

利用 Oracle 数据库 10g 实现
即时数据仓库 — 按企业所需速
度提供信息

Oracle 白皮书

2004 年 3 月

利用 Oracle 数据库 10g 实现按时 数据仓库 — 按企业所需速度提供信息

引言	3
信息可用性的重要意义	3
您的企业需要实时或按时数据吗?	4
按时和批处理	5
通过 ORACLE 数据库 10G 实现按时仓库	6
异步更改数据收集	6
利用 ORACLE 数据库 10G 移动大量数据	9
异类可传输表空间	9
Oracle 数据泵	10
外部表格表数据泵卸载	10
新的可伸缩高速导出和导入工具	11
结论	11

利用 Oracle 数据库 10g 实现按时数据仓库 — 按企业所需速度提供信息

引言

数据仓库已经成为成功企业的基础，许多公司意识到可以通过仓库内的信息进一步利用其数据仓库投资以更加有效地驱动日常业务运作。传统上，数据仓库的作用只是为策略决策的制定提供支持，但是现在其角色已大为拓宽，涉及到越来越多的关键任务，不仅支持策略决策，还支持业务操作流程。

为了在数据仓库中支持这些关键任务，数据库管理系统不应当只是能够简单满足性能、可伸缩性和可管理性的基本要求。此外，成败还取决于系统处理需求日益增长的分析工作量的能力，以及保持数据仓库内数据与来自事务处理系统的数据同步所形成的沉重负担。

极品**优秀**的数据仓库应当可以按照企业所需速度来支持分析，以便最终用户始终能够在需要时使用数据仓库中的信息，如果必要的话，还可以获得“即时信息”。提供对数据仓库内的最新事务处理变更的及时访问成为了一个即便不是最重要也是非常重要的成功条件。

本白皮书论述了实时对于数据仓库的含义，并定义了“按时”提供信息的概念。通过 Oracle，您可以按时获得所需的任何数据；无论企业要求如何，Oracle 都会在您需要时提供所需信息。本白皮书将论述 Oracle 数据库 10g 近乎实时的数据仓库所具有的高效而可伸缩的数据提取和传播能力，其中将重点介绍在从基于 Oracle 的操作系统提取数据并将该数据传播到基于 Oracle 的数据仓库方面所做的功能改进以及新增功能。在当今市场上，一个非常普遍和重要的配置就是将 Oracle 作为所有类型的事务处理系统的基础数据库。包括异类**异构**服务和透明网关在内的 Oracle 完整解决方案论述不在本白皮书的讨论范围之内。

信息可用性的重要意义

近年来，许多与“实时”有关的术语（例如活动的仓库贮存**存储**、零延迟企业、实时分析、实时仓库贮存**存储**）开始更加频繁地出现在文章、研讨会、甚或是与“数据仓库”或“商业智能”相

关的出版物的头版位置¹。毋庸置疑，数据仓库在企业内所扮演的角色和所处位置的重要性已经发生了变化，并且这些系统也已经发展到了充当关键角色的地步，远远超出了传统策略计划工具的范畴。然而，绝大多数与“实时”有关的文献仅论及数据仓库文化领域内这一明显的变化和附加的商业价值。一个有趣的现象是，现在所谓的“为仓库而建的数据库系统”吹嘘其能够处理混有少量并发负载和查询的工作量

（Oracle 已经实现多年的功能）。遗憾的是，这些文章几乎没有触及成功将操作任务混合到数据仓库中所需的最关键技术问题之一：**事务处理数据是如何进入数据仓库的？**

为什么说这个问题重要呢？在理想化的企业中，所有的应用程序都应共享和访问同一企业范围的数据模型（存储在单个系统中）。所有重要数据均只存储一次，并且每个人都能够查看单一的事实源。围绕该核心数据模型而建的是其他数据结构，用于可通过自动和透明方式进行维护的特定用途（例如高性能分析）。遗憾的是，现实世界比这复杂得多。

经过调查，任何一家大中型公司都拥有许多种类繁多的自行研发以及预打包的系统与应用程序。所有这些系统都包含一部分公司数据资产，并且相同的商业信息不但多次进行存储，而且存储格式也各不相同。为了全方位了解一家公司的业务实体（客户信息是 CRM 空间内的一个经典例子），公司必须在其仓库内整合所有这些各不相同的信息池，以创建单一的事实源。

为了在制定策略决策时对数据仓库环境加以利用，公司还必须及时把全部经过整合的信息放入系统之中，以备需要时随时取用。

“按时”数据仓库具有许多重要的因素。数据仓库必须及时地汇聚来自所有源系统的数据。数据仓库还必须直接向最终用户提供这些数据，并支持最终用户所需的任何种类的分析。本白皮书只关注前一个要求，即用“按时”的数据填充数据仓库的能力；如果数据未能“按时”就位，则“按时”数据仓库的其他方面亦将毫无实际意义。

您的企业需要实时或按时数据吗？

在讨论 Oracle 的功能之前，让我们首先从大体上了解一下“实时”数据仓库的概念及其与传统的批处理方式之间的不同之处。严格地讲，“实时”的定义意味着源系统中出现的任何数据更改会立即自动地反映在数据仓库中。这就是说，数据仓库环境下的所有更改都与源系统中的更改同时发生——只有

¹ 数据仓库和商务智能并非业界明确定义的首字母缩略词，但常用作同义词。在本白皮书中，我们统一只用“数据仓库”这一术语。

当两个更改属于同一个原子事务处理的时候才能实现。任何不符合本规则的机制事实上只不过是“接近实时”罢了，在源系统的事务处理和数据仓库系统中的对应条目之间总是显示某些延迟现象。

尽管通过 **Oracle** 技术实现实时已经有很长时间了，但是让数据仓库同步成为原子源事务处理的一部分的性能开销却时常不为人所接受。对大多数数据仓库环境来说，这样的实时是否有必要值得怀疑。我们得承认，数据仓库中的实时总是意味着“接近”实时，当然也有例外情况。

有了实时数据仓库这个新定义之后，我们现在转向下面的问题：你在实时数据仓库中真正寻求的是什么？不仅仅是业务的按时吗？商业活动管理系统真的需要有关您刚在数百家商店中的一家销售出的最近一个廉价物品的信息吗？毫无疑问，对某些数据仓库环境来说即时的精确性是有必要的，但那是例外。一般来讲，最理想的解决方案是无法通过凭空想象或生搬硬套技术来实现的。

最理想的解决方案往往也最为有效：以最低限度的资源消耗通过可靠且可伸缩的方式满足企业要求。**Oracle** 在提供最全面、强大的功能以实现此目标方面可以说是独一无二，本文稍后将对此进行论述。

按时和批处理

按时提供必要信息意味着，在原始事务（或事件）的时间与更改反映在仓库环境中的时间之间可以出现几秒、几分甚至几个小时的延迟。延迟是指从初始创建数据到将该数据填入数据仓库所用的总时间，也是下列流程中各个步骤的延迟时间的总和：

1. 在初始创建之后捕获更改的时间。
2. 从源系统向数据仓库系统传送（或传播）这些更改所用的时间。
3. 为进一步的 ETL 处理做好一切准备所用的时间。例如：等待相关源更改到达。
4. 转换并应用细节更改所用的时间。
5. 维护其他结构（例如刷新物化视图）所用的时间。

这些步骤所用时间会根据不同的体系结构而各异（有些步骤甚至根本不需要），但是，对于特定的仓库体系结构来说，无论是传统的批处理还是按时处理，其步骤是一样的。

那么，传统的批处理和按时处理之间有什么差别呢？按时处理因何优于批处理方式呢？

使用**批处理**时，所有更改都保持“待命”状态，直到下一个预定

的处理发生；例如，许多数据仓库每晚都会进行刷新。数据的加载根据数据仓库管理员定义的固定批处理计划进行，该计划不受自上一次批处理以来所发生的更改数据量以及该更改对数据仓库任务的重要性影响。在**按时处理**中，该流程得到了最大限度地优化以减少总的延迟时间，并进而满足数据仓库关于“按时”的要求，同时最大限度地优化该过程中的每一步，从而以最小的资源消耗量完成数据的加载。这就是通过**Oracle** 实现按时数据仓库所依赖的简单而强大的基本原理。

优化按时处理需要一个进行数据仓库流程中所有工作的功能强大的综合平台。**Oracle9i** 是业界第一个完善的“商务智能平台”。在数据库内提供了进行高效 ETL 处理的服务器基础架构是该数据库的一项主要功能增强，为按时数据仓库的实现打下了基础。**Oracle 数据库 10g** 以其新增功能和增强功能拓展了 **Oracle** 在该领域的领导地位。这些新增功能特别有益于 **Oracle** 系统之间信息的提取和传播，以最少的必要资源消耗实现了高效、可伸缩的按时仓库。

通过 ORACLE 数据库 10G 实现按时仓库

从 **Oracle9i** 开始，针对 ETL 环境下的某些任务，**Oracle** 的数据库能力得到了大幅增强。ETL 处理流程发生了显著变化，而数据库则变成了集成的数据转换引擎。其中引入了诸如外部表格**表**、SQL MERGE 命令和多表格**表** SQL INSERT 等令人兴奋的新功能，造就了“边加载边转换”的范例。²

Oracle 数据库 10g 的功能得到了进一步扩展，尤其是 **Oracle** 系统之间的提取和传播。**Oracle** 在数据仓库环境和操作系统方面均占据着市场主导地位。因此，以 **Oracle** 作为操作系统和数据仓库系统的基础数据库是一种非常普遍的方案，成就了 **Oracle** 独一无二的地位：**Oracle** 特定的功能可以用来更加紧密地集成这些系统之间的数据流。不管 **Oracle** 操作数据库是自行开发还是预打包的应用程序，**Oracle** 均提供成功实现按时数据仓库所必需的功能。

异步更改数据收集

有效的更改数据捕获一直是数据仓库所面临的挑战。从源系统中提取数据时，只有新近修改或新增的数据会受到关注。从概念上讲，可以用两种方法捕获（和传播）增量更改：在应用程序层或在数据层。

在许多情况下，在这两个层上进行数据捕获的解决方案都不

² 有关边加载边转换范例的详细资料，请参见技术白皮书：位于 OTN 上的“**Oracle9i** 中的 ETL 处理”

尽人意。数据层的捕获机制会对操作系统产生极大的性能影响（例如，通过在操作系统内执行触发器记录更改），或者维护数据仓库系统的费用非常昂贵（例如，每天晚上捕获操作系统的完整快照并将这些快照与前一晚上的快照相比较）。应用程序层的捕获机制一直以来面临着其自身的一系列问题，包括对操作系统的性能影响，以及将应用程序层对象从操作系统映射到数据仓库中的数据层对象的这一挑战。

所需要的是一个能够有效地捕获来自某操作系统的任何（或全部）更改的更改捕获机制，同时该机制又不会对操作系统的性能产生任何影响。而 **Oracle 数据库 10g** 的“更改数据捕获”功能完全符合要求。

在实现高效率的按时仓库时，“更改数据捕获”即便不是最重要的，也是不可或缺的元素之一。在按时环境下，提取过程不能将宝贵的时间花费在提取大量的数据并继而决定哪些数据是新的还是修改过的。“更改数据捕获”提供的机制可以只提取必要的信息。将源系统中的提取限制在有关数据的范围之内，这样不但减少了源系统端为执行提取而必须完成的工作量，而且还最大程度地减小了数据量以及从源提取向仓库目标转换的处理时间。

“更改数据捕获”还解决了从源系统中识别新近修改过的数据这一难题。一旦识别出更改的数据，必须使其能够为需要该信息的目标所访问。您必须确保目标对于同一更改过的数据只能获得一次，并且始终保持该更改过的数据处于可以访问的状态，直到所有目标都已能够获得该数据。**Oracle** 更改数据捕获（CDC）不仅能够捕获更改数据，而且还能够将更改数据发布为关系结构，以允许应用程序以受控方式预定更改数据。

Oracle9i 引入了 **Oracle** 更改数据捕获，支持同步的更改数据捕获。发生在某相关源表格的每一个更改都明确地视为原子事务处理的一部分，并被存储在一个更改表格中以备进一步的使用。尽管同步 CDC 对任何类型的应用程序都是完全透明的，但它还是向源表格添加了触发器，因而增添了原始事务的开销（不管是否可以计量）。这一数据库水平上的物理增加导致许多 **DBA** 和应用程序开发人员不使用同步 CDC。

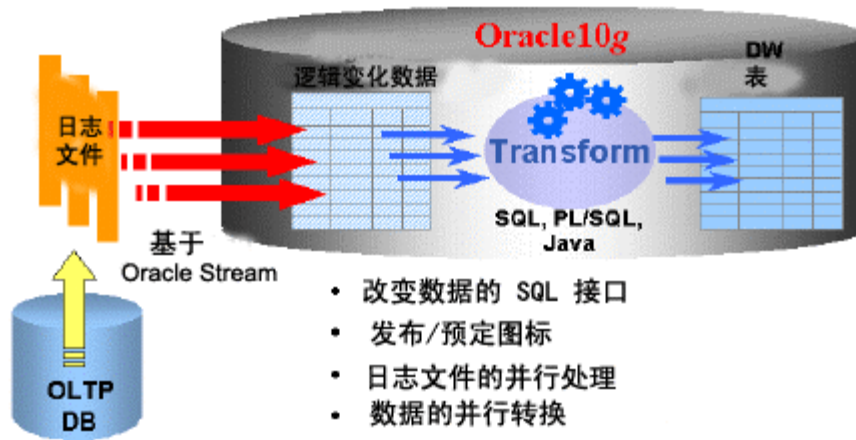


图 1: 异步更改数据捕获的体系结构

Oracle 数据库 10g 支持异步更改数据捕获，从而克服了这个问题。这就为所有 Oracle 系统提供了增量更改跟踪能力，同时对原始源事务处理不会产生任何影响。

异步 CDC 捕获更改时丝毫不会干扰源；更改信息取自日志文件，对原始事务处理系统不做任何改动。

异步 CDC 是专门为数据仓库更改提取和传播设计的轻量级 Oracle Streams 应用程序；对源表格表的更改将以关系格式进行公布，以备进一步使用。因为所有更改都公布为了关系数据，之后的 ETL 处理可以通过 SQL 语言丰富的批处理和并行功能来完成。能够使用 SQL 就意味着能够利用 Oracle 数据库中内置的所有 ETL 功能。在某些 ETL 方案中，源表格表被复制到数据仓库中的临时区域。此时，这些临时表格表甚至可以通过透明方式由 Oracle 管理的更改表格表来取代，而不需要任何更改！

当识别出更改来并将其存入更改表后，就轮到用户使用这些更改信息了。每个用户通过控制其相关时间窗口对其个人的增量更改数据量进行控制。当用户每次使用更改数据时，意味着数据已成功地通过了下游 ETL 过程，用户就可清除旧的数据（逻辑操作）并扩展其相关的窗口以获取下一组需要处理的更改数据。理论上讲，您可以使这种时间窗口无限小，甚至小到秒。但是，时间窗口越小，单个更改数据集中的记录就越少，你从 SQL 的批处理能力中获益也就越小。图 2 显示了关于更改数据量的相关时间窗口的效果。

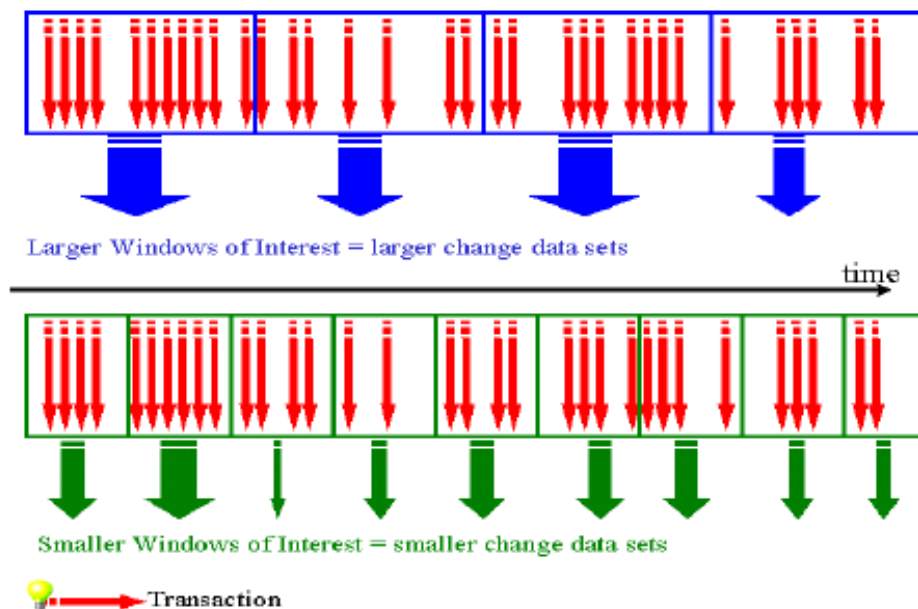


图 2: 相关时间窗口的效果

Oracle 建议不要对时间窗口的大小过于计较。拥有大小合理的更改数据集有助于尽可能多地利用针对任何后续处理的批处理能力。请记住，在大多数情况下真正的实时是不必要的。一旦必须将比较复杂的 ETL 转换应用到捕获数据时，稍大的时间窗口将在不显著影响整体延迟时间的前提下最大程度地减少总的系统工作量。有时，转换非常简单，甚至是零转换，同时需要尽可能地接近实时（比较接近普通的复制，与更改数据捕获相反），这种情况很少见。此时你应该考虑利用 Oracle Streams 建立自己连续的少量传送应用程序。

利用 ORACLE 数据库 10G 移动大量数据

下面将要论述专门为 Oracle 系统之间移动大量数据设计的 Oracle 数据库 10g 的新功能和增强功能。所有将要讨论的功能都是互补性的，可在 Oracle 系统之间实现高效的提取和传输，满足不同的特定业务需求。

异类可传输表空间

引入 Oracle8 的可传输表空间是在数个 Oracle 实例之间共享封装数据的一个良好机制。可传输表空间（TTS）为您提供在数据库之间物理移动封装表空间的机会，而不是利用它来进行逻辑复制。只有表空间的数据字典（即元数据）信息是从源系统导出，又导入到目标系统内的。数据本身可以直接从复制到目标系统中的数据文件访问。

自 Oracle8 以后的每一个 Oracle 版本都显著增强了可传输表空间的功能。Oracle 数据库 10g 为系统间普遍存在的

传输扫清了最后一个大障碍 — 平台依赖性。表空间通过以与平台相关的格式进行存储；平台间的不同之处在于 endian 格式（即字节顺序）。从 Oracle 数据库 10g 开始，表空间是平台敏感的了。如果源平台和目标平台的字节顺序不同，则必须在源平台或目标平台上执行另一个附加步骤，以将正在运输的表空间转换成目标格式。如果平台的字节顺序相同，则不必进行转换，表空间的运输如同在同一平台上执行。该转换是通过 RMAN 实用程序进行的。

可传输表空间是用来从一个 Oracle 数据库向另一个数据库移动数据的高效机制，因为可传输表空间允许在移动数据时不必卸载或重新加载数据。数据运输的时间可以预测，并且不易出错。Oracle 建议，在因业务需要而要在 Oracle 系统之间传输大量数据并想最大程度地为源系统减少工作量时，请使用可传输表空间。源系统上仅有的工作量是提取元数据和将可传输表空间复制（或移动）到其目标地点的潜在 I/O。

Oracle 数据泵

Oracle 数据库 10g 引入了 Oracle 数据泵。这是一个用来在 Oracle 数据库系统之间高速、大批量地移动数据和元数据的统一的、基于服务器的框架。Oracle 数据泵利用直接路径 API 提供了向 Oracle 系统或从其中加载和卸载数据的最快速机制。作为企业级服务器基础结构，Oracle 数据泵为加载和卸载、可重启性以及监视功能提供了完全的并行能力。所有调用 Oracle 数据泵的接口都将进行外部化，以便您可以编写自己的数据移动实用程序。此处，我们不会面面俱到地论述 Oracle 数据泵的所有细节，而是仅仅从 ETL 数据移动的角度对其主要功能进行着重探讨。

外部表格表数据泵卸载

只要您选择以物理方式卸载数据，Oracle 数据泵的服务器端基础架构便会利用外部表格表作为卸载机制。然后，外部表格表数据泵卸载驱动程序将数据卸载到平台独立的 Oracle 专有文件中。这个新的外部表格表卸载功能可以单独使用，无需新的导出/导入工具，这就为卸载随意查询的结果提供了一个强大而又灵活的机制，并将第一批转换和卸载过程本身结合了起来。作为数据卸载一部分的转换最大程度地优化了需要传输的数据量以及卸载过程之后的所有后续 ETL 处理。

- Data Pump 文件的创建

- 改善的 CTAS 句法
- 任意的 SQL SELECT 语句

```
10, Swastika, ... 20.00, 9.00  
20, Star, ... 20.50, 9.50  
30, Trousers, ... 20.00, 9.00  
40, ...
```

- Oracle DMP 格式

- 提取元数据进行转换

- DBMS_METADATA
程序包提供 DLL



图 3: 外部表格表数据泵卸载

新的可伸缩高速导出和导入工具

Oracle 数据泵替代了原来服务器端的 Export (exp) 和 Import (imp) 工具。Oracle 提供了新的导出和导入命令行客户端 — expdp 和 impdp。它们在外观和感觉上都与现有的导出和导入客户端相类似。它们是使 API 调用进入数据泵基础架构的瘦客户端。

这个新的导出和导入工具是可以重启的，支持卸载到平面文件中（通过外部表格表数据泵卸载），也支持数据库到数据库的直接导出/导入模式，从而避免了所有的物理阶段。连同灵活的对象选择和数据子集过滤，可将该新工具视为是 Oracle 实例之间移动对象组（包括完整的数据或数据子集）的简易而快速的方法。

结论

从 Oracle9i 开始，Oracle 的数据库能力得到了显著提高，以专门完成某些 ETL 环境下的任务。Oracle 数据库 10g 又特意针对 Oracle 系统之间的数据提取和传播扩展了这一功能。作为第一个面向事务处理系统且支持高度优化的专用提取和传输机制的数据库，Oracle 提供了目前市场上非常普遍而重要的配置，为利用 Oracle 数据库 10g 实现按时仓库奠定了基础。无论贵企业有何要求，Oracle 都能让您按时获得一切所需数据。



按企业所需速度提供信息

2004 年 3 月

作者: Hermann Baer

副编:

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

版权所有(C) 2003 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。

本文档不保证没有错误, 也不受其他任何口头表达或法律暗示的担保或条件的约束, 包括对特定用途的适销性或适用性的暗示担保和条件。我们特别声明: 拒绝承担与本文档有关的任何责任, 本文档不直接或间接形成任何合约职责。未经预先书面许可, 不允许以任何形式或任何方式(电子方式或机械方式)、出于任何目, 复制或传播本文档。

Oracle 是甲骨文公司和/或其附属公司的注册商标。其他名称可能是其各自所有者的商标。

利用 Oracle 数据库 10g 实现按时数据仓库 第 13 页