另参见 **transience (瞬态)**。

(2) 对于会话而言，指会话存储机制。

persistence manager (持久性管理器) 负责 **entity bean (实体 bean)** 或容器中安装的 bean 的 **persistence (持久性)** 的实体。

persistent state (持久状态) 其中，对象状态保存在持久存储器中（通常为数据库）。

personal digital certificate（个人数字证书） 参见 [PDC](#)。

personal folder（个人文件夹） 只能由所有者读取的文件夹。另参见 [shared folder](#)（共享文件夹）。

pk12util 从您的内部计算机导出证书和密钥数据库并将它们导入外部 PKCS#11 模块所需的软件实用程序。

PKI（public key infrastructure，公共密钥基础结构）使用户的标识能够链接到浏览器或移动设备。“无线 PKI”指出现在手提设备上的 [certificate-based authentication](#)（基于证书的验证）。

plaintext（纯文本） 传送数据的一种方法。其定义依赖于上下文。对于 [SQL](#)，纯文本密码会被加密，因此不会作为明文发送。对于 [SASL](#)，纯文本密码将被散列，并且仅将密码的一个散列作为文本发送。

plaintext authentication（纯文本验证） 参见 [password authentication](#)（密码验证）。

pluggable authentication（可插接式验证） 允许 J2EE™ 应用程序从 J2SE™ 平台使用 Java™ 验证和授权服务 (Java™ Authentication and Authorization Service, JAAS) 软件的一种机制。开发者可以插入他们自己的验证机制。

plug-in（插件） (1) 一种对浏览器的代码扩展，可显示或执行 Web 页面中的内容。插件使浏览器能够显示浏览器在未使用插件时不能显示的页面内容元素。

(2) 一种可以加载，然后作为整体系统的一部分使用的辅助程序。例如，Calendar Server 可以使用插件访问非 LDAP 目录服务。

pointer CoS（指针 CoS） 一种只使用模板 DN 标识模板条目的指针类服务。

point-to-point delivery model（点对点传送模型） 一种模型，其中生成者将消息定址到特定队列，使用者从为保存他们的消息而建立的队列提取消息。一条消息只传送给一个消息使用者。

policy（策略） (1) 描述授权何人在特定条件下访问特定资源的规则。可以基于组织中的用户组或角色来建立规则。

(2) 在 Directory Server Access Management Edition 中，定义规则以帮助保护组织的 Web 资源。策略只分配给组织和角色。

poll (轮询) Instant Messaging Server 中的一种功能，使您能够询问用户对某一问题的回答。您可以向选定用户发送问题和可能的答案，他们回馈所选答案。

pooling (入池) 提供许多预配置资源以提高性能的过程。如果某一资源入池，组件可使用池中的现有实例，而不必实例化一个新实例。在 Java Enterprise System Application Server 中，数据库连接、servlet 实例和企业 bean 实例都可以入池。

POP3 (Post Office Protocol Version 3, 邮局协议版本 3) 一种可提供标准传送方法而且不需要 MTA 对用户的邮件文件夹具有访问权限的协议。在邮件客户机和消息传输代理通常位于不同计算机上的联网环境中，不需要具有访问权限是一个有利条件。

port (端口) 进行 TCP/IP 连接的位置 (套接字)。Web 服务器传统上使用端口 80，FTP 使用端口 21，telnet 使用端口 23。Java Enterprise System Portal Server 使用特殊端口 (尤其是在客户机系统上) 通过 Portal Server 会话与服务器安全地进行通信。

portal (门户) 企业想要使其对门户用户可用的一组资源的一个入口点。对于某些使用者门户，资源组包括整个 World Wide Web，但是对于大多数企业，资源组包括特定于用户和企业之间关系的信息、应用程序和其他资源。Portal Server Desktop 是用于在 Portal Server 中生成门户的应用程序。

Portal Desktop (Portal 桌面) 由 Portal Server 生成的任一桌面。

portal node (门户节点) 运行 Portal Server 软件或 Portal Server Pack 软件的物理计算机。又称为 host (主机)。

port number (端口号) 用于指定主机上个别 TCP/IP 应用程序的一个编号。它为传送的数据提供目的地。

postdeployment (后期部署) Java Enterprise System 解决方案生命周期过程中的一个阶段，在此阶段中，将启动分布式应用程序，并对其进行监视、调优性能，以及进行动态升级以纳入新功能。

postmaster account (邮寄主管帐户) 从 Messaging Server 接收系统生成的消息的电子邮件组和电子邮件地址的别名。邮寄主管帐户必须指向一个或多个有效邮箱。

Post Office Protocol Version 3 (邮局协议版本 3) 参见 *POP3*。

potential master (潜在主站) 参见 *potential primary* (潜在主节点)。

potential primary (潜在主节点) *primary* (主节点) 节点发生故障时能够控制 *failover resource* (故障转移资源) 类型的群集成员。另参见 *default master* (默认主站)。

predeployment (前期部署) Java Enterprise System 解决方案生命周期过程中的一个阶段, 在此阶段中, 业务需要被转换为 *deployment scenario* (部署方案): 一个 *logical architecture* (逻辑体系结构) 和解决方案必须满足的一组服务质量要求。

prepared command (制备命令) *spider* 中的一个数据库命令, 它已经过预编译, 使得重复执行更加有效。制备命令可以包含参数。另参见 *prepared statement* (制备语句)。

prepared statement (制备语句) 封装重复用于取得数据的 *QUERY*、*UPDATE* 或 *INSERT* 语句的一个类。制备语句至少包含一条 *prepared command* (制备命令)。

presence index (存在索引) 一种过滤方法, 它允许高效搜索包含特定类型的某一属性的条目, 而不必考虑条目中该属性的值如何。

presentation layout (表示布局) Web 页面内容的格式。

presentation logic (表示逻辑) 在应用程序中创建页面的活动, 包括处理请求、生成响应内容以及格式化客户机页面。通常由 Web 应用程序处理。

preset message (预设消息) 可以编写并保存为 Portal Server Mobile Access 移动首选项, 以备将来在移动电子邮件应用程序中使用的短消息。

primary (主节点) 资源组或设备组当前联机的一个 *node* (节点)。主节点是当前托管或实现与资源相关联的服务的一个节点。另参见 *secondary* (辅助节点)。

primary document directory (主文档目录) 参见 *document root* (文档根目录)。

primary host name (主要主机名) 主要公共网络上的节点的名称。主要主机名总是 `/etc/nodename` 中指定的节点名称。另参见 **secondary host name (辅助主机名)**。

primary key (主键) 客户机借以查找特定实体 bean 的唯一标识符。

primary key class name (主键类名) 一个变量，它指定了 bean 的主键的全限定类名。用于 Java Naming and Directory Interface™ (JNDI) 查找。

principal (主体) 主体是指可以获得联合身份、有决策能力且自我执行验证操作的实体。例如，主体可以是个人用户、一组个人、公司、其他法律实体，或者是 Liberty 体系结构的组成部分。

private host name (专用主机名) 通过 **cluster interconnect (群集互连)** 与 **node (节点)** 通信时所使用的 **host name (主机名)** 别名。

private key (专用密钥) 参见 **public-key cryptography (公共密钥密码学)**。

privilege (权限) 授予单个用户、一组用户或单个资源的一类访问权。

process (进程) (1) 操作系统建立的一种自包含、功能完全的执行环境。应用程序的每个实例通常在一个单独的进程中运行。另参见 **thread (线程)**。

(2) 活动程序的执行序列。一个进程由一个或多个线程组成。

produce (生产) 将消息传递给客户机运行时环境以送达目标。

producer (生产者) 由会话创建的对象 (MessageProducer)，用于将消息发送到目标。在点对点传送模型中，生产者是发送者 (QueueSender)。在发布/订阅传送模型中，生产者是发布者 (TopicPublisher)。

production environment (生产环境) 应用程序生命周期过程中的一个阶段，在此阶段，将启动分布式应用程序，对其进行监视、性能调优，并进行动态升级以加入新功能。

programmatic security (程序安全性) 在代码中显式控制安全性的处理方法，此方法不允许诸如组件的容器、bean 的容器或 servlet 引擎等进行安全处理。与 **declarative security (声明性安全)** 相对。

programmer-demarcated transaction (程序员界定事务) 参见 **bean-managed transaction (bean 管理事务)**。

propagation behavior (传播行为) 使用者与供应者间的同步过程。

property (属性) 定义应用程序组件行为的单个属性。在 `server.xml` 文件中，属性是含有名称值对的元素。

protocol (协议) (1) 描述网络设备如何交换信息的一组规则。

(2) 对要交换的消息以及两个或多个系统交换信息所要遵循的规则作出的形式描述。

provider (提供者) 频道的程序学形态。向提供者添加配置数据即会使其摇身变为频道实例。提供者是一个 Java™ 类，负责将文件内容或是应用程序或服务的输出转换成适当的频道格式。Portal Server 随带有多个提供者，其中包括书签提供者、应用程序提供者和备忘录提供者。桌面成像时，会轮流查询每个提供者所关联频道的内容。一些提供者能够基于各自的配置生成多个频道。

内容提供者的例子有 `UserInfoProvider` 和 `BookmarkProvider`。**container (容器)** 提供者的示例有 `TabContainerProvider` 和 `SingleContainerProvider`。活页提供者的例子有 `JSPProvider`、`XMLProvider`、`URLScrapperProvider` 和 `SimpleWebServicesProvider`。

provisioning (置备) 在 Java Enterprise System Directory Server 中添加、修改或删除条目的过程。这些条目包括用户、组及域信息。

proxy (代理) (1) 一个系统借以代表另一系统对协议请求作出响应的机制。在网络管理中使用代理系统，便无需在简单设备（如调制解调器）中实现全部的协议栈。

(2) 代表客户机发出或服务请求的中介程序。代理轮流充当服务器和客户机，并用来控制各种网络服务的内容。另参见 **reverse proxy (反向代理)**。

proxy authorization (代理授权) 一种特殊形式的验证，其中，客户机以自身身份绑定到目录，但基于每一操作授予其另一用户的访问权。此另一用户称为代理用户，其 *DN* 为代理 DN。

proxy DN (代理 DN) 一个条目的 *DN*，该条目拥有对客户机应用程序正要在其上执行操作的目标的访问权限。与 **proxy authorization (代理授权)** 一起使用。

Proxylet 一种动态代理服务器，在客户机机器上运行以将 URL 重定向到 SRA 网关。另参见 *Sun Java™ System Portal Secure Remote Access (SRA)*。

PTA (pass-through authentication, 让渡验证) 一个 Java Enterprise System Directory Server 借以向另一 Directory Server 查询绑定规则的机制。

PTA Directory Server 在 *PTA* 中，PTA Directory Server 将其收到的绑定请求发送 (让渡) 给验证 Directory Server。

PTA LDAP URL 在 *PTA* 中，定义验证 Directory Server、让渡子树或子树以及可选参数的 URL。

public folder (公共文件夹) 具有多个所有者的文件夹，由多个可对其访问的人员共享。根据为该文件夹设置的 *ACL*，该文件夹可由多个人更新或管理。

public information directories (公共信息目录) (仅限 UNIX) 文档根目录以外位于 UNIX 用户主目录或处于用户控制下的目录，或者是处于用户控制下的目录。

public key (公共密钥) 公共密钥加密中使用的加密密钥。

public-key certificate (公共密钥证书) 一种数据结构，其中含有用户的公共密钥以及有关证书有效时间和日期的信息。

public-key cryptography (公共密钥密码学) 一种加密方法。在公共密钥密码系统中，每个人都有两个互补密钥：一个公示的密钥和一个秘密的密钥 (又称为专用密钥)。每个密钥可对另一密钥生成的代码进行解锁。知道公共密钥对于推断相应的秘密密钥并无帮助。公共密钥可以公布并在通信网络上广泛散播。此协议无需传统密码系统所要求的安全通道即可提供保密功能。又称为非对称密钥密码学。

public-key encryption (公共密钥加密) 一种加密方法，使用由公共和私有组件组成的两部分密钥（代码）。加密邮件使用的是收件人已经公布的公共密钥。收件人使用只有自己知道的未公布专用密钥来解密邮件。

public key infrastructure (公共密钥基础结构) 参见 *PKI*。

Public Network Management (公共网络管理) 通过故障监视和故障转移防止因单个网络适配器或电缆故障而丧失节点可用性的软件。“公共网络管理”故障转移使用称为 **network adapter failover group (网络适配器故障转移组)** 的多组网络适配器，在群集节点与公共网络之间提供冗余连接。故障监视和故障转移功能相互配合，共同确保资源的可用性。

publish and subscribe delivery model (发布和订阅传送模型) 在此模型中，发布者和订户通常为匿名，并且可以动态发布或订阅某一主题。该系统将从某个主题的多个发布者那里收到的消息分发给它的多个订户。

purge a message (清除消息) 永久除去已删除并且在用户和组文件夹中不再引用的消息。随后将空间返还给消息存储文件系统。另参见 **delete a message (删除消息)** 和 **expunge a message (抹去消息)**。

QOS (quality of service, 服务质量) 为服务器实例或虚拟服务器设置的性能限制。例如，如果您是一个 ISP，您可能希望根据所提供的带宽来收取不同的虚拟服务器费用。您可以限制带宽总量和连接数量。

queue (队列) 管理员为实现点对点传送模型而创建的对象。队列始终可用于存放消息，即使在使用其消息的客户机处于非活动状态时亦是如此。队列被用来作为生产者与消费者之间的一个中间存放地。

quorum device (法定设备) 为两个或更多节点所共享的磁盘，该磁盘的投票数用来确立群集运行所需的法定数目。群集只有在达到法定票数时才能运行。当群集分割成多个单独的节点组时，使用法定设备来确定哪组节点构成新的群集。

RAF (robot application function, robot 应用函数) 一种可在 robot 过滤器配置文件中使用的函数。用户定义的 robot 应用函数又称为插件函数。这些函数通过指令调用。

RAM (random access memory, 随机存取存储器) 计算机中用半导体制成的物理存储器。

RAR file (RAR 文件) (resource archive file, 资源归档文件) 含有资源适配器的 Java™ 归档 (Java™ archive, JAR) 文件。

RC2 由 RSA Data Security 开发的可变密钥大小块加密器。

rc.2.d file (rc.2.d 文件) (仅限 UNIX) UNIX 机器上的一种文件, 描述机器启动时运行的程序。此文件因其所在位置又称为 /etc/rc.2.d。

RC4 由 RSA Data Security 开发的流加密器。比 RC2 速度快。

RD 参见 [resource description \(资源描述\)](#)。

RDB 关系数据库。

RDBMS 关系数据库管理系统。

RDM 参见 [resource description message \(资源描述消息\)](#)。

RDN (relative distinguished name, 相对标识名) 在将实际条目的祖先附加到字符串末尾形成完整 *DN* 前, 该条目本身的名称。多数 RDN 由来自条目的单个属性类型和值组成。

read-only bean (只读 bean) 从不会被 EJB™ 客户机修改的实体 bean。另参见 [entity bean \(实体 bean\)](#)。

realm (疆域) 安全服务的安全管理员定义和实施公共安全策略的范围。在 J2EE™ 规范中又称为安全策略域或安全域。

redirection (重定向) 一种机制, 用以将访问特定 URL 的客户机发送到相同或不同服务器上的另一不同位置。重定向在资源移动后想要客户机透明地使用新位置时很有用。重定向还用于在不带尾随斜杠访问目录时维护相对链接的完整性。

reference deployment architecture (**参考部署体系结构**) 经过设计、实现和性能测试的 **deployment architecture** (**部署体系结构**)。参考部署体系结构可作为设计自定义解决方案的部署体系结构的起点。

referential integrity (**引用完整性**) 一种机制，用于确保在目录内保持 *DN* 取值属性所表示的条目之间的关系。

referral (**引用**) 当服务器从客户机那里收到无法处理的搜索或更新请求时，服务器会向客户机回送一个指针，该指针指向可以处理该请求的 Java Enterprise System Directory Server。

referral hop limit (**引用中继站限制**) 客户机在一行中应遵从的最大引用数。

regular expression (**正则表达式**) 一种文本字符串，它使用特殊字符来表示字符的范围或分类，目的是要进行模式匹配。

relative distinguished name (**相对标识名**) 参见 *RDM*。

relaying (**中继**) 将消息从一个消息传递服务器传递给另一消息传递服务器的过程。

remote interface (**远程接口**) Enterprise JavaBeans™ 组件的两个接口之一。远程接口定义可由客户机调用的业务方法。

rendering (**转译**) 将采用“抽象标记语言”(Abstract Markup Language, AML) 编写的内容转换为适合特定移动设备的设备特定标记语言的过程。

rendering channel (**转译频道**) 用于显示转译内容的 Portal Server Mobile Access 频道。

rendering engine (**转译引擎**) 在 Portal Server 中，用于将 AML 转换为适合于给定移动客户机的语言。

rendering filter (**转译过滤器**) 用于在转译引擎与客户机间传递所要转换内容的过滤器。

replica (**副本**) **replication** (**复制**) 区域在服务器上的实例。另参见 **consumer replica** (**使用者副本**) 和 **supplier replica** (**供应者副本**)。

replica cycle (**副本周期**) 参见 **replication cycle** (**复制周期**)。

replica directory server (副本目录服务器) 接收全部或部分数据副本的目录。

replica group (副本组) 持有特定复制区域实例的服务器。一个服务器可以是多个副本组的一部分。

replication (复制) 同步分布在多个 Directory Server 上的数据并纠正更新冲突的过程。

replication agreement (复制协议) 一组存储在供应者服务器上的配置参数，这些参数指明了要复制的后缀、数据推送到的使用者服务器、**replication (复制)** 可以发生的次数、供应者用来绑定到使用者的 **DN** 和凭证，以及连接所采用的安全方式。

replication base entry (复制基条目) 所复制区域之根的 **DN**。

replication cycle (复制周期) 在两个或更多副本间交换更新信息的时间间隔。复制周期在尝试向另一副本或另一组副本推送数据或从中拉取数据时开始，在成功交换数据或遇到错误时结束。

replication session (复制会话) 作为 **replication cycle (复制周期)** 的一部分，在 **replica group (副本组)** 中的两个服务器间为传递更新信息而建立的会话。

request object (请求对象) 含有页面和会话数据的对象，这些数据由客户机产生，作为输入参数传递给 servlet 或用 **JSP™ technology (JSP™ 技术)** 创建的页面。

required attribute list (必需属性列表) 指定对象类所必需的属性列表。必需属性有前导关键字 **MUST**。

required attributes (必需属性) 使用某特定对象类的条目中必须要有的属性。另参见 **allowed attributes (允许的属性)**、**attribute (属性)**。

resource (资源) (1) 资源类型的实例。可以存在许多相同类型的资源，其中每个资源都有各自的名称和一组属性值，这样便可以在群集上运行底层应用程序的许多实例。

(2) 网络中可由 **URL** 标识的任何项，如 **web** 页、文档或 **FTP** 目录。在非正式场合，常常将资源称为文档。

(3) 服务器可以访问并发送给请求客户机的任何 **URL**、目录或程序。

resource calendar (资源日历) 与资源（如会议室）或设备（如笔记本电脑或高射投影仪）关联的日历。

resource description (资源描述) 通过 [URL](#) 与资源关联的属性值对列表。资源描述可由代理自动生成，也可人工编写。一旦组装好资源描述系统信息库，服务器便可通过资源描述消息以程序化方式将该系统信息库导出，以供 web 代理对资源描述进行搜索和检索。资源描述以 [SOIF](#) 格式存储。

resource description message (资源描述消息) 一种机制，用以搜索和检索可通过网络访问的资源的元数据（称为资源描述）。

resource group (资源组) 作为一个单元由 [RGM](#) 统一进行管理的资源集合。由 RGM 管理的每个资源均须在资源组中进行配置。通常将相关或相互依赖的资源分成一组。

resource group manager (资源组管理器) 参见 [RGM](#)。

resource group state (资源组状态) 资源组在任意给定节点上的状态。

resource invocation (资源调用) 在节点上运行的资源类型实例。这是一个抽象概念，表示已在节点上启动的资源。

Resource Management API (资源管理 API) SunPlex™ 系统内用于实现群集环境中应用程序高可用性的应用编程接口。又称为 [RMAPI](#)。

resource manager (资源管理器) 在某资源（如数据库或消息代理）与该资源的一个或多个客户机（如 Java Enterprise System Application Server 进程）之间提供便利的对象。控制着全局可用的数据源。

resource monitor (资源监视器) 可选的资源类型实现部件，它会周期性地对资源进行故障探测，以确定资源运行是否正常及其具体执行情况。

resource offering (资源出具) 在“搜索服务”中，资源出具定义了一段身份认证数据与提供其访问权的服务实例之间的关联。

resource reference (资源引用) 部署描述符中的一种元素，用于标识组件的资源编码名称。

resource state (资源状态) RGM 资源在给定节点上的状态。

resource status (资源状况) 故障监视器所报告的资源情况。

resource type (资源类型) 对 [data service \(数据服务\)](#)、LogicalHostname 或 SharedAddress 群集对象的唯一命名。数据服务资源类型可以是故障转移类型，也可以是可伸缩类型。另参见 [failover resource \(故障转移资源\)](#)、[scalable resource \(可伸缩资源\)](#)。

resource type property (资源类型属性) 一组键值对，用于描述和管理给定类型的资源，并作为资源类型的一部分由 RGM 进行存储。

response buffer (响应缓冲区) Portal Server Mobile Access 服务器响应缓冲区将较大的响应存储为多个单独的较小响应，以使它们适合有限的设备缓冲区。

response object (响应对象) 引用调用客户机并为该客户机提供输出生成方法的对象。

restart (重新启动) 在不删除其状态信息的情况下启动 *robot*，这样做可使 robot 以其先前停止时的同一状态开始运行。与 [fresh start \(全新启动\)](#) 相对。

restore (恢复) 将文件夹内容从备份设备复制到消息存储。另参见 [back up \(备份\)](#)。

ResultSet object (ResultSet 对象) 实现 java.sql.ResultSet 接口的对象。ResultSet 对象用于封装一组从数据库或其他表格数据源检索到的行。

retro changelog (推后更改日志) 按更改到达本地服务器的顺序而不是其应用于系统的顺序存储这些更改。推后更改日志并非设计用于多主复制环境。与 [change log \(更改日志\)](#) 不同，推后更改日志并不在复制中使用。它提供与 Directory Server 4 的向后兼容性。

reusable component (可重用组件) 为了能在一个以上的功能体中使用（例如，被一个以上的资源或应用程序使用）而创建的组件。

reverse DNS lookup (反向 DNS 查找) 通过查询 *DNS* 将数值 *IP address* (*IP 地址*) 解析成相当的 *fully qualified domain name* (*全限定域名*) 的过程。

reverse proxy (反向代理) 在客户机与服务器之间执行双向 *URL* 重写和转换的代理。与存在于客户机端的代理不同，反向代理存在于网络的服务器端。在 *Java Enterprise System Portal Server* 中，反向代理存在于 *Java Enterprise System Portal Server Secure Remote Access Pack* 中。

Rewriter (重写器) 重写器提供了一个 *Java™* 类库，用于以各种 *web* 语言（如 *HTML*、*Javascript* 和 *XML*）和在 *HTTP* 位置头中（重定向）重写 *URL* 引用。重写器定义了 *Java Enterprise System Directory Server Access Management Edition* 服务来存储规则，这些规则定义要如何进行重写以及要重写的数据。重写器还包括一个用于编辑这些规则的管理控制台模块。

rewrite rule (重写规则) *MTA* 用来将消息路由到正确主机以进行传送的工具。重写规则执行以下功能：(1) 从收到的消息的地址中提取主机和域说明；(2) 以重写规则模式对主机和域说明进行匹配；(3) 根据域模板重写主机和域说明；(4) 决定应将消息置于哪个频道队列。又称为域重写规则。

RFC (*request for comments*, 请求注解) 由 *Internet* 工程任务组维护的文档系列，描述 *Internet* 协议套件以及相关实验。几乎没有 *RFC* 描述 *Internet* 标准，但所有 *Internet* 标准都作为 *RFC* 发布。参见 <http://www.imc.org/rfc.html>。

RGM (*resource group manager*, 资源组管理器) 一个软件工具，用于通过所选群集节点上自动启动和停止群集资源，使这些资源具有高度可用性和可伸缩性。如果硬件或软件发生故障或重新引导，*RGM* 会根据预先配置的策略进行操作。

RMI (*remote method invocation*, 远程方法调用) *Java API* 标准集，开发者能够用其来编写可将对象传递给远程进程的接口。

RMIC 远程方法调用编译器。

robot 一个程序，用于查找位于网络特定部分的所有资源。

robot application function (robot 应用函数) 参见 *RAF*。

role (角色) (1) 应用程序中主题对象的功能分组，在部署环境中用一个或多个组来表示。另参见 **user (用户)**、**group (组)**。

(2) 在 Java Enterprise System Directory Server Access Management Edition 中，指的是表示特许操作选择集的分组。通过将角色应用于用户或服务，主体可以执行这些操作。例如，通过将某些权限限定于“员工”角色或“经理”角色并将该角色应用于某用户，该用户的访问能力便被限定于该角色所授予的权限范围内。角色是使用访问控制指令 (**ACI**) 定义的。

(3) 一种条目分组机制。每个角色均有成员，这些成员是拥有该角色的条目。

role-based attributes (基于角色的属性) 因条目在关联的 **CoS** 模板内拥有某个特定角色而出现在该条目中的属性。

rollback (回滚) 撤销 **transaction (事务)**。

root (超级用户) (仅限 UNIX) UNIX 机器上享有最高特权的用户。超级用户拥有对机器上所有文件的完全访问权限。

root DN (根 DN) **Directory Manager (目录管理员)** 的 **DN**。

Root DSE (根 DSE) 一个条目，由 Directory Server 自动生成，并以一个空（长为零字节）的 **DN** 从 baseObject 搜索中返回。“根 DSE”向客户机提供有关服务器配置的信息，如指向 **subschema entry (子模式条目)** 的指针、服务器所持命名上下文的 **DN** 列表以及服务器支持的 **LDAPv3** 控件和扩展名的列表。另参见 **DSE**。

root entry (根条目) **DIT** 分层结构的顶级条目。

root suffix (根后缀) 一个或多个 **sub suffix (子后缀)** 的父项。目录树可包含一个以上的根后缀。

router (路由器) 负责确定网络通信流动路径的系统。路由器使用路由选择协议获得有关网络和算法的信息，以根据称为“路由选择矩阵”的多个判别条件选择最佳路由。在“开放系统互连”术语中，路由器为“网络层”中间系统。另参见 **fully qualified domain name (全限定域名)**。

routing (路由选择) 参见 **message routing (消息路由选择)**。

routing tables (路由表) 存放有关消息始发者和接受者信息的内部数据库。

row (行) 单个数据记录，其中含有表中每一列的值。

RowSet object (RowSet 对象) 一个对象，其中封装了一组从数据库或其他表格数据源检索到的行。RowSet 对象对 `java.sql.ResultSet` 接口进行了扩展，从而使 **ResultSet object (ResultSet 对象)** 能够充当基于 JavaBeans™ 组件体系结构的组件。

RPC (remote procedure call, 远程过程调用) 一种用于访问远程对象或服务的机制。

RTT (\round trip time, 往返时间) 在一个闭合回路 (从服务器到客户机，再从客户机返回服务器) 中传输一个信号所需的时间。这一延迟在要求双向交互通信的系统中很重要。在此系统中，RTT 会直接影响吞吐率。在 Java Enterprise System Directory Server 上下文中，RTT 和 TCP 窗口会通过广域网对复制性能产生显著影响。又称为往返延迟时间。

rules (规则) 用于确定某个条件是否得到满足的逻辑测试。**robot** 将规则用作过滤器的组成部分，以确定要建立索引的内容类型，及在分类规则中确定分配给资源的类别。

runtime system (运行时系统) 程序运行的软件环境。运行时系统包括加载用 Java™ 编程语言编写的程序、动态链接本机方法、管理内存以及处理异常所需的全部代码。它包括一个 Java 虚拟机实现，该虚拟机实现可以是一个 Java 解释程序。

SAF (server application function, 服务器应用函数) 参与处理请求和其他服务器活动的函数。

safe file system (安全文件系统) 一种文件系统，可执行日志记录功能。这样，当某个系统崩溃时，该系统可以将数据回滚到崩溃前的状态，并恢复所有数据。例如，Veritas 文件系统 (Veritas File System, VxFS) 就是一个安全文件系统。

SASL (simple authentication and security layer, 简单验证和安全层) 一种控制机制的方法，POP、IMAP 或 **SMB protocol (SMB 协议)** 客户机通过该机制向服务器表明身份。Java Enterprise System Messaging Server 对使用 SMTP SASL 的支持符合 RFC 2554 (ESMTP AUTH) 要求。SASL 在 RFC 2222 中定义。另参见 [POP3](#)、[IMAP4](#)。

scalable coherent interface (可伸缩相关接口) 用作 **cluster interconnect (群集互连)** 的高速互连硬件。

scalable resource (可伸缩资源) 一种资源，该资源运行在使用 **cluster interconnect (群集互连)** 的多个节点上（每个节点上一个实例），从而给服务的多个远程客户机一种单一服务的印象。

scalable service (可伸缩服务) 同时运行在多个节点上的已实现数据服务。

schema (模式) 说明何种信息类型可以在目录中存储为条目的定义。当与模式不匹配的信息存储在目录中时，试图访问该目录的客户机可能无法显示正确结果。

schema checking (模式检查) 一种可确保在目录中添加或修改的条目与所定义的模式相符的检验过程。默认情况下，模式检查处于启用状态。当用户试图保存与模式不符的条目时，会收到一个错误提示。

schema name (模式名称) *SOIF* 的模式或类型。例如，一个文档的 *SOIF* 具有模式名称 *@DOCUMENT*，而一个 **resource description message (资源描述消息)** 头的 *SOIF* 具有模式名称 *@RDMHeader*。

SCM 参见 **service control manager (服务控制管理器)**。

scoping (作用范围) 在导入代理导入的资源说明中设置的限制条件。所用的语法与用户搜索的语法相同。

search base (搜索基) 参见 **base DN (基 DN)**。

Search database (搜索数据库) 通常由 *robot* 生成的资源说明的可搜索数据库。另参见 *robot*。

Search Engine (搜索引擎) 集成在 *Portal Server 6.0* 中的搜索功能。以前称为 *Compass Server (Portal Server 3.0)*。*Search Server* 持有由 *robot* 收集的资源说明组成的数据库，这些资源说明通常都进行了分类。用户可以搜索资源说明或通过浏览类别查找特定资源。

secondary (辅助节点) 可以在主节点发生故障时控制磁盘设备组和资源组的群集成员。另参见 **primary (主节点)**。

secondary host name (辅助主机名) 用于访问辅助公共网络上的节点的名称。另参见 **primary host name (主要主机名)**。

secure socket layer (安全套接字层) 参见 *SSL*。

security (安全性) 一种确保应用程序资源仅由授权客户机访问的筛选机制。

security-module database (安全模块数据库) 含有描述 *SQL* 加密器硬件加速器信息的文件。又称为 *secmod*。

self access (自访问权限) 如果将该权限授予用户，则表示当 **bind DN (绑定 DN)** 与目标条目匹配时，用户对他们自己的条目具有访问权限。

self-generated certificate (自生成证书) 仅当用条目的公共值的消息摘要为条目进行命名并且这些名称进行安全通信时，才能使用的公共密钥值。另参见 **issued certificate (已颁发证书)**。

sendmail (仅限 UNIX) 公共 MTA。在大多数应用程序中，Java Enterprise System Messaging Server 可用作 *sendmail* 的混杂信息替换。

serializable object (可序列化对象) 可以解构和重构、从而使之可在多个服务器之间存储和分发的对象。

server (服务器) 一种多线程软件进程（有别于硬件服务器），可为通过外部接口访问服务的 **client (客户机)** 提供一组分布式或紧密结合的服务。

server administrator (服务器管理员) 执行服务器管理任务的人员。服务器管理员根据任务 *ACI* 提供对特定服务器任务的受限访问。配置管理员必须为用户分配服务器访问权限。当某个用户拥有服务器访问权限时，该用户就是一个服务器管理员，可以向用户提供服务器访问权限。

server assembly (服务器集合体) 一组相互依赖或密切相关到足以作为一个整体安装和部署的 Java Enterprise System 服务器。

server authentication (服务器验证) 一种可使客户机确保连接到安全服务器的验证方法，可防止其他计算机仿冒服务器或避免将并不安全的服务器当作安全服务器。

server daemon (服务器守护进程) 在运行时侦听并接受来自客户机的请求的进程。

server instance (服务器实例) 在同一计算机上安装的同一服务器可以含有多个实例。每个实例都有其自己的目录结构、配置和部署的应用程序。每个实例还可以含有多个虚拟服务器。另参见 [virtual server \(虚拟服务器\)](#)。

Server Message Block protocol (Server Message Block 协议) 一种协议，可为计算机中的客户机应用程序提供对文件的读取和写入权限，以及从计算机网络中的服务器程序请求服务的方法。SMB 协议可通过 Internet 在其 *TCP/IP* 协议或其他网络协议（例如 Internetwork Packet Exchange 和 NetBEUI）之上使用。Java Enterprise System Portal Server 对 NetFile 使用 SMB。

server plug-in API (服务器插件 API) 一个扩展，允许扩展和自定义 Java Enterprise System 服务器的核心功能，并提供一个可伸缩的高效机制，用于构建 HTTP 服务器与后端应用程序之间的接口。又称为 NSAPI。

server process (服务器进程) 一种由操作系统设置的自包含、功能完全的执行环境。应用程序的每个实例通常在一个单独的进程中运行。

server root (服务器根目录) (1) 服务器计算机上的一个目录，专门用于存放服务器程序及配置文件、维护文件和信息文件。又称为 ServerRoot。

(2) 相对于服务器上其他文件的一个目录位置。例如，Solaris 系统的默认 Calendar Server 安装使用路径 /opt/SUNWics5/ 作为服务器根目录。

(3) 用于安装所有与特定主机上的特定 Java Enterprise System Administration Server 关联的 Java Enterprise System 服务器的目录。另参见 [installation directory \(安装目录\)](#) 和 [instance directory \(实例目录\)](#)。

server-side rules (服务器端规则) 一组实现服务器端邮件过滤的规则。基于 Sieve 邮件过滤语言。

service (服务) (1) 服务器提供的一种功能。例如，Java Enterprise System Messaging Server 提供 SMTP、POP、IMAP 和 HTTP 服务。

(2) 一种针对一个或多个 [client \(客户机\)](#) 执行的软件功能。此功能可以是非常低的级别（如内存管理），也可以是较高的级别（如信用核查 [business service \(业务服务\)](#)）。较高级别的服务可以由一系列单项服务组成。服务可以是本地的（供本地客户机使用），也可以是分布式的（供远程客户机使用）。

service control manager (服务控制管理器) (仅限 Windows NT) 一种用于管理服务的管理程序。

service quality component (服务质量组件) Java Enterprise System 中包括的若干种 **system component (系统组件)** 之一。支持组件, 包括访问组件和管理组件, 为 **system service component (系统服务组件)** 提供支持。

service provider (服务提供者) 提供基于 web 服务的商业或非赢利性组织。可以包括 internet 门户、零售商、运输提供商、金融机构、娱乐公司、图书馆、大学和政府机构。

service stack (服务栈) 支持分布式企业应用程序所需的分布式服务的分层。该分层反映栈中较高级别服务对其下层服务的依赖性。

servlet (1) 使用 Java™ 编程语言编写的服务器端程序, web 服务器运行这些程序来生成内容以响应客户机请求。Servlet 在服务器端运行, 这一点与 applet 相似, 但 servlet 不使用用户界面。

(2) Servlet 类的一个实例。一个 servlet 就是在服务器上运行的一个可重用应用程序。在 Java Enterprise System Application Server 中, servlet 通过执行表示逻辑、调用业务逻辑及调用或执行表示布局, 充当应用程序中每次交互的中央分发程序。

servlet engine (servlet 引擎) 处理所有 servlet 元功能的内部对象。总体来说, 即为 servlet 提供服务的一组进程, 包括实例化和执行。

servlet runner (servlet 运行器) Servlet 引擎的组成部分, 可调用含有一个请求对象和一个响应对象的 servlet。参见 **servlet engine (servlet 引擎)**。

session (会话) (1) servlet 用于通过多个 **HTTP** 请求跟踪用户与 web 应用程序交互的对象。

(2) 客户机服务器连接实例。另参见 **client-server model (客户机/服务器模型)**。

(3) 对于 Java Enterprise System Portal Server, 指用户与一个或多个应用程序交互的序列, 从登录开始, 到注销或超时结束。

(4) 对于 Message Queue, 指用于发送和接收消息的单线程上下文。它可以是一个队列会话或一个主题会话。

session bean (会话 bean) 由客户机创建，并且通常只在单个客户机服务器会话期间存在的企业 bean。会话 bean 为客户机执行操作，如计算或访问其他企业 bean。虽然会话 bean 可以是事务性的，但它无法在系统发生崩溃时恢复。会话 bean 对象可以是无状态的（不与特定客户机相关联），也可以是有状态的（与特定客户机相关联），因此它们可以跨多个方法和事务维护对话状态。另参见 [stateful session bean \(有状态会话 bean\)](#) 和 [stateless session bean \(无状态会话 bean\)](#)。

session cookie (会话 cookie) 返回含有用户会话标识符的客户机的 cookie。另参见 [sticky cookie \(粘性 cookie\)](#)。

session key (会话密钥) 一种常见的加密技术，使用单独的密钥对两人之间的每个对话进行加密。

session timeout (会话超时) 一个指定的时间段，服务器可以在此时间段后使用户会话失效。

shared address (共享地址) 一个网络地址，可受运行在群集内节点上的所有可伸缩服务限制，以使这些服务在这些节点上可伸缩。一个群集可拥有多个共享地址，一个服务可限定到多个共享地址。

shared component (共享组件) Java Enterprise System 中包括的若干种 [system component \(系统组件\)](#) 之一。共享组件（通常是库）向其他系统组件提供本地服务。相反，[system service component \(系统服务组件\)](#) 向其他系统组件（或向 [application component \(应用程序组件\)](#)）提供分布式基础结构服务。

shaded component descriptor file (共享组件描述符文件) 含有特定共享组件的元数据的文件（通常为 XML 格式）。

shared folder (共享文件夹) 可由一个以上人员读取的文件夹。共享文件夹有一个所有者，他可以指定该文件夹的读取权限并可从共享文件夹中删除消息。共享文件夹还可以有一个仲裁人，他可以编辑、拦截或转发收到的消息。只有 IMAP 文件夹可以共享。另参见 [personal folder \(个人文件夹\)](#)、[public folder \(公共文件夹\)](#)。

shared-key cryptography (共享密钥密码学) 一种各方必须使用相同的密钥加密或解密密文的密码学。又称为对称密钥密码学。

SHTML (server-side include Hypertext markup language, 服务器端包含超文本标记语言) 一种包括有嵌入式服务器端包含 (server-side include, SSI) 的 HTML 文件。

Sieve 建议的邮件过滤语言。

simple authentication and security layer (简单验证和安全层) 参见 [SASL](#)。

simple index (简单索引) 一种目录列表, 仅显示文件名称, 不带任何图形元素。与复杂索引相对。

Simple Mail Transfer Protocol (简单邮件传输协议) 参见 [SMB protocol \(SMB 协议\)](#)。

Simple Network Management Protocol (简单网络管理协议) 参见 [SNMP](#)。

Simple Object Access Protocol (简单对象访问协议) 参见 [SOAP](#)。

SIMS Solstice Internet Mail Server™ 和 Sun Internet Mail Server™。

single field substitution string (单个字段替换字符串) 在重写规则中, 可动态重写主机的指定地址令牌和域地址的域模板的组成部分。另参见 [domain template \(域模板\)](#)。

single identity (单一身份) 用户因 Java Enterprise System 目录中的单用户条目而具有的身份。基于此单用户条目, 用户可以获准访问各种 Java Enterprise System 资源, 例如门户、Web 页以及诸如讯息传递、日历和即时讯息传递之类的服务。

single-instance resource (单一实例资源) 在整个群集范围内最多有一个资源可以处于活动状态的资源。

single logout (单点注销) 用户从身份认证提供者或服务提供者注销, 以及从该验证域的所有服务提供者或身份认证提供者注销的能力。

single-master replication (单主复制) 一种复制模型, 该模型只有一个服务器, 即主站, 允许对复制的数据具有 [LDAP](#) 写访问权限。在单主复制模型中, 供应者服务器或主服务器维护更改日志。

single sign-on (单点登录) (SSO) 一种功能，允许将用户对分布式系统中一项服务的验证自动应用到该系统中的其他服务。

(2) 一种情况，在此情况下用户的验证状态可跨单个虚拟服务器实例中的多个 J2EE™ 应用程序共享。参见 [SSO](#)。

(3) 当拥有联合身份的用户向身份认证提供者验证时建立的验证进程。由于该用户拥有联合身份，该用户可以不必重新验证即可访问附属的服务提供者。

site (站点) *robot* 为查找资源而访问的网络位置。在 [site definition \(站点定义\)](#) 中确定站点的地址以及要在该站点索引的文档的种类。

site definition (站点定义) 关于 *robot* 为寻找资源可以访问的位置的约束。可以使用站点定义将 *robot* 限定到一个特定的服务器、一个特定的服务器组或一个域。站点定义包括描述 *robot* 应从站点索引的文档类型的过滤器。

SIZE 一个 [SMTP](#) 扩展，可以使客户机向服务器声明某一特定消息的大小。服务器可以基于所声明的消息大小，告诉客户机其是否愿意接受该消息。服务器可以向客户机声明它愿意接受的最大消息大小。在 RFC 1870 中定义。

slapd.exe (仅限 Windows) 负责 Directory Server 所有操作的进程或服务。在 UNIX 系统中，与之对应的是 [ns-slapd](#)。

slave channel program (从通道程序) 接受由远程系统启动的传输的通道程序。另参见 [master channel program \(主通道程序\)](#)。

smart host (智能主机) 某个域中的邮件服务器，当其他邮件服务器无法识别收件人时，将消息转发到该邮件服务器。

SMB protocol (SMB 协议) 参见 [Server Message Block protocol \(Server Message Block 协议\)](#)。

SMTP (Simple Mail Transfer Protocol, 简单邮件传输协议) Internet 中最常用并受 Java Enterprise System Messaging Server 支持的电子邮件协议。该协议在 RFC 821 中定义，相关联的消息格式说明包含在 RFC 822 中。

SMTP AUTH 参见 [AUTH](#)。

SMTP proxy (SMTP 代理) SMTP 的变体, 将消息从网络中的一台计算机发送到另一台计算机, 并在 Internet 上用于对电子邮件进行路由。

sn attribute (sn 属性) 姓氏的 LDAP 别名。

SNMP (Simple Network Management Protocol, 简单网络管理协议) 用于交换有关网络活动数据的协议。通过 SNMP, 数据可在受管理的设备 (任何运行 SNMP 的设备, 如主机、路由器、您的 web 服务器, 以及您网络上的其他服务器) 和 NMS 之间传输。

SNMP master agent (SNMP 主代理) 在各种子代理和 NMS 之间交换信息的软件。

SNMP SOCKS 一种防火墙软件, 当受到防火墙软件或硬件 (例如, 路由器配置) 阻止而不能建立直接连接时, 该防火墙软件可建立防火墙内部到外部的连接。

SNMP subagent (SNMP 子代理) 收集有关受管理设备的信息, 并将该信息传递给主代理的软件。

SOAP (Simple Object Access Protocol, 简单对象访问协议) 一种协议, 可为调用分布在 Internet 内不同操作环境中的对象中的方法定义标准化方式。该协议将基于 XML 的数据结构和超文本传输协议 (Hypertext Transfer Protocol, HTTP) 相结合。

soft restart (软重启) 一种重新启动服务器的方式, 可使服务器通过重新读取其配置文件进行内部重启。软重启向进程发送 HUP 信号 (信号 1)。该进程本身不会像在硬重启中那样彻底终止。

SOIF (summary object interchange format, 摘要对象交换格式) 用于传送资源说明及其他类型的结构化对象的语法。每个资源说明表示为一个属性值对列表。SOIF 将文本和二进制数据作为值和带一些微小扩展的多值属性处理。SOIF 是一种允许许多资源说明以单个高效的流大量传输的流化格式。

SOIF attribute (SOIF 属性) 一种数据库属性类型。搜索数据库中的每个资源说明都拥有多个属性或字段。这些属性称为 SOIF 属性。

Solaris™ logical name (Solaris™ 逻辑名称) 该名称通常用于管理 Solaris™ 操作系统设备。对于磁盘，它们通常具有类似 `/dev/rdisk/c0t2d0s2` 的形式。对于每个 Solaris 逻辑设备名称，都存在一个基础 Solaris 物理设备名称。另参见 [DID name \(DID 名称\)](#) 和 [Solaris™ physical name \(Solaris™ 物理名称\)](#)。

Solaris™ physical name (Solaris™ 物理名称) 在 Solaris™ 操作系统中由设备驱动程序为设备指定的名称。该名称在 Solaris 计算机上显示为 `/devices` 树下的一个路径。例如，一个典型 SI 磁盘拥有一个类似下面形式的 Solaris 物理名称：`devices/sbus@1f,0/SUNW,fas@e,8800000/sd@6,0:c,raw`。另参见 [Solaris™ logical name \(Solaris™ 逻辑名称\)](#)。

Solstice DiskSuite™ software (Solstice DiskSuite™ 软件) SunPlex™ 系统使用的卷管理器。另参见 [virtual server class \(虚拟服务器类\)](#)。

spare node (备用节点) 可替换发生故障的活动节点的 HADB 节点。如果活动节点出现故障，备用节点会从镜像节点复制数据并转为活动状态。另参见 [HADB node \(HADB 节点\)](#)、[active node \(活动节点\)](#)、[mirror node \(镜像节点\)](#)、[data redundancy unit \(数据冗余单元\)](#)。

spider 参见 [robot](#)。

split brain (记忆分裂) 群集分成多个分区的状态，其中每个分区在不知道其他分区存在的情况下建立。

spoofing (电子欺骗) 一种试图对服务器进行访问或向其发送消息的客户机虚报其主机名的网络攻击形式。

SQL (structured query language, 结构化查询语言) 关系数据库应用程序中常用的语言。SQL2 和 SQL3 指明了该语言的版本。

SSL (secure socket layer, 安全套接字层) 由其他协议（如 HTTP 和 FTP）使用的一种安全、低层加密形式。SSL 协议包括服务器验证、传输数据加密以及可选客户机验证的置备。

SSL authentication (SSL 验证) 一种验证方法，该方法通过将客户机证书中的信息作为身份证明或验证 LDAP 目录中发布的客户机证书来使用安全性证书确认用户的身份。

SSL certificate (SSL 证书) 一种表示您或供应商已批准使用 *PKI* 对安全事务进行加密和解密的电子令牌。您在安装 Java Enterprise System Portal Server 软件时将创建一个自签名的“SSL 证书”。不过，您可以通过 Internet 从授权使用安全通信服务的证书供应商处获得“SSL 证书”。

SSO 参见 [single sign-on \(单点登录\) \(SSO\)](#)。

SSR 参见 [server-side rules \(服务器端规则\)](#)。

standard index (标准索引) 在默认情况下维护的索引。

starting points (起点) “搜索引擎” robot 为开始枚举资源而访问的站点列表。

state (状态) (1) 实体在任意给定时间的情况或状态。

(2) 一种分布式数据存储机制，可用于存储使用 Java Enterprise System Application Server 功能接口 `IState2` 的应用程序的状态。另参见 [conversational state \(会话状态\)](#)、[persistent state \(持久状态\)](#)。

stateful session bean (有状态会话 bean) 一个表示与特定客户机进行的会话，并跨多个客户机调用的方法自动维护状态的会话 bean。

stateless session bean (无状态会话 bean) 一个表示无状态服务的会话 bean。无状态会话 bean 是完全瞬态的，并封装了特定客户机在有限时间跨度内所需的临时业务逻辑片段。

static group (静态组) 通过枚举每个组成员静态定义的邮件组。另参见 [dynamic group \(动态组\)](#)。

static web content (静态 web 内容) 静态 HTML 文件、图像、applet Java™ 归档 (Java™ archive, JAR) 文件，以及其他任何可由 web 服务器在不使用 Java web 容器的情况下直接提供的内容。对于 Java Enterprise System Portal Server，web 文件安装在 web 服务器中（与动态 web 应用程序的位置相同）。

status event (状态事件) 用户的状态，包括是否联机。

sticky cookie (粘性 cookie) 返回客户机并强制该客户机始终连接到相同服务器进程的 *cookie*。另参见 [session cookie \(会话 cookie\)](#)。

sticky load balancing (粘性负载均衡) 一种 **load balancing (负载均衡)** 方法，在该方法中，初始客户机请求已达到负载均衡，但后续请求被引导至与初始请求相同的进程。

stop word (止用词) 对搜索功能标识为搜索功能不应搜索的词，例如，“the”、“a”、“an”和“and”。又称为丢弃词。

stored procedure (存储过程) 在 *spider* 中编写并在数据库中存储的语句块。可以使用存储过程执行任何类型的数据库操作，如修改记录、插入记录或删除记录。使用存储过程可藉由减少通过网络发送的信息量提高数据库性能。

streaming (流化处理) 管理数据如何通过 *HTTP* 通信的技术。当对结果进行流化处理，数据的第一部分可立即使用。当结果未流化处理时，在使用结果的任何部分以前必须先接收整个结果。流化处理提供了一种可使大数据量以更高效率返回的方式，从而提升了应用程序的性能体验。

strftime function (strftime 函数) 将日期和时间转换为字符串的函数。服务器在附加尾缀时使用此函数。strftime 函数具有一种特殊的日期和时间格式语言，服务器可以在尾缀中使用这种格式语言说明文件上次的修改日期。

subagent (子代理) 参见 **SNMP subagent (SNMP 子代理)**。

subdomain (子域) **fully qualified domain name (全限定域名)** 的倒数第二部分，用于标识拥有该域名的公司或组织内的分公司或部门（例如，support.example.com 和 sales.example.com）。子域并不是始终需要指定的。

subnet (子网) **IP address (IP 地址)** 的一部分，用于标识主机 ID 字块。

subordinate reference (次级参考) 目录服务器所持有的命名上下文的子命名上下文。另参见 **knowledge information (知识信息)**。

suborganization (子组织) 在 Java Enterprise System Directory Server Access Management Edition 中，指在某个组织下创建的对象，它为企业提供了对其部门和资源的更高控制粒度。例如，在设置 Java Enterprise System Portal Server 时，可能会在顶级对象 isp 下创建一个名为 mycompany 的子组织。

subschema entry (子模式条目) 此类条目包含部分目录树中的条目所使用的所有模式定义 (对象类、属性、匹配规则等项的定义)。

substring index (子串索引) 一种用于搜索的文件信息语言执行例程, 借此可对条目中的子串执行高效搜索。子串索引的每个索引键最多限定为三个字符。

sub suffix (子后缀) 根后缀下的分支。

suffix (后缀) 目录树中其下存有数据的条目的名称。同一目录中可以存在多个后缀。每个数据库只有一个后缀。

summary object interchange format (摘要对象交换格式) 参见 [SOIF](#)。

Sun™ Cluster software (Sun™ Cluster 软件) SunPlex™ 系统的软件部分。另参见 [SunPlex™ system \(SunPlex™ 系统\)](#)。

Sun Java™ System Communications Express 此软件提供了基于 web 的集成式通信和协作客户机, 用以满足企业用户访问电子邮件、日历和通讯录信息的需要。

Sun Java™ System Connector for Microsoft Outlook 一种插件, 通过它可将 Microsoft Outlook 作为桌面客户机与 Sun Java Enterprise System 一起使用。

Sun Java™ System Portal Secure Remote Access (SRA) SRA 允许大多数客户机设备通过安全连接访问个性化门户应用程序、内容、文件和服务。

Sun Java™ System Synchronization 此软件运行于 Microsoft Windows 个人计算机上, 用户可以利用它来使日历事件及任务与移动设备和个人信息管理器 (personal information manager, PIM) (如 Microsoft Outlook) 保持同步。

SunPlex™ system (SunPlex™ 系统) 集成的硬件及 Sun™ Cluster 软件系统, 用于创建高度可用和可伸缩的服务。

superuser (超级用户) (仅限 UNIX) UNIX 机器上现有的享有最高特权的用户。又称为根用户。超级用户拥有对机器上所有文件的完全访问权限。

supplier (供应者) 一种服务器，其中含有复制到使用者服务器的目录树或子树的原版。

supplier replica (供应者副本) 包含原版目录信息并且可以更新的副本。服务器可持有任意数目的主副本。

supplier server (供应者服务器) 在复制上下文中，将持有复制到另一不同服务器的副本的服务器称为该副本的供应者服务器。

switchback (切换回原状态) 参见 [failback \(故障恢复\)](#)。

switchover (切换) (仅限 UNIX) 依序将资源组或设备组从群集中的一个主站(节点)转移到另一个主站(或多个主站，如果资源组是为多个主节点配置的)。切转由管理员使用 `scswitch` 命令启动。

symlinks (符号链接) (仅限 UNIX) 一个特殊的文件或目录，它指向另一文件或目录，从而使两个文件或目录具有相同的内容。

symmetric encryption (对称加密) 加密和解密使用同一密钥的加密方法。例如，“数据加密标准”(Data Encryption Standard, DES) 就是一种对称加密算法。

symmetric key cryptography (对称密钥密码学) 参见 [shared-key cryptography \(共享密钥密码学\)](#)。

synchronization (同步) (1) 由主目录服务器对副本目录服务器进行的数据更新。
(2) [MTA](#) 目录高速缓存的更新。

system administrator (系统管理员) 负责管理 Java Enterprise System 软件以及部署 Java Enterprise System 应用程序的人员。

system component (系统组件) 包括在 Java Enterprise System 中并通过 Java Enterprise System 安装程序进行安装的任何软件包或任何一组软件包。有多种系统组件：提供分布式基础结构 [service \(服务\)](#) 的 [system service component \(系统服务组件\)](#)、通过提供访问和管理服务支持系统服务组件的 [service quality component \(服务质量组件\)](#)，以及向其他系统组件提供本地服务的 [shared component \(共享组件\)](#)。

system index (系统索引) 不能删除或修改的索引，因为它对 Directory Server 操作是必不可少的。

system service (系统服务) 定义 Java Enterprise System 所提供的独特功能的一项或多项分布式 **service (服务)**。系统服务通常需要若干 **service quality component (服务质量组件)** 和/或若干 **shared component (共享组件)** 的支持。

system service component (系统服务组件) Java Enterprise System 中包括的若干种 **system component (系统组件)** 之一。系统服务组件提供以下主要的 Java Enterprise System 基础结构服务：门户服务、通信和协作服务、身份认证和安全服务、web 和应用程序服务，以及可用性服务。

System Service Processor (系统服务处理器) 在 Sun Enterprise™ 10000 服务器配置中，群集外部专门用来与群集成员进行通信的设备。

table (表) 数据库中已命名的一组相关数据，以行和列表示。

takeover (接管) 参见 **failover (故障转移)**。

target (目标) 在访问控制上下文中，目标指明了具体 **ACI** 应用于的目录信息。

target entries (目标条目) **CoS** 范围内的条目。

task (任务) 在客户机端的 Calendar Express 中，指定待做事情的日历组成部分。在服务器端，任务又称为 **todo (待办事项)**。

taxonomy (分类法) Java Enterprise System Portal Server 搜索引擎中资源的分类系统。

TCP/IP (Transmission Control Protocol/Internet Protocol, 传输控制协议/Internet 协议) Internet 和企业网络采用的主要网络协议。

telnet (远程登录) Internet 协议套件中的虚拟终端协议。借此，某一主机的用户可以登录到远程主机，并象该主机的正常终端用户那样进行交互。

telnet proxy (远程登录代理) 位于远程登录客户机和远程登录服务器之间起智能中继作用的应用程序。

template entry (模板条目) 参见 **coserver (协同服务器)**。

terminal concentrator (终端集中器) 在非 Sun Enterprise™ 10000 配置中，群集外部专门用来与群集成员进行通信的设备。

thread (线程) 进程内的执行序列。一个进程可以允许多个同时执行的线程，在这种情况下，该进程为多线程进程。如果一个进程按顺序执行每个线程，则该进程为单线程进程。

timeout (超时) 一段指定的时间，过了这段时间之后服务器将放弃尝试完成似乎已挂起的服务例程。

time zone (时区) 使用相同时间的地理区域。从 -12 至 +12 (GMT 为 0) 共有 25 个小时时区。每个时区以 GMT 为基准测定。多数时区都有由三个字母组成的本地化缩写名称。Calendar Server 还使用时区 ID (time zone ID, TZID) (如美国/洛杉矶或亚洲/加尔各答) 来标识时区。

TLS (Transport Layer Security, 传输层安全) 一种协议，在传输层提供加密和认证，这样，无需对客户机和服务器应用程序进行较大更改，数据流便能通过安全通道。它是 [SQL](#) 的标准，一种基于公共密钥的协议。

todo (待办事项) 在服务器端，指定待做事情的日历组成部分。在客户机端的 Calendar Express 中，待办事项称为 **task (任务)**。

top (仅限 UNIX) 一些 UNIX 系统中用于显示系统资源当前使用状况的程序。

topic (主题) 管理员为实现发布和订阅传送模型所创建的对象。可将主题视为内容分层结构中负责收集和分发针对该主题消息的节点。以主题作为中介，便将消息发布者与消息订户分隔开来。

top-level administrator (顶级管理员) 拥有以下管理权限的用户：可通过“用于讯息传递和协作的委托管理员” GUI 或 CLI，在整个 Messaging Server 名称空间中创建、修改和删除邮件用户、邮件列表、家庭帐户和域。默认情况下，此用户可以担任拓扑中所有讯息传递服务器的消息存储管理员。

top-level domain authority (顶级域授权) 主机名称分类的最高类别，通常表示域的组织类型（例如，.com 代表公司，.edu 代表教育机构）或域的原属国家/地区（例如，.us 代表美国，.jp 代表日本，.au 代表澳大利亚，.fi 代表芬兰）。

topology (拓扑) 目录树在物理服务器间的划分方式以及这些服务器相互链接的方式。

transaction (事务) 一组数据库命令，这些命令的成功或失败都密不可分。涉及的所有命令必须都成功，整个事务才算成功。

(2) 必须完成或完全回滚的原子工作单元。

transaction attribute (事务属性) 控制事务范围的属性。

transaction context (事务上下文) 事务的范围（本地或全局）。参见 [local transaction \(本地事务\)](#)、[global transaction \(全局事务\)](#)。

transaction isolation level (事务隔离级别) 确定数据库中并发事务相互之间的可见程度。

transaction manager (事务管理器) 控制全局事务的对象，通常使用 [XA protocol \(XA 协议\)](#)。另参见 [global transaction \(全局事务\)](#)。

transaction recovery (事务恢复) 分布式事务的自动或手动恢复。

transience (瞬态) 一种协议，当资源不用时将其释放。与 [persistence \(持久性\)](#) 相对。

transient failure (瞬态故障) 消息处理过程中出现的错误情况。远程 [MTA](#) 不能在消息送达时处理消息，但也许能在以后处理该消息。本地 [MTA](#) 将消息送回到队列，并安排在以后某一时间重新传输该消息。

Transmission Control Protocol (传输控制协议) (TCP) Internet 协议套件中的基本传输协议，用于在两主机间提供面向连接的可靠流服务。

Transmission Control Protocol/Internet Protocol (传输控制协议/Internet 协议) (TCP/IP) 为 Internet 协议套件使用的网络协议集合所起的名称。该名称指的是套件的两个主要网络协议：作为传输层协议的 TCP (传输控制协议) 和作为网络层协议的 IP (Internet 协议)。

Transport Layer Security (传输层安全) (TLS) SSL 的标准化形式。另参见 [secure socket layer \(安全套接字层\)](#)。

transport protocols (传输协议) 用于在两个 *MTA* 之间提供消息传送手段的协议，例如 SMTP 和 X.400。

trust database (信任数据库) 含有公共和专用密钥的安全性文件。又称为 [key-pair file \(密钥对文件\)](#)。

trusted provider (受信提供者) [circle of trust \(信任环\)](#) 中的一组服务提供者和身份认证提供者之一。用户可以在安全的环境中与受信提供者进行事务处理和通信。

UA 参见 [user agent \(用户代理\)](#)。

UAProf 由“开放移动联盟”定义的规范，允许移动设备将其功能传达给网络服务器。

UBE 参见 [unsolicited bulk email \(未经许可的海量电子邮件\)](#)。

UDDI (Universal Description, Discovery, and Integration, 通用描述、搜索和集成) 为搜索和集成提供全球 web 服务注册。

uid (仅限 UNIX) 与 UNIX 系统上的每个用户相关联的唯一号码。

unified messaging (统一讯息传递) 此概念是指对电子邮件、语音邮件、传真和其他通信形式使用单一消息存储。Java Enterprise System Messaging Server 是实现完全统一讯息传递解决方案的基础。

uniform resource indicator (统一资源指示符) 参见 [URI](#)。

uniform resource locator (统一资源定位符) 参见 [URL](#)。

uninstallation (卸载) 彻底删除软件组件的过程。

universal principal name (通用主要名称) 登录用户的值，包括并在一起的登录名及该用户所属的域。例如，域 `example.com` 中用户 `bill` 的“通用主要名称”是 `bill@example.com`。又称为 UPN。

unsolicited bulk email (未经许可的海量电子邮件) 大批散发者通常出于商业目的的发来的未请求和不需要的电子邮件。又称为垃圾邮件。

upgrade (升级) 在现有产品位码之上安装更新产品位码的过程。

upper reference (上层参考) 在 *DIT* 中，所持命名上下文高于您所在目录服务器命名上下文的目录服务器。

URI (universal resource identifier, 统一资源标识符) 描述域中的特定资源。它在本地被描述为基目录的子集，这样，`/ham/burger` 便为基目录，而 URI 指定的是 `toppings/cheese.html`。相应的 URL 将是 `http://domain.port/toppings/cheese.html`。

URL (uniform resource locator, 统一资源定位符) 服务器和客户机用来请求文档的寻址系统。URL 通常称为位置。URL 的格式为 `protocol://machine.port/document`。`http://www.example.com/index.html` 是一个 URL 样例。

URL database repair (URL 数据库修理) 对已损坏的 URL 数据库进行修复和更新的过程，损坏的原因可能是：软件故障、系统崩溃、磁盘失灵或文件系统已满。

URL mapping (URL 映射) 文档目录物理路径名到用户定义别名的映射过程，这样做使得目录中的文件只需引用目录的别名，而无需引用文件的完整物理路径名。您不用将文件指明为 `usr/JES/servers/docs/index.html`，而可将该文件指明为 `/myDocs/index.html`。此映射使得用户不必知道服务器文件的物理位置，从而增强了服务器的安全性。

URL pool (URL 池) robot 要处理的 URL 列表。当 robot 启动时，URL 池由起始点组成，但该池会随着枚举过程中发现的资源而迅速增大。

use case (使用案例) 一项特定的最终用户任务或一组任务，这些任务由 [distributed enterprise application \(分布式企业应用程序\)](#) 执行，是设计、测试和衡量应用程序性能的基础。

user (用户) 使用应用程序的人员或服务。从程序学角度而言，用户由用户名、密码以及应用程序借以识别用户的属性集组成。另参见 **group (组)** 和 **role (角色)**。

user account (用户帐户) 访问目录服务器上以条目形式维护的服务器时所用的帐户。

userAgent 对于 Portal Server Mobile Access，它是引用 HTTP 用户代理报头的属性。用户代理报头对于特定移动设备通常是唯一的，并且可用来检测和检索某客户机类型的数据。

user agent (用户代理) 允许用户创建、发送和接收邮件的客户机组件，如 Netscape™ Communicator。又称为 UA。

user entry (用户条目) 说明每个用户信息的字段（必填和可选）。例如，标识名、全名、称谓、电话号码、寻呼机号码、登录名、密码、主目录等等。又称为用户配置文件。

user folders (用户文件夹) 用户的电子邮箱。

user group (用户组) Message Queue 客户机用户所属的组，这样做是为了授权访问 Message Queue 消息服务器资源，如连接和目标。

User/Groups Directory Server (用户/组 Directory Server) 维护组织中用户及组信息的 Directory Server。

user ID (用户 ID) (1) 用户标识。用于向系统标识用户的唯一字符串。又称为 userid。

user quota (用户配额) 分配给用户的电子邮件空间数量，由系统管理员进行配置。

user provisioning (用户置备) 使服务对最终用户可用或向最终用户提供服务访问权的过程。置备涉及身份认证、策略和用户帐户管理活动，如在目录中为每个最终用户创建帐户并以各种服务所需的用户特定信息填充帐户。

user session (用户会话) 服务器跟踪的一系列用户应用程序交互。会话维护着用户状态、持久对象以及身份验证。

UUCP (UNIX to UNIX Copy Protocol, UNIX 到 UNIX 复制协议) (仅限 UNIX) 用于在受准 UNIX 系统间进行通信的协议。

vanity domain (虚域) 与单个用户关联而不与特定服务器或托管域关联的域名。虚域通过 `MailAlternateAddress` 属性指定。虚域没有与域名相应的 *LDAP* 条目。虚域适用于希望自定义域名而不想负担支持自身托管域所需管理开销的个人或小型组织。又称为自定义域。

/var/mail 此名称通常用来指 Berkeley 式收件箱, 在此类收件箱中, 新邮件依序存储在单个平面式文本文件中。

Veritas Cluster Server Veritas Software 提供的高可用性群集软件, 可与 Java Enterprise System Messaging Server 集成在一起。

VERITAS Volume Manager (VERITAS 卷管理器) SunPlex™ 系统使用的卷管理器。另参见 [virtual server class \(虚拟服务器类\)](#)。

versioning (版本处理) 参见 [dynamic reloading \(动态重新加载\)](#)。

virtual domain (虚拟域) (1) ISP 托管域。

(2) 由“讯息传递多路复用器”添加到客户机用户 ID 的域名, 用于 LDAP 搜索和登录到邮箱服务器。另参见 [document type definition \(文档类型定义\)](#)、[hosted domain \(托管域\)](#)。

virtual list view index (虚拟列表视图索引) 一种过滤方法, 如果带用户界面的客户机使用虚拟列表视图扩展, 采用此方法可加快 Directory Server 控制台 (或其他图形用户界面) 中条目的显示速度。可在目录树的任何分支上创建虚拟列表视图索引来提高特定搜索的显示性能。又称为浏览索引。

virtual private network (虚拟专用网络) 一种网络, 它具有常规网络的外观和功能, 但与公共网络中的专用网络相似。通过在较低协议层进行加密, 可通过未实施这种加密的不安全网络 (通常是 Internet) 进行安全连接。VPN 通常比使用专线的真正专用网络成本低。VPN 依靠在两端使用相同的加密系统。加密可由防火墙软件或路由器执行。

virtual server (虚拟服务器) (1) 针对特定 URL 提供内容服务的虚拟 web 服务器。多个虚拟服务器可以使用相同或不同的主机名、端口号或 IP 地址来提供内容服务。HTTP 服务可根据 URL 将收到的 web 请求定向到不同的虚拟服务器。又称为虚拟主机。

(2) 通过虚拟服务器可以为一个已安装的服务器设置多个域名、IP 地址以及服务器监视功能。

virtual server class (虚拟服务器类) 共享 obj.conf 文件中相同基本配置信息的虚拟服务器集合。

voice Portal Desktop (语音 Portal 桌面) 通过电话或类似设备所呈现的 Portal Server 站点的音频表现形式。

voiceXML 一种标记语言，用于为交互式语音应答应用程序创建音频对话。

VoIP (IP 语音) 基于 IP 网络提供语音通话的技术。

volume manager (卷管理器) 一种软件产品，通过磁盘分散读写、级联、镜像以及元设备或卷的动态增长来提供数据可靠性。

VPN 参见 [virtual private network \(虚拟专用网络\)](#)。

VPN gateway (VPN 网关) VPN 的入口点。通常受防火墙保护。

VRFY 用于检验用户名的 SMTP 命令。在 RFC 821 中定义。

WAP (Wireless Application Protocol, 无线应用程序协议) 一种通过无线通信运行应用程序的开放标准。

WAR file (WAR 文件) 参见 [web application archive \(web 应用程序归档\)](#)。

WCAP (Web Calendar Access Protocol, Web 日历访问协议) 一种基于命令的高级协议，客户机使用此协议与 Calendar Server 进行通信。

web application (web 应用程序) 由 servlet、采用 **JSP™ technology (JSP™ 技术)** 创建的页面、HTML 文档以及其他 web 资源组成的集合，其中可能包括图像文件、压缩归档及其他数据。web 应用程序可以打包成 web 归档 (WAR 文件)，也可存在于开放目录结构中。Java Enterprise System Application Server 还支持一些非 Java web 应用程序技术，如 **SHTML** 和 **CGI**。

web application archive (web 应用程序归档) 一种归档文件，内含压缩形式的完整 web 应用程序。Java Enterprise System Web Server 不能访问 WAR 文件中的应用程序。必须先解压缩 web 应用程序 (使用 wdeploy 实用程序进行部署)，Java Enterprise System Web Server 方可为其提供服务。

web cache (web 高速缓存) Java Enterprise System Application Server 的一个功能，利用此功能，servlet 或采用 **JSP™ technology (JSP™ 技术)** 创建的页面可将其结果在高速缓存中持续存放特定的一段时间，以提高性能。在这段持续时间内对该 servlet 或 JSP 页面的后续调用将得到高速缓存中的结果，这样，该 servlet 或 JSP 页面便不必重新执行了。

web connector plug-in (web 连接器插件) 对 web 服务器进行的扩展，使得 web 服务器能够与 Java Enterprise System Application Server 进行通信。

web container (web 容器) 参见 **container (容器)**。

webmail (web 邮件) 对基于浏览器的电子邮件服务的通称。基于浏览器的客户机称为“瘦”客户机，因为更多的处理是在服务器上进行的。这种客户机访问的邮件始终存储在服务器上。另参见 *Messenger Express*。

web module (web 模块) 单独部署的 web 应用程序。参见 **web application (web 应用程序)**。

web server (web 服务器) (1) 存储和管理 HTML 页面和 web 应用程序 (但不是全部 J2EE 应用程序) 的主机。web 服务器响应来自 web 浏览器的用户请求。

(2) 对诸如 HTTP、FTP 等 web 请求作出响应的应用程序。

(3) 为提供“万维网”访问而配备的软件程序或服务器计算机。web 服务器可接纳用户请求、检索请求的文件或应用程序，以及发出出错消息。

web server plug-in (web 服务器插件) HTTP 反向代理插件，借此可指示 Java Enterprise System Web Server 或 Java Enterprise System Application Server 将特定 HTTP 请求转发给另一服务器。

web service (web 服务) 一种服务，它符合为实现可访问性、服务封装和搜索功能而制订的标准化 Internet 协议。这些标准包括 SOAP (简单对象访问协议) 讯息传递协议、WSDL (Web 服务定义语言) 接口定义以及 UDDI (通用搜索、描述和集成) 注册标准。

(2) 通过 web 提供的一种服务。一种自包含、自描述的模块化应用程序，可以跨 Internet 或内联网接受来自系统的请求、对该请求进行处理并返回响应。

web service consumer (web 服务使用者) web 服务使用者通过向 Web 服务提供者发出请求，调用 Web 服务提供的操作。

web service provider (web 服务提供者) web 服务提供者根据 Web 服务使用者发来的请求履行 Web 服务。它可以与使用它的 Web 服务使用者在同一 Java™ 虚拟机上运行。

wildcard (通配符) 搜索字符串中的特殊字符，可以代表一个或多个其他字符或字符范围。

Windows CGI (仅限 Windows NT) 采用基于 Windows 的编程语言 (如 Visual Basic) 编写的 CGI 程序。

wireless desktop dispatcher (无线桌面分发程序) 一个组件，它确定了将用户请求路由到标准 Portal 桌面、移动 Portal 桌面还是语音 Portal 桌面。

withdrawn patch (撤除的修补程序) 已从分发系统中删除的修补程序。

WML (wireless markup language, 无线标记语言) 基于 XML 的标记语言，它是 WAP 的组成部分。

workgroup (工作组) 局部工作组环境，在此环境中，服务器在局部办公室或工作组内执行自身的路由选择和传送。部门间的邮件被路由到主干服务器。另参见 [backbone \(主干\)](#)。

WSDL (web services description language, web 服务描述语言) 一种基于 XML 的语言，用于以标准化方式定义 web 服务。描述 web 服务的三个基本属性: web 服务的定义、该 web 服务的访问方式以及该 web 服务的位置。

X.400 一种消息处理系统标准。

X.500 standard (X.500 标准) ISO/ITU-T 文档集，其中概括了建议在 Directory Server 实现中使用的信息模型、对象类和属性。*LDAP* 是 X.500 标准所使用的“目录访问协议”(Directory Access Protocol, DAP) 的轻量版本。

XA protocol (XA 协议) 用于分布式事务的数据库行业标准协议。

XHTML (extensible hypertext markup language, 可扩展超文本标记语言) HTML 4.0 的改订方案，可通过添加新元素和属性进行扩展。

XML (extensible markup language, 可扩展标记语言) 万维网联盟 (World Wide Web Consortium, W3C) 所开发的一种灵活的编程语言，用于创建公用信息格式以及在 web、内联网和其他领域实现格式及数据共享。XML 具有可扩展性，因为它与 HTML 不同，其标记符号没有数量限制，并且是自定义的。Calendar Server 使用 XML 和 XSL 生成 Calendar Express 用户界面。

XML namespace (XML 名称空间) 一种标准，允许为 DTD (文档类型定义) 所定义的元素名称集指定唯一的标签。可将使用该 DTD 的文档加入到其他任何文档中，而不会引起元素名称间的冲突。这是因为该 DTD 中定义的元素随后会被唯一地标识。举例来说，如此解析程序便可确定何时应按您的 DTD 而不是另一文档类型定义的 DTD 来解释元素。

XSL (extensible style language, 可扩展样式语言) 一种用于创建 XML 样式表的语言，它与用于 HTML 的层叠样式表 (cascading style sheet, CSS) 类似。在 XML 中，内容和表示是分开的。XML 标记并不指出其显示方式。XML 文档得先进行格式化方可阅读，格式化通常是用样式表来完成的。样式表由格式化规则组成，这些规则规定了特定 XML 标记如何影响文档在计算机屏幕或打印页面上的显示。

XSLT (extensible style language transformation, 可扩展样式语言变换) XML 样式表在将一种 XML 文档形式转变为另一种 XML 形式时所使用的语言。这种转变在电子商务和电子业务中特别有用，因为它可为多个平台上的各种 XML 文档编码提供一个通用的标准。

Zulu time (祖鲁时间) *GMT* 和 UTC (协调通用时间) 的军方称呼。