

# Oracle 数据库 10g 的 OLAP 选项： 唯一用于商业智能的关系 - 多维数据库

*Oracle 白皮书*  
2004 年 6 月

# Oracle OLAP 10g: 唯一用于商业智能的关系 – 多维数据库

执行概要.....	3
引言.....	3
目前的商业智能市场.....	3
商业智能应用程序的类型.....	3
传统分析服务器.....	4
Oracle 关系 – 多维数据库.....	5
集成 RDBMS-MDSS 的优势.....	6
简化的管理.....	6
高可用性.....	6
提高的安全性.....	7
数据等待时间最短.....	7
开放访问.....	7
Oracle OLAP 10g 概述.....	7
开放接口.....	8
集成数据库过程.....	8
集成的 RDBMS-MDSS 数据存储.....	8
多维数据模型.....	9
OLAP DML.....	10
假设分析.....	13
查询 API.....	14

## 执行概要

大多数企业中，商业智能应用程序的部署是沿着技术线路划分的。不太复杂的报表和即席查询工具的部署通常使用的是数据仓库和基于 SQL 的报表工具，而较复杂的分析和计划应用程序的部署则是使用其他特殊的数据仓库和工具。

从传统角度看，对不同类的应用程序使用不同的技术是有理由的。关系数据仓库易于扩展，可支持非常大的数据集。单独的分析数据库提供了关系数据库所没有的分析功能。

数据仓库和分析数据库间的这种划分带来了高昂的代价。维护单独的分析数据库需要在硬件、软件和数据库管理员方面付出额外的开销。将数据复制分析数据库的过程也使数据对分析员和决策人员的可用被大大推迟。

随着数据集规模的增加，这些问题也变得更加尖锐。需要更多的时间将数据复制到分析数据库中。通常，单独的分析数据库难以应付非常大的数据库。这些问题转化成了与更及时、更有效的分析所相关的机会成本。

Oracle 数据库 10g 通过提供支持分析的 Oracle 数据库消除了这些障碍。集成在 Oracle 数据库 10g 中的 OLAP 选项提供了特殊分析数据库的分析功能，消除了管理一个单独数据库系统的需要。

## 引言

唯一用于商业智能的关系 – 多维数据库意味着什么？这不是简单地在关系数据库上包装和安装传统的分析服务器。这也不是两个不同数据库服务器间的协作或相互依赖。这是在同一数据库实例中组合关系数据类型和多维数据类型。最终在 Oracle 数据库的核心中引进一个全功能的多维引擎。

从 Oracle9i 数据库 Release 2 以来，Oracle 数据库的 OLAP 选项是目前唯一可用的关系 – 多维数据库。多维技术和关系技术共存在同一平台上。多维数据是 Oracle 数据库中最好的数据类型。还没有其他的供应商做出这一声明。

## 目前的商业智能市场

在进一步讨论 Oracle OLAP 10g 前，有必要检阅一下分析服务器的市场。本部分讨论四种类型的商业智能应用程序和当前用于支撑这些应用程序的架构。

### 商业智能应用程序的类型

企业需要各种不同的商业智能应用程序以满足不同的用户群。这些应用程序通常可分为四类：

标准报表  
即席查询和报表  
分析报表  
计划应用程序

Oracle OLAP 10g 满足各种类型应用程序的要求。

#### **标准报表**

报表应用程序通常提供静态或参数化的报表。受众通常很广，例如，一个企业中所有的销售经理。具有最低分析要求的报表应用程序通常是基于关系数据库的，并使用 SQL 作为查询语言。

#### **即席查询和报表**

即席查询和报表应用程序通过各种数据选择和导航技术的使用为用户提供了高层次的交互。即席查询和报表应用程序通常是基于关系数据库的，使用 SQL 支持“一维”查询，例如“我的前 10 个客户是谁”。对大多数的用户群，例如销售人员来说，这种程度的分析功能已经足够。

#### **分析报表**

分析报表应用程序也支持数据的即席查询，但它们也回答更复杂的问题。虽然下面的查询看起来有些复杂，但这是营销分析员常问的问题：

*与一年前的同期相比，销售总额的百分比变化是多少？对最前面的 10 个客户中每个客户的前 10 种产品进行分析，以本年度对每个客户的销售总额与去年同期的总额的百分比变化为基准。*

该查询是多维的。查询中的计算是在多个方向上进行的。同期前和本年度至今日的计算是在时间维上。排列则是在客户和产品维上。产品排列嵌套在客户排列中。产品和客户排列中嵌入了一个计算后的尺度。

传统上，大多数的即席分析应用程序使用了独立的分析数据库来支持所要求的复杂多维查询。

#### **计划应用程序**

计划应用程序可使用户预知结果。它们允许用户提出如下的问题：

*本公司本季度的利润将是多少？*

*价格变动将如何影响单位产品销售额和净利润？*

*货币波动对净利有什么影响？*

计划应用程序非常不同于查询和报表应用程序，因为它们使用模型、预测、特殊集合和分配方法等分析工具生成新的数据。计划应用程序的常见例子包括企业预算和财务分析，以及需求计划系统。通常这些应用程序使用了独立的分析数据库。

#### **传统分析服务器**

在过去，分析应用程序的开发者必须做一个本质决定：应用程序应采用关系数据库访问数据仓库中的数据，还是使用特殊的分析数据库？平衡是非常重要的。

关系数据库提供了成本效益最高的数据管理方法，对各种应用程序提供了

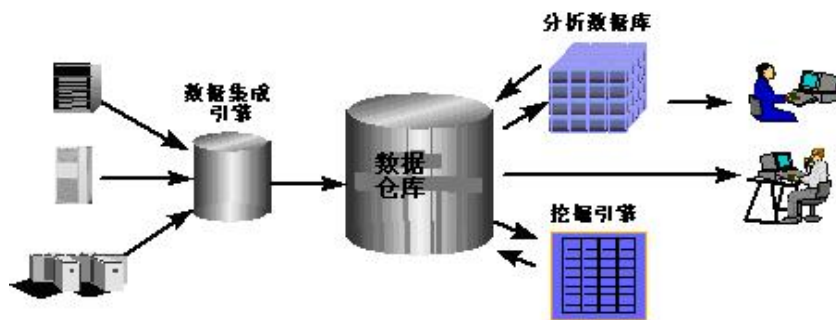
最开放的访问。被管理的数据集中在一个中心位置。不需要将数据复制到其他特殊的数据库。由于所有的数据都保存在数据仓库中并能使用 SQL 查询，基于 SQL 的应用程序都可以访问数据。不幸的是，SQL 的分析功能非常有限，而且性能比不上特殊的分析数据库。

特殊的分析数据库提供了完整的分析功能集，并为复杂的查询提供了更好的性能，但分析数据库的维护需要巨大的开销。分析数据库要求数据复制和独立的管理设施。数据复制是个高代价的过程，将造成数据可用性的重大延迟。独立的管理过程同样是昂贵的，因为它需要单独的数据建模、ETL 过程、安全步骤和灾难恢复计划。

因此，开发者倾向于使用 SQL 编写仅具有有限分析要求的应用程序，并对有更多分析要求的应用程序使用分析数据库。企业需要承担管理独立数据库系统和复制数据的费用。

关系 OLAP (ROLAP) 技术为一些使用 SQL 的分析计算提供了支持，但由于分析能力和性能的限制，这些技术只是取得了有限的成功。

下面的图片显示了一个用于支持各种分析应用程序典型数据库架构。



这个典型环境中，数据由源系统（例如总帐或销售定单系统）访问，经数据集成引擎处理后被装载到关系数据库的一个数据仓库中。即席查询和报表应用程序可直接查询数据仓库并提供对数据的及时访问。但是，基于特殊分析数据库的分析应用程序用户需要等上数小时乃至数天，等待数据被复制到分析数据库中。

从 DBA 的角度看：

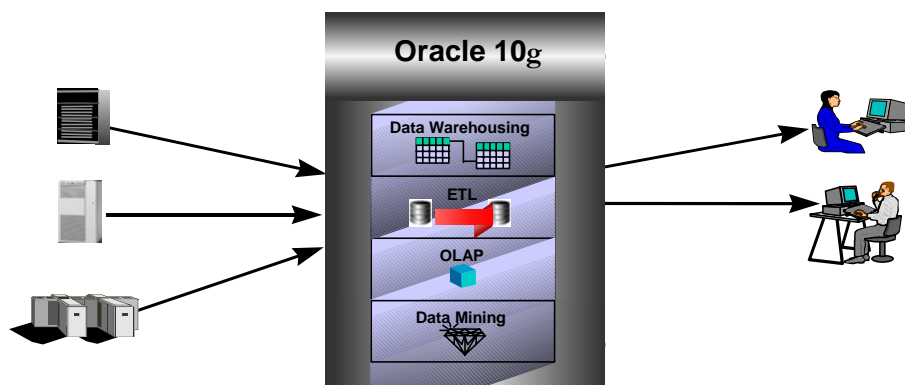
- 有三个技术要掌握（数据集成工具、关系数据库和分析数据库）
- 有三个元数据库要管理
- 有三个管理过程
- 有两个数据存储（仓库和多维）

无论是在管理系统方面，还是由于无法访问最新数据而导致的低效这一机会成本方面，管理这三个过程的代价都是非常高昂的。

### ORACLE 关系 – 多维数据库

Oracle 数据库 10g 通过提供集成的关系 – 多维数据库挑战对分析数据库的传统认识。Oracle RDBMS-MDDS 消除了可管理性和分析能力及性能之间的矛盾。Oracle 数据库 10g 简化了过程，降低了维护数据的费用，同时能够支持复杂的分析查询并提供出色的性能。Oracle 数据库 10g 扩展

了 SQL 的分析能力并以关系数据库集成组件的形式提供 Oracle OLAP 10g，从而实现了这一目标。



此外，Oracle OLAP 10g 是集成数据仓库和商业平台的一部分，该商业平台将 ETL、OLAP 和数据挖掘功能集成进了 Oracle 数据库。结果就是一个更易于管理和成本效率更高的平台，它全面支持分析应用程序并提供对更新信息的及时访问。

### 集成 RDBMS-MDDS 的优势

与分离、独立的多维数据库相比，完全集成的 RDBMS-MDDS 具有很多优势。这些组件包括：

- 简化的管理
- 高可用性
- 提高的安全性
- 数据等待时间最短
- 开放访问

### 简化的管理

Oracle RDBMS-MDDS 将所有的管理任务整合到单一的数据库中，从而简化了管理。初始的安装设置只是简单地以数据仓库数据库配置安装企业版。随后，只需要管理一个过程、一个数据存储和一个元数据库。只有一套数据库工具和数据库管理包需要掌握。

传统的 OLAP 技术要求为关系数据库和多维数据库管理单独的过程和元数据库。每个管理任务也需要单独的管理工具以及更多的时间和更多的人员。换句话说，它们的管理成本更高。

### 高可用性

数据库和互联网通过将数据库应用程序范围扩展到整个企业和组织，实现了全世界的协作和信息共享。范围中扩大数据库应用程序的范围。高可用性对企业商业智能平台的成功实施至关重要。Oracle OLAP 10g 是 Oracle 数据库的一部分，因此它能全天 24 小时提供一致的数据访问。相比较而言，独立的 OLAP 服务器则通常缺少 Oracle 这种对计算机故障和数据损坏，例如存储故障、人为错误、数据出错和站点失效的支持。

## 提高的安全性

坦白地说，攻破 Oracle 数据库是不可思议的。没有授权的用户是无法访问 Oracle 数据库的。数据库中的所有数据，包括 OLAP 数据，都得到了单一安全策略的保护。所有的用户都被定义在单一的用户目录中，通过标准的 Oracle 安全功能，例如 GRANT 和 PRIVILEGE，分配权限。这些用户处于数据库本地，并且紧密结合了其他的安全功能，例如 Oracle 标签安全性、虚拟专用数据库和细粒度审计。

而传统的 OLAP 服务器则需要手动同步用户目录和数据访问权限或者寻求中心用户目录的服务。

## 数据等待时间最短

在数据仓库环境中，如果使用 SQL 应用程序或基于 OLAP 服务器的应用程序访问数据，那么必须将仓库数据复制到多维数据库中。这导致了一个包含两个步骤的数据库维护过程。首先，更新数据仓库。其次，更新多维数据库。这个两步骤的过程不仅耗费时间，而且耗费计算资源。最重要的是从源系统数据可用到数据可供用户分析之间的时间大大增加了。

由于 Oracle 数据库 10g 允许将数据保存在关系表或多维表中并对两种表均提供 SQL 和 OLAP API 访问，所以不再需要将数据同时复制到关系表和多维表中。Oracle 数据库 10g 只需要一个单步的 ETL 过程，这减少了等待数据可用所需的时间和计算资源。

## 开放访问

从 Oracle9i 数据库 Release 2 开始，Oracle 就是唯一能够对关系和多维数据同时提供 SQL 和 OLAP API 访问的数据库。应用程序开发者可选择使用 OLAP API 的计算和多维数据导航功能，或使用标准的 SQL 访问多维数据。这使企业能够将在基于 SQL 的工具上的投资平衡到分析报表空间。报表生成程序可被用于表示复杂多维计算的结果。即席查询工具使用户能够利用新的计算功能研究数据。任何 OLAP 计算都可通过 SQL 进行查询。

较旧的技术需要使用私有的 OLAP API 访问多维数据库，从而导致无法使用基于 SQL 的应用程序查询多维数据库中的数据。

## Oracle OLAP 10g 概述

Oracle RDBMS-MDDS 10g 包含了三个技术：

- 关系技术管理关系数据库对象并为数据提供 SQL 接口。

- 对象技术使数据库可以处理非关系数据。

- OLAP 技术提供了多维模型环境下的高端分析功能。

关系和 OLAP 技术均包含如下层次：

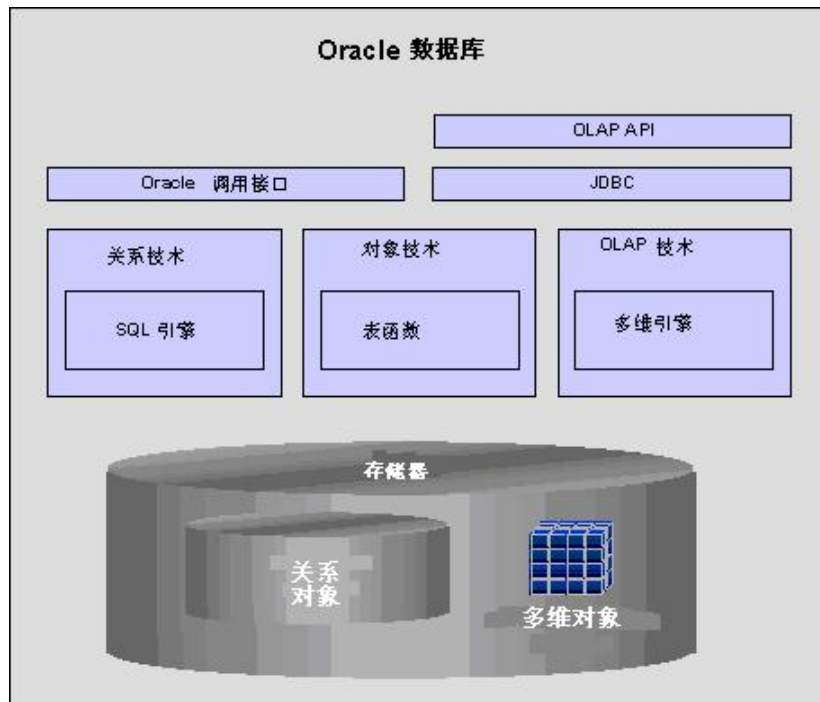
- 接口

- 处理引擎

- 数据存储

对象技术提供了以关系数据类型表示非关系数据类型的功能。在 Oracle OLAP 10g 中，对象技术为多维数据提供了 SQL 接口。

Oracle 数据库 10g 包括关系、对象和 OLAP 技术，如图所示。



### 开放接口

OLAP API 为数据库提供了一个 OLAP 接口，通过 JDBC 连接。基于 SQL 的客户端可选择通过 Oracle 调用接口 (OCI) 或 JDBC 连接数据库。OLAP API 使用 SQL 查询数据库，可访问关系和多维表。SQL 可通过表函数或关系视图访问多维表。利用 PL/SQL，SQL 应用程序也可以直接与多维引擎进行交互。

这些核心技术由 Oracle 企业管理器数据库控制、分析工作空间管理器和 Oracle Warehouse Builder 所提供的管理架构进行支持。Oracle 商业智能工具，例如 Reports、Discoverer Plus、OLAP Spreadsheet Add-In 和 Business Intelligence Beans，利用集成数据库提供了可用于查询数据库和分析数据的工具。

### 集成数据库过程

所有的技术 — 关系、对象和 OLAP — 都包含在一个单一的集成过程中，数据库以单一、集成化的实例进行管理。内部模块间的所有通信都通过可信呼唤进行，使所有的数据交换都能通过最低层次的数据管道进行。因此，与必须通过网络进行通信并且使用额外的软件层管理安全策略的分离过程相比，单一过程提高了性能。

因为 OLAP 技术是单一数据库过程的一部分，所以它同样享有 Oracle 数据库的可伸缩性和可靠性。Real Application Clusters 和企业级网络计算等功能受到自然地支持。

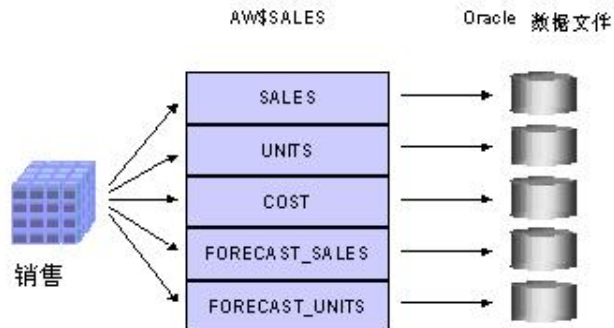
### 集成的 RDBMS-MDDS 数据存储

Oracle 数据库 10g 中，所有数据 — 关系和多维 — 都保存在 Oracle 数据文件中。不存在单独的多维文件。关系和多维数据类型共存在相同的数据文件中，即标准的 Oracle 数据文件。

保存在数据库中的多维数据被称为 *分析工作空间*。分析工作空间中含有很多的多维对象，可被认为是多维模式。在单一的数据库中，可创建并在用户间共享多个分析工作空间。与关系模式一样，某个特定的用户 ID 拥有一个分析工作空间，其他用户可被赋予访问的权限。

分析工作空间可作为二进制的大型对象 (BLOB) 存储在关系 AW\$ 表中。在 AW\$ 表中，通过指定行的最大数据量，可将分析工作空间划分成多个行。对大型分析工作空间的支持需要该功能，因为 BLOB 数据类型的每一行都有大小限制。AW\$ 表自身可进一步进行分割。

除段大小划分外，AW\$ 表还可将变量放进指定的行。例如，AW\$SALES 表可按下图所示进行逻辑划分。



好处就在于数据库管理员可完全控制数据在数据文件间的分发，从而最大化 I/O 性能。此外，在多处理器服务器上可获得更大的提高，因为 OLAP DML 的 UPDATE 和 AGGREGATE 命令可充分利用并行化。

### 多维数据模型

Oracle OLAP 10g 是基于多维数据模型的，该模型可对复杂的查询提供快速的回答。它注意到了分析人员在交互会话中，而不是通宵运行的成批作业中会提出重复问题的事实。多维数据模型是设计用于实时解决复杂查询的。Oracle OLAP 10g 通过 OLAP API 和 OLAP DML 表达该逻辑模型，OLAP DML 是一种多维编程语言。

### 逻辑模型

多位模型为分析应用程序的最终用户提供了众多重要的好处。最重要的好处是：

容易为大多数商业用户所理解。例如，大多数用户都会发现销售应客户、产品和时间段的不同而不同是非常直观的，同样直观的是在不同的角度上有着不同级别的汇总。

为数据选择过程提供上下文。模型使应用程序可根据级别（以月为级别的时间）、属性（存储经理为“Joe Smith”）或层次关系（“品牌 X”的后代）选择数据。

大大简化了复杂计算的定义过程，即使是非程序员也可使用计算语言。例如，“逐年销售的增长”可通过“sales -lag(sales 1 year)”计算。这与电子表格并无太大的不同。

通过 OLAP API 和 OLAP DML 展现的多维逻辑模型含有如下元素：

多维数据集

度量  
维度  
层次结构  
级别  
属性

*多维数据集*提供了组织具有相同形式，或者说相同维度的度量的方法。

*度量*用事实数据填充多维数据集中的单元。销售价值和销售单位就是例子。度量按照维度进行组织，维度通常包括一个时间维。

*维度*为事实数据提供了结构。它们形成多维数据集的边缘。这样的例子如产品、地理位置、渠道和时间。

*层次结构*使用不同的聚合级别组织数据。例如，在时间维上，可使用层次结构从月度、季度到年度的各种级别聚合数据。

*级别*代表着层次结构中的位置。

*属性*是维度成员的描述性特征，提供了额外的信息。例如，产品纬度可将库存单元 (SKU) 用作维度成员。由于 SKU 对大多数人是无意义的，因此您可为描述性标签定义属性。

#### **分析工作空间中的物理模型**

因为分析工作空间存储数据，所以需要使用物理模型组织数据。Oracle OLAP 10g 使用一个多维模型作为分析工作空间中物理模型的基础。该模型提供了数个重要的好处：

强制参考完整性。模型不允许度量数据包含非维度成员的键值。

数据一致性。模型鼓励多度量维度的重用。

保持的成员顺序。每个维度都有一个默认状态，该状态是按存储顺序排列的成员列表。除非被故意改变，否则默认状态列表始终不变。这简化了时间序列分析等计算。当维度成员的顺序保持不变时，提前与延期支付等函数可利用维度成员的位置，从而简化计算的定义并减少对纬度属性的需求。

完全预计算。无论数据是被预解决，还是在运行时进行计算，分析工作空间总会表现为已解决。应用程序查询分析工作空间时无需指定汇总规则或联接。SQL 应用程序无需关心如何计算度量，它们只需选择数据。

定制成员和度量。计算后的维度成员和度量得到透明的评估。模型支持会话级别的维度成员和度量，根据计算规则这些维度成员和度量可得到透明的评估。其他计算可引用这些计算后的成员和度量，就想存储的数据成员和度量。

#### **OLAP DML**

OLAP DML 是用于创建分析工作空间、定义数据容器和处理容器中所存储数据的数据定义和处理语言。它是特别设计用于支持多维计算的。虽然语言非常不同，但可将 OLAP DML 视作多维版的 PL/SQL。

OLAP DML 集成了大量的分析函数，可用于产生任何类型的多维计算。由于它是一种过程语言，所以计算可包含条件逻辑。这使计算规则可根据数据值的不同而不同。例如，应用程序可对小于 100,000 的销售值运用一

个计算规则，对大于 100,000 的销售值运用另一个计算规则。

#### **OLAP DML 函数的类型**

- 汇总
- 分配/n 分摊
- 数据选择
- 财务
- 预测和回归
- 数学和统计
- 模型
- 定制维度成员

#### **汇总**

OLAP DML 支持各种汇总方法，包括取最大、取最小、平均、加权平均和求和。汇总方法可应维度的不同而不同，并且提供对存储和运行时汇总的支持。对于什么数据可预先解决，什么数据可在运行时进行计算，可进行非常细化的定义。

管理员可根据级别、单个成员、成员属性、时间范围或数据的值有选择地实体化数据。例如，对于一个所含销售数据以产品、客户和时间为维度的多维数据集，可采用只预汇总最近时间段、最大客户和销量最大产品的数据的汇总策略。

要注意的是，任何未得到预汇总的数据都会在运行时得到自动和透明的汇总。

#### **分配**

OLAP DML 含有一个非常复杂的分配系统。系统中关键的功能包括：

- 支持层次结构和非层次结构的分配。层次结构的分配根据层次结构内的亲子关系分发数据。例如，季度级别的数据可能被分配给月。

- 非层次结构的分配允许对维度中的任意成员集的分配。例如，分配定额的 40% 给北京，60% 给上海。

- 各种分配方法包括复制方法（层次结构复制、最小、最大、第一、最后）、平均分配方法（平均、层次平均）和比例（包括加权分配）。例如，使用销售预测系统的用户可选择根据上一年薪水和年销售量的百分比变化为不同的销售办公室分配下一年的薪水。

- 单元级锁定可使应用程序避免单元遭到分配系统的覆盖。这在预算应用程序中是通行的做法，这些应用程序中一些用于计划时间段的数据是众所周知的。

#### **数据选择**

OLAP DML 提供的数据选择方法经过了特殊设计，适用于多维模型。这里包括基于层次关系、汇总级别、属性、时间序列函数和数据值的方法。

例子有：

- 选择所有月度和季度级别的时间段

- 选择 UPC 代码级别上所有是 X 品牌后代的产品

- 选择所有大尺寸的蓝色产品

选择所有销售年增长超过 20% 的客户  
与 SQL 不一样，分析工作空间中的数据选择在整个会话中都是一致的，从而提供了利用增量选择标准在数据间导航的方法。例如，您可以采用如下的多步骤选择：

1. 在品牌级别选择所有的产品
2. 保留大尺寸的产品
3. 保持销售量最大的 10 个产品

### **财务功能**

具有类似电子表格的财务功能，例如利率计算、折旧和付款计划。

### **预测和回归功能**

OLAP DML 支持非常复杂的预测和回归功能，包括非线性回归方法、单指数平滑法、双指数平滑法和 Holt/Winters。所提供的很多选项允许应用程序对预测方法进行细调。

### **数学和统计功能**

分类、相关和偏差等数学和统计功能使应用程序可以解决任何类型的数学方程并执行统计分析。

### **模型**

模型根据单个维度成员自己唯一的计算规则计算它们的值。表达式将被自动排列，也可使用联立方程。模型是分析工作空间中的对象（也就是说，它们以元数据保存），经常与财务应用程序一起使用。

### **定制维度成员**

OLAP 选项支持在维度中永久性地或运行时定义定制维度成员。一旦在维度中得到定义，定制成员就能在任何计算中得到透明使用，就和存储的维度成员一样。

例如，应用程序可允许用户定义新的维度成员‘North America’：

```
'North America' = 'Mexico' + 'United States' + 'Canada'
```

定制成员也可以是做减法的。可创建计算 California less Los Angeles 的成员：

```
'California less LA' = 'California' - 'Los Angeles'
```

### **假设分析**

对 OLAP 计算引擎非常关键的是通过会话级更新、会话级提交、导出会话数据到私有持续存储和合并数据到数据库全局视图等功能支持假设分析的能力。

会话级更新使用户可以在不影响数据全局视图的情况下更改数据。例如，一个分析定价变化效果的应用程序可允许用户输入新的价格数据，根据这些新的价格建立期望销售量和利润率的模型，然后根据新的销售量和利润率进行利润预测。这可在用户会话内完成，无需将变动提交给数据库的全局视图。

会话中用户也可能需要添加新对象到数据库中。例如，用户可能需要创建一个新的计算，度量基线预测和使用新定价模型的预测之间销售单位的方差。用户在会话中可为此定义一个新的公式。该公式仅在用户会话内可视。

并非所有的假设分析都能在单一用户会话内完成的。分析可能需要跨越多个会话或涉及其他用户。这种情况下，OLAP 选项可将会话数据和 DDL 保存在一个持久数据存储中，该存储为用户专用或在一个工作组内共享。稍后该会话数据可作为会话数据或对数据库的永久更新被合并到数据库的全局视图中。

### 多输入模式

多维引擎支持一种多输入附件模式下的假设分析，这种模式下多个会话可同时修改一个分析工作空间。在多输入模式下，用户可通过为单个变量、关系、值集和维度指定附件模式（只读或读写），在受控的方式下同时修改相同的分析工作空间。MULTI 附件模式提供了分析工作空间内并行任意数目活动的机会。

### 查询 API

Oracle 数据库 10g 同时支持 SQL 和 OLAP API 对关系数据表和分析工作空间的查询。对分析工作空间提供 SQL 支持使得基于 SQL 的工具可以访问多维数据和计算。OLAP API 提供额外的功能，例如多维导航（例如追溯、主元和多维游标）和用 Java 定义计算的能力。

客户端应用程序访问 Oracle OLAP 数据的方法有四种：

- Java OLAP API

- SQL 通过关系视图

- SQL 通过 OLAP\_TABLE

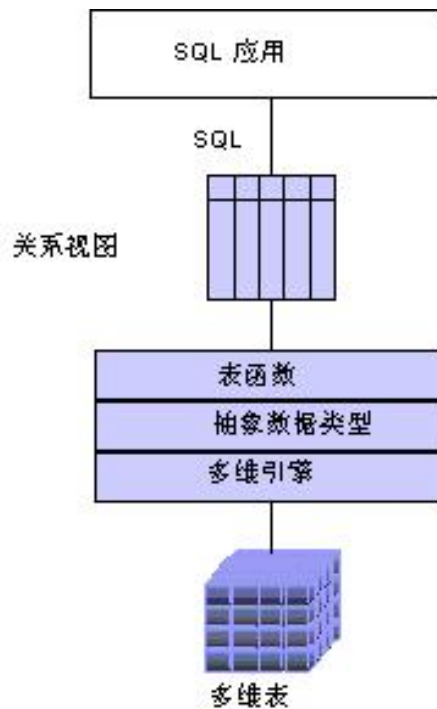
- OLAP DML 通过 PL/SQL 包

在介绍数据访问路径前，有必要首先提供一些有关 Oracle 如何访问多维数据表中数据的背景信息。

Oracle OLAP 10g 使用 Oracle 数据库的对象技术向 SQL 表示多维表。这是利用 *抽象数据类型* (ADT) 和 *表函数* 实现的。抽象数据类型描述 SQL 的数据格式（即列和行），表函数从多维表（分析工作空间）中提取数据。表函数将 SQL 语句元素（SELECT 和 WHERE 子句）转换成 OLAP 选项的多维引擎可以理解的命令。

表函数需要针对特定数据源（这里是 OLAP 选项）的 SQL SELECT 语句参数。通过在基于 SQL 的应用程序上放置关系视图，可使表函数对应用程序透明。这样做后，数据源 — 包括多维引擎中执行的汇总和计算 — 对基于 SQL 的应用程序就完全透明了。这使任何基于 SQL 的应用程序都可透明地访问分析工作空间中的数据。

下图描述了 SQL 命令、关系视图、表函数、抽象数据类型和多维引擎之间的关系。本文后面的四个主题将介绍该技术下可用的不同数据访问路径。



#### Java OLAP API

OLAP API 是面向对象的 Java API，提供了封装、抽象和继承。Java OLAP API 提供了连接、多维导航、数据选择、分析功能和游标管理功能。

Java OLAP API 实现了商业智能应用程序与物理数据存储、从数据源获取数据所需的访问方法之间的隔离。应用程序不需要知道数据源 — 关系或多维表 — 或如何访问数据。

这一隔离使 DBA 可以完全自由地管理数据库中的物理存储，同时不会干扰基于 OLAP API 的应用程序。例如，DBA 可通过改变模式自由调整数据库，不必担心这样做是否会影响应用程序的运行。

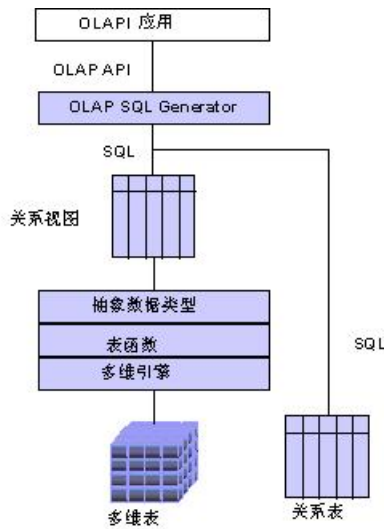
Java OLAP API 还隔离了 Oracle OLAP 10g 应用程序开发者和 SQL。这有两个主要的好处：

当数据存储的关系表中时，解决多维查询所需的 SQL 可能极度复杂。即使是最有经验的 SQL 应用程序开发者也会发现为这些类型的查询生成 SQL 非常困难。生成性能好的 SQL 就更是挑战了。应用程序开发者使用 OLAP API 后可获得更高的生产力。它们可将生成最优 SQL 的任务留给 Oracle OLAP 10g。

使用 OLAP API 的应用程序开发者不需要担心更高版本 Oracle 数据库对 SQL 或 OLAP DML 的更新。随着新的分析和执行功能被引进 Oracle 数据库，OLAP 选项的 SQL 生成器将得到更新，用 OLAP API 写的应用程序将享受无需重写代码的好处。

当应用程序 OLAP API 时，OLAP API 连接数据库、验证用户身份并初始化会话。当通过 OLAP API 发出查询后，OLAP 选项的 SQL 生成器将产生一个针对关系视图或表的 SELECT 语句。如果关系视图从分析工作空间访问数据，抽象数据类型和 OLAP 表函数将服务查询。

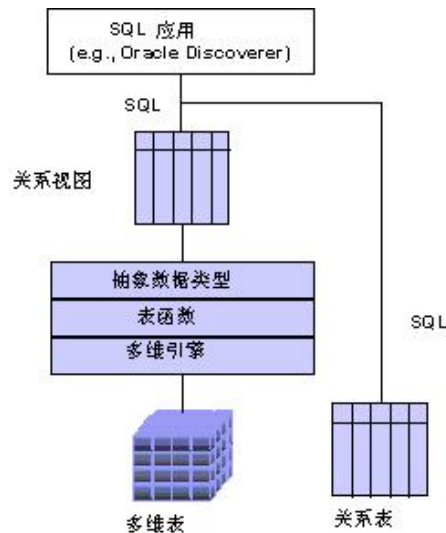
下图中基于 OLAP API 的应用程序给出了一个同时从关系表和多维表（分析工作空间）请求数据的查询。请注意，所有的数据都是在 SQL 中联接的。



#### 选择关系视图的 SQL 查询

分析工作空间在数据库中以关系视图展示，因此可用标准的 SQL 对它们进行查询。也就是说，不需要为了能够访问分析工作空间中的数据而修改基于 SQL 的应用程序，所要做得只是使用标准 SQL 查询以关系视图表示的分析工作空间。同样，发生在多维引擎中的计算对基于 SQL 的应用程序也是透明的。

基于 SQL 的应用程序，例如下图中的 Oracle Discoverer，既可访问多维表（分析工作空间）中的数据，也可访问关系表中的数据。



这里，程序给出了一条指向表示分析工作空间的关系视图和关系表的 SQL SELECT 语句。SELECT 和 WHERE 子句被表函数自动转换成多维引

擎的 OLAP DML 命令，数据通过 OCI 或 JDBC 返回。同样可以注意到的，来自多维表和关系表的数据可在 SQL 中联接。

#### 直接选择表函数的 SQL 查询

Oracle 表函数可被直接查询。也就是说，可以绕过表示表函数的关系视图，直接给出面向表函数的 SELECT 语句。这里，SELECT 语句采取如下形式：

```
SELECT columns FROM TABLE(OLAP_TABLE(analytic workspace,
adt, OLAP DML, data map))
```

*columns* 是抽象数据类型中表述的一个或多个列。这些通常是表示维度成员和事实的列。

*OLAP\_TABLE* 是表函数的名称。

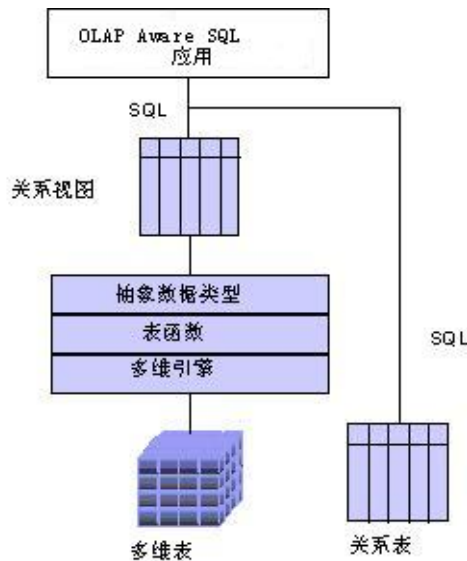
*adt* 是抽象数据类型的名称。

*OLAP DML* 是多维引擎执行的命令。该参数是可选的。

*data map* 描述了 ADT 中列到分析工作空间中物理模型的映射。

在理解直接从表函数选择数据的基本概念时，不必关心表函数、抽象数据类型或数据映射的细节，它们只是提取数据的方法。目前，只需要知道可直接查询表函数以及该方法提供了指定多维数据选择和计算的好机会。

查询路径如下图所示。



这里，应用程序被称为“OLAP 意识”，因为它直接查询表函数。应用程序必须知道 OLAP 选项表函数存在以及函数的参数。

与其他 SQL 查询一样，数据通过 OCI 或 JDBC 返回，可与 SQL 中的其他数据源（关系表或多维表）联接。

直接从表函数选择使应用程序可以以 SELECT 语句的一部分调用 OLAP

存储过程。OLAP 存储过程是在返回数据给客户端前执行的，因此可被用来动态定义和执行分析功能（例如，运行预测、解决模型或定义定制维度成员），或者被用于限制将通过 SQL 返回给客户端的数据。

#### 直接面向多维引擎的 PL/SQL 包

最后一种可用于访问分析工作空间中数据的方法是使用 PL/SQL 包，这将使应用程序可以直接向多维引擎发出 OLAP DML 命令并返回数据给客户端。如果应用程序需要使用 OLAP 命令或 PL/SQL 程序需要访问分析工作空间中的数据，就可以使用这种方法。若分析工作空间中的数据未通过视图或表函数向 SQL 表示，用户又需要这些数据时这种情况就会发生。

OLAP 选项提供了三个允许应用程序与多维引擎进行交互的 PL/SQL 包：

DBMS\_AW.EXECUTE 允许向多维引擎给出 OLAP DML 命令。

DBMS\_AW.INTERP 用于返回少量数据到 PL/SQL 程序的变量中。

DBMS\_AW.INTERPCLOB 用于返回大量数据到 PL/SQL 程序的变量中。

下面的框图显示了一个直接对多维引擎进行 PL/SQL 调用的 OLAP 应用程序。



例如，一个 SQL 应用程序需要在“所有产品”中选择到“2001 年当前为止”与上一年相比销量的百分比变化最大的 10 个产品，并报出销量和百分比变化。度量“销售增长百分比变化”和客户维度成员“到 2001 年当前为止”已在分析工作空间中预先得到定义。

```
DBMS_AW.EXECUTE('limit product to top 10 based on
PctChgSalesYrAgo
(product 'ALL PRODUCTS' customer 'ALL CUSTOMERS'
time 'YEAR TO DATE 2001')')
DBMS_AW.EXECUTE('limit customer to customer.level 'TOP')
DBMS_AW.EXECUTE('limit time to time.level 'YEAR')
SELECT PERCENT.CHANGE.SALES.GROWTH FROM SALES.AW.VIEW
```

注意到这个例子中，OLAP DML 命令被用于选择数据，而一条 SQL 选择语句被用于返回数据。

从前面的四个查询方法可以看到，应用程序决定如何与 OLAP 选项交互和如何访问分析工作空间中的数据时具有极大的灵活性。Java 应用程序可使用 OLAP API 或 SQL。基于 SQL 的应用程序可使用各种技术选择数据，并可直接访问多维引擎。

### **结论**

现在 Oracle 数据库已提供多维技术。企业不再需要在分析数据库和关系数据库之间进行选择。维护两个不同系统的费用被消除了。通过在数据库中集成完整功能的 OLAP 引擎，Oracle 提供了强大和易用的多维分析以及 Oracle 数据库的可管理性、安全和可靠性。结果就是一个用于高端分析和计划应用程序的企业级平台，支持现有基于 SQL 的查询和报表工具。

Oracle 数据库 10g 的 OLAP 选项是唯一用于商业职能的关系 – 多维数据库。



Oracle OLAP 10g: 唯一的关系 – 多维数据库

2004年6月

作者: Bud Endress, Anthony Waite

Oracle Corporation

全球总部

500 Oracle Parkway

Redwood Shores, CA 94065

U. S. A.

全球咨询热线:

电话: +1.650.506.7000

传真: +1.650.506.7200

[www.oracle.com](http://www.oracle.com)

版权所有 © 2004, Oracle。保留所有权利。

本文档只用于提供信息，其中的内容如有更改恕不通知。

不保证本文档中没有错误，也不提供任何其它保证或条件（无论是口头表达还是法律暗示），包括商用的隐含保证和条件或者对特殊目的的适用性。我们明确拒绝与本文档有关的任何责任，并且本文档不构成任何直接或间接的契约义务。未经我们事先的书面许可，不得以任何形式或方法（电子或机械方法）为任何目的复制或传输本文档。

Oracle 是 Oracle Corporation 和/或其会员的注册商标。

其他名称可能是其各自所有者的商标。