



Lotus Domino/Notes 技术丛书

4

R5.1

Lotus Developer's Guide

应用开发指南

本丛书编委会 主编



本书光盘内容包括：
与本书配套的电子书



北京希望电子出版社
Beijing Hope Electronic Press
www.bhp.com.cn

第 1 章 Lotus Notes 和 Domino

Lotus Notes 是一个企业级和工作组级的计算环境，它帮助人们有效地协同工作，不受平台或技术、组织以及地理位置和时间的限制。基于 Lotus Notes 的信息能够在任何时间和距离实现共享。

对于开发提供个人和工作组通讯、团队和工程合作以及任务和活动协调一致的 Internet 和 Intranet 应用，Lotus Notes 是理想的开发系统。Lotus Notes 具有独一无二的群件能力，具备安全模型和应用开发环境，提供给用户一个在整个企业开发和传送应用的基础结构。

Lotus Notes 还是一个开发和配置群件应用的客户/服务器平台。在它的最底层，Lotus Notes 是一个文档数据库，其中可存储文本和其他信息，比如，图像、幻灯片和电子表格。Lotus Notes 提供在整个企业通过复制发布信息的能力，同时能够提供足够的安全性，使得只有必须看到信息的人才能看到信息。

Lotus Notes 包含三个主要部分：

- Domino 服务器，服务器端；
- Lotus Notes 客户，客户端；
- 运行在服务器和客户端的服务。

1.1 Domino 服务器

Domino 是一个应用和信息的服务器，集成了一系列的服务，使你很容易地在 Internet 和公司的 Intranet 上创建安全的、交互式的商业解决方案。利用 Domino，可以快速地建立、管理和实施应用，帮助本公司合作伙伴以及客户在线地合作和协调重要的商业活动。Domino 支持许多种客户程序和设备，包括 WEB 浏览器，Lotus Notes 客户端以及各种 Mail 客户。

Domino 具有跨平台性，支持 Microsoft Windows 95、Window NT (Intel 和 DEC Alpha)，IBM OS/2, OS/390, OS/400, and AIX, Hewlett Packard HP_UX, Sun Solaris (SPARC and Intel Edition), Novel NetWare。

Domino 为 Lotus Notes 用户和其他 Domino 服务器提供存贮、共享数据库和邮件路由的服务。

Domino 服务器也是一个 Web 服务器，集成了 Lotus Notes 数据库和应用，并且支持任何浏览器作为客户端。Lotus Notes 应用可以通过 Internet 或 Intranet 上的浏览器访问。

1.2 Lotus Notes 客户端

Lotus Notes 为用户提供三种客户程序：

- Lotus Notes Mail 是一个高级消息客户，拥有完全集成的日历与计划、任务管理、讨论、基本工作流以及浏览器集成。它可以满足那些只需要高级信息传递能力，

而不需要访问自定义 Intranet 应用的用户的要求。它完全支持移动用户, 包括通过浏览 Internet 访问和管理在线和离线的 Web 信息;

- Lotus Notes Desktop 是 Lotus 战略级的 Intranet 的客户, 是一个把群件、消息传递能力和 Web 访问结合在一起的集成工作环境。Notes 的对象服务使个人和组织可以支持丰富的、新创的、集成的 E-mail、工作流、合作、日程和调度, Lotus Notes Desktop 支持移动用户、强大的自定义商业应用以及通过浏览 Internet 访问和管理 Web 信息;
- Lotus Notes Designer for Domino 是改进的 Notes 设计客户端, 用来开发 Domino 应用。它功能齐备, 用来开发强大的客户/服务器和 Internet 应用, 包括使用户可以改进 Web 站点创建、设计和管理的各个方面。Lotus Notes designer for Domino 非常适合设计 Web 站点 Notes 应用的开发者和那些需要安全的、协同工作的商业站点的 Web 开发者。

Lotus Notes 客户端可以运行在如下的通用操作系统上: Microsoft Windows3.1、Windows 95、Windows NT、IBM OS/2、AIX、Hewlett Packard HP_UX、Sun Solaris(SPARC 和 Intel 版本)、Apple Macintosh OS (PowerPC and 68K), 这些操作系统提供了与 Domino 服务器通信的手段, 使你可以在客户/服务器环境中使用共享数据库和收发邮件。

人们把 Lotus Notes 客户用于收发电子邮件、制定日程表、编制计划、访问 Web 以及用于基于 Domino 的应用, Lotus Notes 客户和服务器利用大量的软件服务去提供消息传递、通信和合作等各方面应用。

Lotus Notes designer for Domino 4.6 是一个增强的 Notes 客户, 包括对创建那些应用程序的开发者的特别支持, 这些应用程序由 Domino 服务器同时发送给 Notes 客户和 Web 浏览器。它具备 Lotus Notes 客户的所有特性, 例如支持 Internet Mail 系统, 支持增强的个人信息管理 (PIM) 功能, 并且支持与字处理器和桌面套件的更好的集成。它还包括许多对 Notes 开发环境的补充和增强, 从而利用 Notes 开发者的知识基础, 同时提供对 Web 更加友好的工具, 这些提高的方面包括:

- 在 Notes 用户界面中, 增强了对超文本标记语言 (HTML) 创作的支持;
- 集成了对 JAVA 的支持;
- 易于开发可以同时 Notes 和 Web 浏览器上显示的应用;
- 丰富的工具, 其中包括 Lotus BeanMachine, 可以快速、容易地通过 Java Applet 向 Web 应用中增加多媒体特性, 而无需编程。

1.3 服 务

下面列出了用于 Lotus Notes 和 Domino 环境中的服务, 大多数服务都适用于 Lotus Notes 客户和 Domino 服务器。

对象存贮

Domino 数据库中的文档可容纳任意数目的对象和数据类型, 包括文本、复杂文本 (rich text)、数值数据、结构化数据、图形、图像、声音、视频、文件附件、嵌入对象以及 Java

和 ActiveX applet。内嵌的全文检索引擎，使检索和搜索文档很容易。对象存贮使 Domino 应用可以动态表示基于用户身份、用户偏爱、用户输入和时间等变量的信息。

目录

用单一目录管理所有关于服务器和网络配置、应用程序管理和安全的资源目录信息。Domino 中包括了在 Windows NT 和 Domino 之间同步的用户帐号，并且符合轻量级目录访问协议 (LDAP)。目录是轻松管理 Internet 和 Intranet 应用，并保障其安全的基础。

安全

Domino 安全模型提供用户授权、数字签名、灵活的访问控制和加密。Domino 的安全性使你可以把 Intranet 应用扩展到客户和商业伙伴中。

复制

双向复制自动完成不同地理位置站点上的应用和信息的分发和同步。复制使你的商业应用不论时间和地点，对公司和世界上的用户都可以使用。Lotus Notes 复制在后续有详细说明。

消息传递

高级的客户/服务器消息传递系统具有内置的日程安排和计划功能，使个人和组织能够很容易地发送和共享信息。消息传输代理把系统无缝地扩展到简单邮件传输协议 (SMTP)、多目标 Internet 邮件扩展 (MIME)、x.400 和 CC: mail 消息传递环境。Domino 消息传递服务用单一服务器支持多种邮件客户——邮局协议 V3 (POP3)、Internet 消息访问协议 (IMAP4)、消息应用编程接口 (MAP1) 和 Lotus Notes 客户。

工作流

工作流引擎按照应用中定义的进程处理分发、路由和跟踪文档，它使你能够在公司内部、与客户、合作伙伴和供应商之间协调和合理安排关键的商业活动。

代理

代理能够使频繁运行的进程自动化，消除繁琐的管理任务，加速商业应用。代理可以被时间或商业应用中的事件触发，代理可运行在 Domino 服务器端和 Lotus Notes 客户端。

开发环境

Notes designer for domino 是具有集成开发环境的通用客户软件，通过它可以无冲突地访问 Domino 服务器的所有特性。Notes designer for domino 的特征是对它的改进使得高级 Web 站点的开发变得快而容易。它还包含了两个有用的工具：Lotus Bean Machine for Java 和 Lotus Notes Global designer。利用 Bean Machine，不须编写任何 Java 代码即可在数分钟内创建多媒体 Java applet。用 Notes Global designer，你的应用可以立刻在许多不同的语言环境中运行。

可扩展性、可靠性和高级服务

可选的高级服务使你能够群集多达 6 个 Domino 服务器, 从而提供可扩展性和容错特性, 并最大化群件和消息传递应用程序的可用性。实时复制技术保持多个服务器同步。

注意 Domino 服务器不同于文件服务器, 文件服务器提供对打印机和应用程序等共享资源的访问, 并管理网络行为。Domino 是一个应用级的服务进程, 提供有效地管理通信和应用所必需的服务。

1.4 Lotus Notes 数据

与管理表的关系型数据库不同, Lotus Notes 数据库管理文档。结构化或非结构化数据与相关的程序逻辑存贮在数据库中。

Lotus Notes 数据库实质上是一个带着相关数据的完整的应用程序, 其中包含了所有的视图、表单、代理、域和管理数据所需代码。Lotus Notes 数据库既可驻留在 Domino 服务器上, 也可驻留在 Lotus Notes 工作站上。

Lotus Notes 文档是在数据库中存贮数据的基础。Lotus Notes 数据库也有许多特性, 表单和视图就是其中主要的两个。

文档

Lotus Notes 文档是一个数据的容器, 类似数据库管理系统 (DBMS) 中的行, 每一行文档包含许多域。在创建时给予每一个文档唯一的标识, 它被用做 Lotus Notes 数据库的主关键字。

表单中包含对文档的设计, 它指定文档的域中包含什么数据, 如数值、日期、文本、复杂文本等。

表单

表单用来创作、读出和编辑文档, 应用开发者设计表单作为文档创建模板或者通过它来查看文档数据, 不同的表单可以用来查看同一个文档。表单由文本、域、按钮和程序代码等元素组成。通过表单访问数据时, 这些元素用来操纵文档数据。

视图

视图用来对文档进行排序、显示、过滤和分类, 应用开发者设计具有特定选择标准的视图, 使相关文档以给定的顺序列表显示。视图允许切换浏览多个文档。文档在视图中打开, 通过视图访问。视图还包含另外一些设计元素, 如列、按钮、程序代码。它们使你可以操纵或改变多个文档。

第 2 章 Lotus NotesPump

Lotus NotesPump 是企业级规模的数据分发服务器,使你可以构建在数目不断增加的关系型数据源和基于因特网的数据源之间读、写数据的应用程序。

用 NotesPump 服务器技术,你可以在某一个预定的时间,基于事件驱动,或者基于某一特定商业逻辑条件转移或传递数据。必要时,最终用户也可以直接从后端数据库中得到数据。

经常可靠地访问公司的数据有很多好处:特别是,用户可以得到更精确、准时的数据,可以获得第一手的公司信息,更精确、准时的数据可以用来及时地反馈和更好的管理商业运作。

NotesPump 支持多个主流数据库之间的连接,为 Domino、IBM DB2、Oracle 7/8、Sybase System 10/11 和 ASCII 纯文本文件系统提供本地的服务器连接。另外两种连接是 ODBC 连接和 Information Builders 公司的 EDA/SQL 连接,使 NotesPump 服务器可以访问 70 种以上的其他关系型连续数据源。

NotesPump 服务器的另一个重要特点是,虽然服务器是由 Lotus Notes 应用来管理的,但是 NotesPump 支持在如 DB2 和 Oracle 这样不相似且非 Notes 数据库之间传送和复制数据。因此,系统管理员能采用单一的企业 NotesPump 系统底层结构来管理不断增长的关系型数据源。

NotesPump 服务器系统是——一个保持在增长的关系型数据源之间进行数据传输操作,而不必编写程序的系统。通用的、可重用的程序逻辑内建在 NotesPump 系统中。可以通过定义参数利用这些逻辑而不是编写新的程序代码。

2.1 结 构

Lotus Notespump 是一个基于服务器的工具,按命令或计划在数据源之间提取传送数据。

如图 2.1 所示那样, Lotus NotesPump 系统服务器是由一个服务器部件、NotesPump 服务器引擎和它的管理应用、一个被叫做 NotesPump Administrator 的 Lotus Notes 数据库组成的。当 NotesPump 管理数据库工作时,系统管理员创建和操作 Notes 文档, Notes 文档是由一个多线程、多进程的任务 NotesPump 服务器引擎运行的指令集。

NotesPump 可以适用于 Windows NT, OS/2, HP-UX, AIX, Solaris SPARC, Solaris Intel 版本和 OS/400。

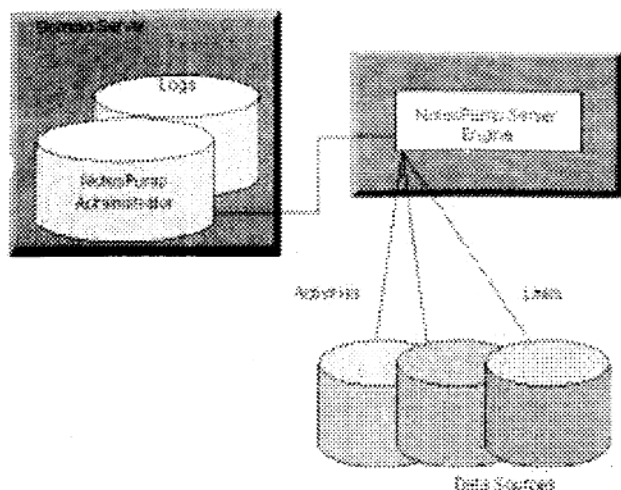


图 2.1 NotesPump 结构

2.2 NotesPump 管理数据库

NotesPump 管理数据库是控制 NotesPump 的核心。其中存放着控制 NotesPump 服务器操作的要素。所有的要素（NotesPump 服务器的指令）都包含在这个数据库的文档中。

文档定义链接、作业以及系统和服务器配置。

NotesPump 支持从下列数据源传入或传出数据：DB2、EDA/SQL、文件、Notes、开放式数据连接(ODBC)、Oracle、Sybase 和文本。你可以在链接文档中定义 NotesPump 访问数据源的方法。链接提供给 NotesPump 访问数据源所需的所有信息。一个链接定义了从 NotesPump 服务器连接到指定的数据源利用什么样的参数，例如用户 ID 和口令。

行为文档包含的是 NotesPump 服务器和 NotesPump 中心元素的指令。在管理数据库中行为信息指示 NotesPump 服务器引擎去做什么，什么时候做，怎样做。NotesPump 能够执行下面这些行为去移动和管理数据：

- 命令行为，执行一个对某个数据库的动作；
 - 数据直接传输行为，从一个数据库中向另一个数据库中传输数据；
 - DPROPR 行为，允许从 IBM Data Propagator(DPROPR)的一致数据变换表到任何 NotesPump 数据源的信息传递；
 - Java 行为，将执行一个 Java 应用；
 - 轮流检测行为，它检测执行某个行为的条件是否满足；
 - Realtime Notes 行为，为应用程序提供对 NotesPump 支持的外部的数据源的同步访问；
 - 复制行为，通过用一个数据库中的数据来更新另一个数据库而使数据库保持同步
- 代码行为，可以执行 LotusScript 命令。

第 3 章 DB2

DB2 是一个关系型 DBMS，允许通过本地或远程的客户应用程序对数据库的数据进行存储、查询、更新、插入或删除操作。它使实现所有的数据库管理任务，例如设置、备份、恢复数据、管理目录、工作安排的媒介管理变得简便。

DB2 产品系列有：AS/400 系统、RS/6000 硬件、IBM 大型机、Hewlett-Packard 和 SUN 微系统公司的非 IBM 结构机器和诸如 OS/2、Windows(95 和 NT)、AIX、HP-UX、SINIX、SCO OpenServer、SUNSolaris、OS/400、OS/390 等作系统。

DB2 产品系列为商业智能、事务处理、各种类型的商业应用程序提供了开放式的、强有力的数据库管理。

3.1 结 构

在一个关系型数据库中，数据可存在于一个或多个的表中，每个表都包含一定数量的列和一定数量未排序的行；表中的每个列按某种关系与其他的列相联系。DB2 访问数据是依照它的内容而不是它存储的地点或组织结构；一张关系型表的行没有固定顺序，但列的顺序总是你定义表时所指定的顺序。

DB2 管理两类结构：

- 系统结构，由 DB2 控制和访问。它们含有关于所有对象的定义信息，例如数据库、表、视图和索引，以及有关用户访问对象权限的安全信息；
- 数据结构，根据用户的指示访问数据，数据按用户的要求组织。下面介绍数据结构。

数据库

一个数据库是许多 DB2 结构的集合。当定义数据库时，需要给出各表及其相关索引的名字，还要指出它们驻留的表空间。例如，一个数据库可以包含与一个应用程序相关的所有数据，这样就可以通过一个操作开始或停止访问数据。

表空间

一个表空间就是一个存储表的数据和系统结构的物理位置。

表

一个关系型数据库以表的集合的形式存储数据。表的数据逻辑上按行和列进行组织。表中的数据是相联系的，并且关联关系可在表间定义。表中的数据可在叫作关系的数学原则和操作的基础上进行查看和操纵。表数据的访问通过结构化的查询语言 SQL 进行。

创建一张表时，必须定义一个有序的列的集合；在每一个行和列的交叉处是值。在同一列的所有数据必须是一样的数据类型，这是在创建表时定义的。

一张表有一个主关键。主关键可以是一个列，也可以是多个列，其取值可以唯一地

标识每个行。

索引

一个索引是指向 DB2 表数据的指针集；通过指针生成一条到数据的直接路径，可有效的访问表中的每一行。索引与它指向的表独立存储，且基于表的一列或多列的数据值。

视图

视图是一种只显示数据但不进行维护的有效显示途径；它定义在某个表上，可包括表中的所有或部分的行和列。

一个视图不是一张实际的表并且不要求永久存储：创建一个视图时，DB2 在它的目录中存储定义但不为视图存储任何数据，因为数据已经存在于基础表中。它只是创建一个虚拟表供使用。

3.2 服 务

DB2 提供下列服务和特征。

结构化查询语言 SQL

SQL 是一种访问 DB2 表中数据的语言。SQL 包含数据定义语句和数据操纵语句。数据定义语句用来定义数据库的结构，数据操纵语句用来对表中的数据进行操作(SELECT、INSERT、UPDATE、DELETE)。SQL 还可以完成授权或废止对资源的访问的功能。

SQL 是非过程性的，因为你指定要做什么，而不是怎么做。要访问表中的数据，只需指定表名和数据所在的列，没有必要指定访问方法。

应用程序进程

所有的 SQL 程序都是作为一个应用程序进程的一个部分执行的。一个应用程序进程包含一个或多个程序的执行，并且是 DB2 分配资源和加锁的单元。不同的应用程序进程可包含不同的程序的执行，或同一程序的不同执行。如何开始和终止应用程序进程依赖于环境。

并行操作

多个应用程序进程可在同一时间请求对同一数据的访问。加锁被用来维持数据一致性，例如，阻止两个不同的应用程序进程同时更新同一行。DB2 获取锁来阻止一个应用程序进程未经提交的数据改变被另一个应用程序进程察觉；一个进程所获得的锁在进程结束时释放，对所有数据库进行永久改变的提交操作也会释放锁。

工作单元

工作单元是应用程序进程内的可恢复的操作系列。在任何时间，一个应用程序进程是单个工作单元，但一应用程序进程在执行过程中由于提交和回退操作会涉及许多工作单元。工作单元在应用程序进程初始化时开始或在先前的工作单元结束时（如果应用程序未结束）开始。工作单元在提交操作、回退操作或应用程序进程结束时结束。

COMMIT 可以结束工作单元，其间的数据库变化将变成永久的改变，可被其他用户

看见。ROLLBACK 是一个将数据的状态从由 SQL 语句对数据进行的改变恢复到上次进行操作前状态的进程。如果一工作单元在运行期间失败, DB2 将恢复所有未提交的数据。

COMMIT 或 ROLLBACK 操作只影响由该操作终止的工作单元期间的数据库变化。当这些变化未提交时, 其他的应用程序进程不能察觉, 所作改变可以回退。一旦提交以后, 数据库的变化可被其他应用程序进程访问, 不能再用 ROLLBACK 回退。

恢复

恢复是将发生故障的系统或系统中损坏的数据恢复到可操作状态的过程。

日志

DB2 在日志文件中记录所有的数据变化和重要事件。在发生故障的情况下, 与数据库备份一起, 用于恢复数据库的一致性, 使其恢复到错误发生前的状态。

3.3 DB2 编程

在关系型数据库中, 必须使用 SQL 访问需要的数据, 但如何将 SQL 集成进你的应用程序由你自己选择。可选择的方法有:

- 嵌入式 SQL
- 调用级接口 (Call Level Interface, CLI) 或 ODBC
- REXX
- 查询产品

嵌入式 SQL

嵌入 SQL 可由一个静态或动态的 SQL 组成, 或由二者混合组成。

如果应用程序运行时, 其中的 SQL 语句的条目和内容都是一成不变的, 就可以考虑在应用程序中嵌入静态 SQL。采用静态 SQL, 应用程序执行时可获得将应用程序绑定到数据库的用户特权。

如果 SQL 语句到应用程序执行时才能确定, 就可以使用嵌入式动态 SQL。使用嵌入式动态 SQL 可生成一个处理各种不同输入的通用应用程序。

包含嵌入式 SQL 的应用程序在编译之前, 需要用 DB2prep 命令作编程准备。另外, 使用 SQL 的应用程序必须绑定到数据库以利于程序运行 (使用 DB2bind 命令)。

调用级接口 (CLI)

CLI 通过 CLI 函数使你的应用程序能作数据访问。所有需要发出的 SQL 语句都通过一个 CLI 功能调用传递给数据库。使用 CLI 的优点包括消除了程序预编译和绑定, 由于 CLI 支持使用 ODBC 接口来访问数据, 增加了应用程序的可移植性。

用 CLI 写的应用程序只能使用动态 SQL。CLI 给处理过程增加了一些额外的负担。

表 3.1 对编写 DB2 应用程序时使用嵌入式 SQL 和 CLI 的优点进行了比较。

表 3.1 嵌入式 SQL 和 CLI 比较

嵌入式 SQL	CLI
静态、动态 SQL	动态 SQL
性能	可移植性
固定查询简单	无预编译、无绑定
支持多种编程语言	C 或 C++
SQLCA, SQLDA	没有 SQLCA 或 SQLDA
封装和安全	数据转换

REXX

REXX 支持动态 SQL 接口，不要求程序准备或把应用程序绑定在数据库上。使用这种解释语言可以加速应用程序开发。

查询产品

各种各样的查询产品——例如 Lotus Approach 可有效支持查询开发和报告。各种产品在如何开发 SQL 语句和采用何种逻辑层次上不同。

3.4 DB2 对象扩展

DB2 提供了若干特征，以允许对象关系能力：大对象支持、用户定义的不同类型、用户定义的函数和触发器等。在这些面向对象扩展之中的协同作用使你能够采用复杂的数据结构。

在这节我们简要介绍 DB2 面向对象的扩展。

3.4.1 用户自定义的数据类型

一个独特的数据类型是用户自定义的数据类型(UDT)，可将它内部的表示与已存在的类型(它的资源类型)共享，但它又被认为与大多数的操作是分离且不兼容的。例如，你可能想要定义一个年龄类型、一个重量类型和一个高度类型，它们都有不同的语义，但在内部表示时都使用内置的数据类型——整型 INTEGER。

下面的例子说明了如何生成一个独特的数据类型 PAY：

```
CREATE DISTINCT TYPE PAY AS DECIMAL (9, 2) WITH COMPARISONS
```

尽管 PAY 与内置的数据类型 DECIMAL (9, 2) 采用一样的表示，但还是被认为是一种不同的类型，与 DECIMAL (9, 2) 以及其他数据类型不兼容。它仅仅同其本身类型兼容。同样地，DECIMAL 所用的运算和函数也不能在这里使用。例如，PAY 数据类型的值不能与整数类型数据相乘。因此，你必须编写只适用于 PAY 数据类型的函数。

使用独特数据类型可以限制错误的发生。例如，如果 EMPLOYEE 表的 SALARY 列定义为 PAY 数据类型，尽管资源类型是一样的，它仍不能加入 COMM 中。

3.4.2 用户自定义函数

DB2 提供了内置的和用户自定义函数(UDF)。然而，这组函数不能满足所有要求。通常，你不得不为特别的任务创建定制的函数。用户自定义函数使你能够创建定制的函数。

有两种类型的用户自定义函数：源函数（sourced）和外部函数（external）。

- 源用户自定义函数使用用户自定义数据类型可以有选择地引用内置的或数据库已知的其他用户定义的函数。既可以使用标量函数，也可以使用列函数。

在下面的例子中，生成了一个用户自定义函数 MAX，它基于内置的 MAX 列函数，后者用 DECIMAL 数据类型作为输入类型。MAX UDF 采用 PAY 数据类型作为输入，并返回一个 PAY 类型数据作为输出。

```
CREATE FUNCTION MAX (PAY) RETURN PAY
SOURCE MAX (DECIMAL)
```

- 外部的用户自定义函数由用户用编程语言编写，有外部的标量函数和外部表函数。假设你编写了一个统计字符串中单词数的函数，便可以用 CREATE FUNCTION 语句把它注册到数据库中，命名为 WORDCOUNT。然后就可以在 SQL 语句中使用这个函数了。

例如，下面的语句返回雇员数和雇员履历 ASCII 表中的单词数。WORDCOUNT 是一个外部标量函数，用户已经把它登记在数据库中，现在可以在语句中使用：

```
SELECT EMPNO , WORDCOUNT (RESUME)
FROM EMP_RESUME
WHERE RESUME_FORMAT ='ascii'
```

3.4.3 二进制大型对象

支持大型对象使得多媒体数据，如文档、视象、图像和声音都可在数据库中存储并象其他数据库对象一样进行操作。

DB2 支持 3 种大型对象：

- 字符型大型对象(CLOB)

CLOB 是与一个代码页相联系的单字节字符串。当在客户机上使用另一代码页进行接收时，可以支持代码页变换。CLOB 适用于面向文本的信息，可以用来存储大型单字节字符集数据，例如文档。一个 CLOB 的最大长度可达 2 GB。

- 双字节字符大型对象(DCLOB)

一个 DCLOB 也是一个与代码页相联系的双字节字符串，支持代码页变换。DCLOB 适用于采用双字节字符的面向文本的信息。一个 DCLOB 最大长度可达 2GB (1, 073, 741, 823 双字节字符)。

- 二进制大型对象(BLOB)

BLOB 是一二进制的字符串，没有相关的代码页。BLOB 用于图像、声音、影象、图形和其他媒介的存储。一个 BLOB 最长可达 2 GB。BLOB 主要用作处理非传统型数据，如图形、声音和混合媒体。

3.5 分布式关系型数据库体系结构

分布式关系型数据库体系结构(DRDA)将 DB2 家族成员连接到分布式数据网络中。DRDA 是一开放式的体系结构，已向许多数据库供应商授权。它是分布式数据库的工业标准。通过 DRDA，终端用户和应用程序可以访问在数据网络中任何数据库的数据，

无需知道数据在哪儿, 没有传送中的完整性损失, 这一点对于关键性任务很重要。

DRDA 是一标准化的数据库协议, 提供异构客户/服务器环境下可靠、快速、安全的数据访问。DRDA 定义了命令、数据格式和规则, 允许任一 DRDA 的客户机都可成功地对任一 DRDA 服务器进行互操作。DB2 家族产品实现了 DRDA。开放组织 (Open Group) 也宣布接受 DRDA 作为数据库访问的互操作性的工业标准。

DRDA 允许在从工作站上用应用程序访问数据的同时, 又能够保持已有的安全、备份、统计和其他数据管理功能。

DRDA 是一套允许在不同平台上的 RDBMS 间透明连接的协议; 它协调系统间的通讯, 定义什么必须被交换和怎么交换。

DRDA 是客户/服务器计算模型的一种实现方式。它向网络任一处的客户提供数据服务, 但不关心他们的连接方法。分发处理的复杂性对客户是透明的, 应用程序不需要驻留在服务器上。

程序要利用驻留在不同硬件和操作系统平台上的数据, 需对平台间差异进行屏蔽。例如, 当大型机数据向 PS/2 传送时, 就需要在 EBCDIC 和 ASCII 数据间进行翻译。必须定义系统间的通讯链接, 必须保护数据的完整性, 发生故障时的恢复必须协调一致。必须知道两端的数据格式, 以便发送和接收系统能够理解被传送的数据。如果一个系统发出数据库请求, 另一系统必须理解该请求、执行该请求并向第一个系统返回结果。

期望用户和程序员考虑这些事情是不合理的。因此要求体系结构(由标准系统软件支持和实现)保证不同的系统能够理解对方。

DRDA 允许应用程序访问分布式关系数据, 允许在关系型 DBMS 之间通讯。

DRDA 提供在应用程序和分布式关系型 DBMS 间的连接服务:

- 描述信息在参加者之间的流向;
- 描述参加者的责任和信息流的用法;
- 提供分布式数据库管理所要求的格式和协议。

DRDA 不为分布式 DBMS 提供应用程序编程接口(API), API 由 SQL 提供。

3.5.1 DRDA 数据库的角色

在一个 DRDA 环境中每个数据库平台同时为不同的应用程序扮演不同的角色。

• 客户

数据库客户是运行应用程序的部件, 发出 SQL 语句请求数据。典型的数据库客户程序是 DB2 CAE。

• 应用程序请求者 AR

应用程序请求者 (Application requester, AR) 是处理分布式连接的应用程序端的代码; 是请求数据或服务的应用程序。在工作站环境中, DB2 连结产品代表应用程序完成 AR 的功能。由于 DRDA 不要求任何本地数据库函数, DB2 连接可在没有本地数据库支持的情况下安装。

• 应用程序服务器 AS

应用程序服务器 (Application server, AS) 是处理连接的数据库端的代码。它为客户执行数据库操作, 并向发出请求的客户返回处理结果。AS 代码是 DBMS 的一部分, 请

求的数据在其上驻留。

3.5.2 分布的程度

DRDA 描述了三级分布:

- 远程工作单元(RUW)

RUW 在单一的远程数据库上按事务的形式处理相关的 SQL 语句。每个 RUW 都能包含多条的 SQL 语句,但这些语句必须访问同一 DBMS。一个应用程序可以更新许多远程数据库,但是在一个工作单元内它只能访问一个数据库。当应用程序准备好提交工作时,提交在执行工作单元的 DBMS 处开始。

- 分布式工作单元(DUW)

分布式工作单元 DUW 由经过协调的提交和恢复机制组成,处理跨越多个远程数据库的事务中的相关 SQL 语句。一个应用程序可在一个工作单元内访问多个数据库服务器;提交数据前,它能在不同的数据库服务器间切换。

提交和回退在数据库服务器处协调,那样,如果系统某处发生故障,仍可以保护数据的完整性。

- 数据库指导 DUW

在工作单元内,应用程序连接到一 DBMS,该 DBMS 可在本地执行一些请求并选择路由把其他请求发送到另外的 DBMS。DBMS 确定哪个系统管理被 SQL 语句所引用的数据,并且自动地将请求引向那个系统。在同一工作单元内的 SQL 语句可分别引到多个 DBMS,但是一条 SQL 语句中的对象必须是关于同一 DBMS 的。

3.6 DB2 家族

规划实现数据管理技术要认真考虑许多因素,可靠性、功能性和功能的可扩展性、易管理性、可伸缩性和性能是评估软硬件平台时要考虑的关键点。一个解决方案的最重要的因素是关系型数据库服务器。利用 DB2 家族,IBM 提供了目前最可靠、可扩展、易管理、可伸缩和高性能的工业分布式数据库解决方案。IBM 公司的关系型数据库管理系统 DB2 家族产品包括广泛的硬件平台:

- 基于 Intel 的 PC 机和服务器,运行 OS/2, SCO OpenServer, 或 Windows NT 操作系统;
- RISC 机器,包括对称的多处理机 (SMP)、处理机集群、大规模并行处理机(MPP),运行 AIX, HP-UX, Sun Solaris 或 SINIX;
- AS/400 中型系统;
- S/390 大型机,运行 OS/390、MVS、VSE、或 VM 操作系统。

除了每种平台上的数据库服务器引擎,DB2 家族还包括为不同种类的数据库访问、数据复制和数据库系统管理提供中间件解决方案。普通客户在流行的桌面平台上用开放的工业标准 API 可以访问所有这些分布式数据库管理服务。DB2 不仅仅是一个关系型数据库产品;它更是今天的网络计算环境中一个完全的分布式数据库解决方案。

3.6.1 服务器

OS/390 操作系统下的 DB2 版本 5 是这个家族的旗舰。自从 1982 年问世以来,它为工业界提供了一个关系型数据库标准,并且在 90 年代继续承担这样的角色,这可由它的新近改进和持续市场增长来证明。在今天的工业界中,没有哪一个关系型数据库产品象 DB2 这样管理那么多数据或支持更多关键任务的应用程序。没有哪个关系型的数据库产品能与它的可靠性、可管理性或性能特征相媲美。DB2 的研究、技术和成功的实际市场经验奠定了 IBM 全方位的分布式数据库产品家族的基础。

DB2 通用数据库(UDB) 5.0 版本是为 UNIX 和包括 AIX、HP-UX、Solaris、OS/2、Windows NT 的 Intel 服务器提供的 IBM 关系对象数据库。一个更早的版本, DB2 通用服务器 V2.1, 可用于 Siemens-Nixdorf SINIX 和 SCO OpenServer。

DB2 UDB 可以在任何平台上运行,从支持移动用户的膝上型电脑,到拥有数兆字节数据和/或几千用户的并行处理机。DB2 UDB 允许你扩充数据库能力满足特定的组织要求,支持更先进的应用程序对包含文件、图像、声音和影象的多媒体数据进行处理。DB2 UDB 与互联网技术紧密结合,可从 Internet 或 Intranet 上采用安全机制方便地访问数据。DB2 UDB 有力地支持了商业智能应用程序,例如数据仓库和联机分析处理(OLAP)。DB2 是工业并行数据库技术和最优化查询的领袖。DB2 UDB 是工业界中最容易建立、使用和管理的数据仓库之一。它包括一全套图形管理工具,使安装、管理和远程操作简便易行。DB2 UDB 能从几乎任何网络的任何客户工作站上访问。

OS/400 系统 DB2 (DB2/400) 是 OS/400 操作系统的一个集成组件。它是世界上应用最广泛的多用户关系型数据库管理系统,演变为具有高度鲁棒性,并与其他 DB2 家族成员高度兼容的产品。它能作为一个局域网(LAN)上使用客户机/服务器协议的数据库服务器,也可作为广域网上的主机数据库。

VSE & VM 系统 DB2 版本 5 (以前叫做 SQL/DS) 为中等 S/390 环境提供了鲁棒性的关系型数据库管理。支持高性能特色,如 VM/ESA 数据空间和 VSE/ESA 虚拟硬盘,同时可与 DB2 家族成员完全兼容。

DB2 DataJoiner 为先进的信息集成能力奠定了基础。它在异种环境的数据复制、数据仓库和对对象/关系型数据管理能力方面都有特色。DB2 DataJoiner 提供了无需考虑数据的位置、格式的数据处理机制,按照透明、有效和连续的方式来集成关系型或非关系型数据。支持基于 Web 的应用程序、大范围的数据源、工业标准 SQL、唯一查询优化技术和同步、异步的数据访问就是该产品众多特色的一些例子。由于可用于 UNIX (AIX 和 HP-UX) 和 Windows NT, DB2 DataJoiner 为各公司最大限度地挖掘他们的数据的价值提供了强有力的工具。

3.6.2 DB2 CAE

DB2 CAE (Client Application Enabler) 提供一个通用的应用程序和用户接口,接入由 DB2 家族或由 DRDA 兼容的非 IBM 数据库管理的数据。它支持包括 ISO ANSI SQL、X/Open CLI 和 Microsoft ODBC 在内的工业标准接口,支持大量的 IBM 或非 IBM 的应用程序。

对于运行需要连接到 DB2 的应用程序的工作站,每一台都必须安装 CAE。从 CAE

可以:

- 与一台 DB2 UDB 服务器或 DB2 连结网关机器通信;
- 从一台远端客户机上发出交互式 SQL 语句访问 DB2 服务器上的数据;
- 利用 UDB 服务器的图形化管理和监视;
- 运行符合 ODBC 标准的应用程序;
- 运行通过 Java 数据库连接 (Java Database Connectivity, JDBC) 访问 UDB 数据库数据的 Java 应用程序。

OS/2、Windows NT、Windows 95、Windows 3.x、AIX、HP-UX 和 Solaris 都支持 CAE。

3.6.3 DB2 软件开发工具包

DB2 软件开发工具包 (SDK) 含有客户、应用程序开发工具和所有支持平台下的示例程序。DB2 软件开发工具包的不同部分可单独订购。DB2 SDK 使你能够使用嵌入式 SQL、CLI、ODBC 以及通过 JDBC 访问数据库的 Java 来开发应用程序。

3.6.4 DB2 连接器

DB2 连接器作为 DB2 产品家族的一个成员, 是 IBM 的分布式数据库连接服务 (DDCS) 的替代品。DB2 连接器提供了到 IBM 大型机数据库的快速、鲁棒性连接, 连接用于 e-business 和其他运行于下面 Intel 和 UNIX 操作系统上的应用程序:

- AIX
- HP-UX
- Microsoft Windows
- OS/2
- Sun Solaris

使用的 DB2 连接器, 客户程序可透明地访问下列 IBM 数据库产品的数据, 它们以 DRDA 应用程序服务器形式运行:

- OS/400 DB2
- OS/390 DB2
- DB2 UDB
- VSE & VM DB2

到主机数据库的连接是 IBM 的 APPC (Advanced Peer-to-Peer Connection) 网络下的 SNA (Systems Network Architecture) 连接。OS/390 DB2 也支持在 TCP/IP 上的数据库连接。

不考虑使用的协议, DB2 连接器为具有两阶段提交处理的 DUW 提供 DRDA 支持。

3.6.5 包装

DB2 UDB 有三种包装选择, 每种都有一套组件:

- DB2 个人版

DB2 个人版包括单机使用的 DB2 UDB。它包含为数据库系统提供的函数和 DB2 UDB 工具, 但是不能为局域网中的客户机充当服务器。

- DB2 工作组版

DB2 工作组版包括在局域网环境中用作服务器的 DB2 UDB。个人版的所有的函数和工具在工作组版中也有。它还包含提供因特网访问的 DB2 Net.Data 和所有被支持平台上的 DB2 CAE。

- DB2 企业版

DB2 企业版包含工作组版所有的组件和供用 DRDA 访问主机数据库的 DB2 连接器产品。

如果要用 DB2 UDB 开发应用程序，则使用 DB2 SDK。

用访问 DRDA 数据库的 DB2 连接器有两个版本：

- DB2 连接器个人版

DB2 连接器个人版为 Intel 工作站上的单个用户所设计，允许单独的工作站访问主机系统。

- DB2 连接器企业版

DB2 连接器企业版执行网关的功能，支持工作组、部门和局域网中的 Intel 和 UNIX 工作站，允许多客户机和主机系统之间的连接。