第3章

数 据 仓 库

自从 1991 年数据仓库之父 Bill Inmon 提出了数据仓库概念以来，数据仓库已从早期的探索走向实用阶段，进入了一个快速发展阶段。在此期间，全球经济急速发展，激烈的竞争、企业间频繁的兼并重组，使企业对信息的需求大大加剧，这是数据仓库发展的根本原因。当越来越多的企业开始重视数据资产的价值时，数据仓库也就成为必然的选择。

目前企业面对经济增长减缓和竞争日益激烈的双重压力，为继续保持经济的高速稳定增长，大量的企业面临着减员增效、股份制改造等各种变革，准确、全面的信息是企业变革制胜的法宝。随着经营策略从以产品为中心转变为以顾客为中心，数据的潜在价值正在得到越来越多的关注，企业已经认识到充分地利用信息是应对挑战的关键，数据仓库正成为 IT 领域中被关注的热点技术。

信息技术的广泛应用使企业的运营更加高效、灵活，但同时也带来了“数据爆炸”的问题，许多遗留下来的历史数据被束之高阁，人们面对浩如烟海的数据显得手足无措，如何有效地组织和存储数据，把其内部隐藏的信息转化为商业价值，为企业效益提供服务成为决策者们迫切关心的问题。数据仓库作为高效集成、管理数据的技术，为各级决策者洞察企业的经营管理状况，及时发现问题，为提高决策水平提供了基础。目前数据仓库逐渐被越来越多的企业应用。

3.1 从数据库到数据仓库

企业的数据处理大致分为两类。一类是操作型处理，也叫联机事务处理（OnLine Transaction Processing，OLTP），它是针对具体业务在数据库联机的日常操作，通常对少数记录进行查询、修改。用户较为关心操作的响应时间、数据的安全性、完整性和并发支持的用户数等问题。传统的数据库系统作为数据管理的主要手段，主要用于操作型处理。另一类是分析型处理，一般针对某些主题的历史数据进行分析，支持管理决策。

企业经过数年的信息化建设，数据库中都会积累大量的日常业务数据，传统的决策支持系统（DSS）直接建立在这种事务处理环境上。然而传统的数据库对分析处理的支持一直不能令人满意，这是因为操作型处理和分析型处理具有不同的特征，主要体现在以下几个方面。

（1）处理性能。日常业务涉及频繁、简单的数据存取，因此对操作型处理的性能要求较高，需要数据库在很短时间内做出响应。与操作型处理不同，分析型处理对系统的响应并不要求那么苛刻。有的分析甚至可能需要几个小时，耗费大量的系统资源。

（2）数据集成。企业的操作型处理通常较为分散，传统数据库面向应用的特性使数据集成困难。数据分散，缺乏一致性，外部数据和非结构化数据的存在使得很难得到全面、准确的数据；而分析型处理是面向主题的，经过加工和集成后的数据全面、准确，可以有效支持分析。

（3）数据更新。操作型处理主要由原子事务组成，数据更新频繁，需要并行控制和恢复机制。但分析型处理包含复杂的查询，大部分是只读操作。过时的数据往往会导致错误的决策，因此对分析型处理数据需要定期刷新。

（4）数据时限。操作型处理主要服务于日常的业务操作，因此只关注当前的数据。而对于决策分析而言，对历史数据的分析处理则是必要的，这样才能准确把握企业的发展趋势，从而制定正确的决策。

（5）数据综合。操作型处理系统通常只有简单的统计功能。操作型处理积累了大量的细节数据，对这些数据进行不同程度的汇总和聚集有助于以后的分析处理。

总的来说，操作型处理与分析型处理系统中数据的结构、内容和处理都不相同。Bill Inmon 归纳了操作型处理与分析型处理的区别，如表 3.1 所示[1]。

表 3.1　操作型处理与分析型处理的比较

操作型处理	分析型处理
细节的	综合的或提炼的
实体—关系（E-R）模型	星状或雪花模型
存取瞬间数据	存储历史数据，不包含最近的数据
可更新的	只读，只追加
一次操作一个单元	一次操作一个集合
性能要求高，响应时间短	性能要求宽松
面向事务	面向分析
一次操作数据量小	一次操作数据量大
支持日常操作	支持决策需求
数据量小	数据量大
客户订单、库存水平和银行账户查询等	客户收益分析、市场细分等

从以上分析可见，操作型数据库是面向企业日常的数据处理，用户是企业的业务人员，难以支持复杂的数据分析；而分析型处理是面向分析、支持决策的，用户多是企业的各级管理人员，通过对企业运营的历史数据进行分析，得到信息、知识辅助决策。

3.2 数据仓库的概念

从本质上讲，设计数据仓库的初衷是为操作型系统过渡到决策支持系统提供一种工具或整个企业范围内的数据集成环境，并尝试解决数据流相关的各种问题。这些问题包括如何从传统的操作型处理系统中提取与决策主题相关的数据，如何经过转换把分散的、不一致的业务数据转换成集成的、低噪声的数据等。

Bill Inmon 认为数据仓库就是面向主题的（Subject-Oriented）、集成的（Integrated）、非易失的（Non-Volatile）和时变的（Time-Variant）数据集合，用以支持管理决策[1]。数据仓库不是可以买到的产品，而是一种面向分析的数据存储方案。对于数据仓库的概念可以从两个层次理解：首先，数据仓库用于支持决策，面向分析型数据处理，不同于提高业务效率的操作型数据库；其次，数据仓库对分布在企业中的多个异构数据源集成，按照决策主题选择数据并以新的数据模型存储。此外，存储在数据仓库中的数据一般不能修改。数据仓库主要有以下特征。

1．面向主题

在操作型数据库中，各个业务系统可能是相互分离的。而数据仓库是面向主题的。逻辑意义上，每一个商业主题对应于企业决策包含的分析对象。从图 3.1 中可以看出，一个保险公司的数据仓库的主题可能有顾客、政策、保险金和索赔等。

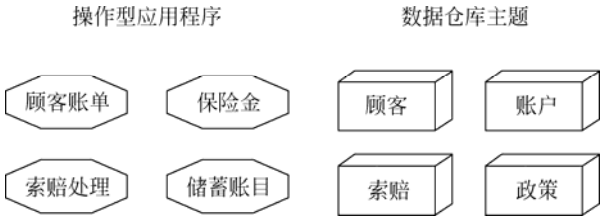

图 3.1　数据仓库的主题

操作型处理对数据的划分并不适用于决策分析。而基于主题组织的数据则不同，它们被划分为各自独立的领域，每个领域有各自的逻辑内涵但互不交叉，在抽象层次上对数据进行完整、一致和准确的描述[2]。一些主题相关的数据通常分布在多个操作型系统中。

2．集成性

不同操作型系统之间的数据一般是相互独立、异构的。而数据仓库中的数据是对分散的数据进行抽取、清理、转换和汇总后得到的，这样保证了数据仓库内的数据关于整个企业的一致性。图 3.2 说明一个保险公司综合数据的简单处理过程，其中数据仓库中与“保险”主题有关的数据来自于多个不同的操作型系统。这些系统内部数据的命名可能不同，

数据格式也可能不同。把不同来源的数据存储到数据仓库之前，需要去除这些不一致。

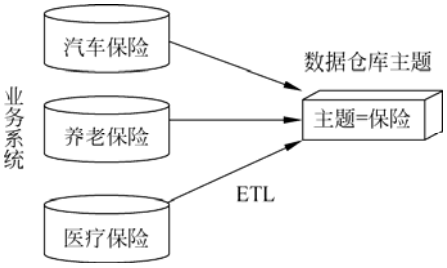

图 3.2　数据仓库的数据集成

3．数据的非易失性

操作型数据库主要服务于日常的业务操作，使得数据库需要不断地对数据实时更新，以便迅速获得当前最新数据，不至于影响正常的业务运作。在数据仓库中只要保存过去的业务数据，不需要每一笔业务都实时更新数据仓库，而是根据商业需要每隔一段时间把一批较新的数据导入数据仓库。事实上，在一个典型的数据仓库中，通常不同类型数据的更新发生的频率是不同的。例如产品属性的变化通常每个星期更新一次，地理位置上的变化通常一个月更新一次，销售数据每天更新一次。

数据非易失性主要是针对应用而言。数据仓库的用户对数据的操作大多是数据查询或比较复杂的挖掘，一旦数据进入数据仓库以后，一般情况下被较长时间保留。数据仓库中一般有大量的查询操作，但修改和删除操作很少。因此，数据经加工和集成进入数据仓库后是极少更新的，通常只需要定期的加载和更新。

4．数据的时变性

数据仓库包含各种粒度的历史数据。数据仓库中的数据可能与某个特定日期、星期、月份、季度或者年份有关。数据仓库的目的是通过分析企业过去一段时间业务的经营状况，挖掘其中隐藏的模式。虽然数据仓库的用户不能修改数据，但并不是说数据仓库的数据是永远不变的。分析的结果只能反映过去的情况，当业务变化后，挖掘出的模式会失去时效性。因此数据仓库的数据需要更新，以适应决策的需要。从这个角度讲，数据仓库建设是一个项目，更是一个过程[3]。数据仓库的数据随时间的变化表现在以下几个方面。

（1）数据仓库的数据时限一般要远远长于操作型数据的数据时限。

（2）操作型系统存储的是当前数据，而数据仓库中的数据是历史数据。

（3）数据仓库中的数据是按照时间顺序追加的，它们都带有时间属性。

数据仓库主要包括数据的提取、转换与装载（ETL）、元数据、数据集市和操作数据存储等部分，常用的数据仓库结构如图 3.3 所示[4]。

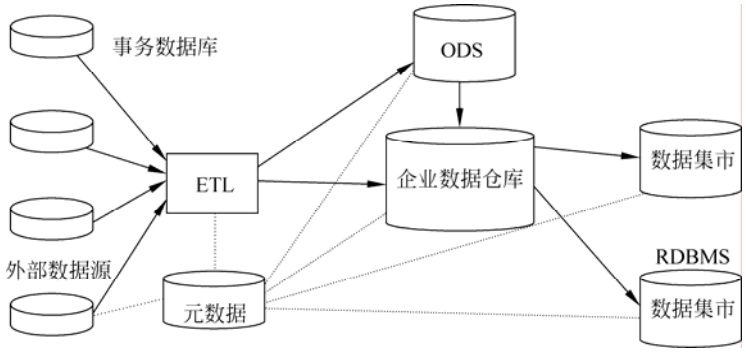

图 3.3　数据仓库的结构

3.3　数据集市

人们在早期开发企业级数据仓库时，一般是先建立一个全局的数据仓库，然后在此基础上建立各种应用，即“自顶向下”的方法。但在开发的过程中会出现以下问题。

（1）如果按“自顶向下”的方法建立企业级数据仓库，建设规模往往较大，建设周期长，投资大。

（2）在数据仓库建好后，随着使用数据仓库的部门增多，对数据仓库资源的竞争将成为企业面临的一个难题。

（3）各个部门希望能定制数据仓库中的数据，但数据仓库是面向企业的。为解决上述问题，人们提出了数据集市的概念，如图 3.4 所示。数据集市支持某一业务单元或部门特定的商业需求，其中的数据可以来自数据仓库。数据集市可以在数据仓库的基础上进行设计（见图 3.4（a）），也可像数据仓库一样直接设计（见图 3.4（b））。数据集市中的数据具有数据仓库中数据的特点，只不过数据集市专为某一部门或某个特定商业需求定制，而不是根据数据容量命名。

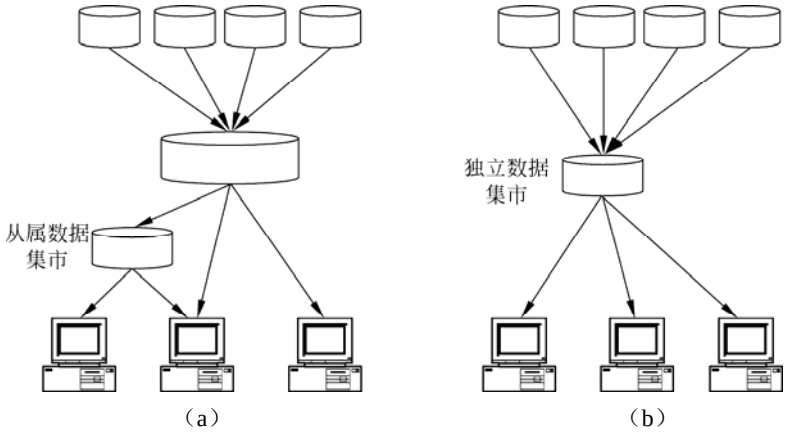

图 3.4　两种类型的数据集市

数据集市面向部门、业务单元或特定应用，因而规模较小，便于快速实现，且成本较低，短期内即可获得明显效果。数据集市的应用不仅满足了部门的数据处理需求，而且作为数据仓库的子集有助于构建完整的企业级数据仓库。

3.4 元数据

当需要了解某地企业及其提供的服务时，电话黄页的重要性就体现出来了。元数据（Metadata）类似于这样的电话黄页。

1．元数据的定义

数据仓库的元数据是关于数据仓库中数据的数据。它的作用类似于数据库管理系统的数据字典，保存了逻辑数据结构、文件、地址和索引等信息。广义上讲，在数据仓库中，元数据描述了数据仓库内数据的结构和建立方法的数据。

元数据是数据仓库管理系统的重要组成部分，元数据管理器是企业级数据仓库中的关键组件，贯穿数据仓库构建的整个过程，直接影响着数据仓库的构建、使用和维护。

（1）构建数据仓库的主要步骤之一是 ETL。这时元数据将发挥重要的作用，它定义了源数据系统到数据仓库的映射、数据转换的规则、数据仓库的逻辑结构、数据更新的规则、数据导入历史记录以及装载周期等相关内容。数据抽取和转换的专家以及数据仓库管理员正是通过元数据高效地构建数据仓库。

（2）用户在使用数据仓库时，通过元数据访问数据，明确数据项的含义以及定制报表。

（3）数据仓库的规模及其复杂性离不开正确的元数据管理，包括增加或移除外部数据源，改变数据清洗方法，控制出错的查询以及安排备份等。

元数据可分为技术元数据和业务元数据。技术元数据为开发和管理数据仓库的 IT 人员使用，它描述了与数据仓库开发、管理和维护相关的数据，包括数据源信息、数据转换描述、数据仓库模型、数据清洗与更新规则、数据映射和访问权限等。而业务元数据为管理层和业务分析人员服务，从业务角度描述数据，包括商务术语、数据仓库中有什么数据、数据的位置和数据的可用性等，帮助业务人员更好地理解数据仓库中哪些数据是可用的以及如何使用。

由上可见，元数据不仅定义了数据仓库中数据的模式、来源、抽取和转换规则等，而且是整个数据仓库系统运行的基础，元数据把数据仓库系统中各个松散的组件联系起来，组成了一个有机的整体，如图 3.5 所示[5]。

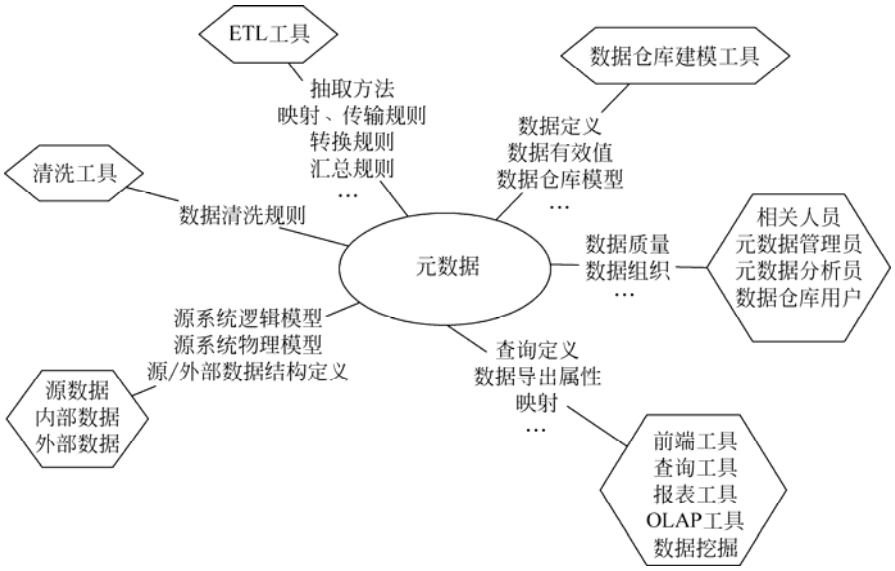

图 3.5　元数据

2．元数据的存储方式

元数据有两种常见存储方式：一种是以数据集为基础，每一个数据集有对应的元数据文件，每一个元数据文件包含对应数据集的元数据内容；另一种存储方式是以数据库为基础，即元数据库。其中元数据文件由若干项组成，每一项表示元数据的一个要素，每条记录为数据集的元数据内容。上述存储方式各有优缺点，第一种存储方式的优点是调用数据时相应的元数据也作为一个独立的文件被传输，相对数据库有较强的独立性，在对元数据进行检索时可以利用数据库的功能实现，也可以把元数据文件调到其他数据库系统中操作；不足是如果每一数据集都对应一个元数据文档，在规模巨大的数据库中则会有大量的元数据文件，管理不方便。第二种存储方式下，元数据库中只有一个元数据文件，管理比较方便，添加或删除数据集，只要在该文件中添加或删除相应的记录项即可。在获取某数据集的元数据时，因为实际得到的只是关系表格数据的一条记录，所以要求用户系统可以接受这种特定形式的数据。因此推荐使用元数据库的方式。

元数据库用于存储元数据，因此元数据库最好选用主流的关系数据库管理系统。元数据库还包含用于操作和查询元数据的机制。建立元数据库的主要好处是提供统一的数据结构和业务规则，易于把企业内部的多个数据集市有机地集成起来。目前，一些企业倾向建立多个数据集市，而不是一个集中的数据仓库，这时可以考虑在建立数据仓库（或数据集市）之前，先建立一个用于描述数据、服务应用集成的元数据库，做好数据仓库实施的初期支持工作，对后续开发和维护有很大的帮助。元数据库保证了数据仓库数据的一致性和准确性，为企业进行数据质量管理提供基础。

3．元数据的作用

在数据仓库中，元数据的主要作用如下。

（1）描述哪些数据在数据仓库中，帮助决策分析者对数据仓库的内容定位。

（2）定义数据进入数据仓库的方式，作为数据汇总、映射和清洗的指南。

（3）记录业务事件发生而随之进行的数据抽取工作时间安排。

（4）记录并检测系统数据一致性的要求和执行情况。

（5）评估数据质量。

4．粒度

粒度反映了数据仓库按照不同的层次组织数据，根据不同的查询需要，存储不同细节的数据。在数据仓库中，粒度越小，数据越细，查询范围就越广泛。相反，粒度级别越高，表示细节程度越低，查询范围越小。例如，当信用卡发行商查询数据仓库时，首先了解某个地区信用卡的总体使用情况，然后检查不同类别用户的信用卡消费记录，这就涉及不同细节的数据。数据仓库中包含的数据冗余程度较高，批量载入和查询会影响到数据管理和查询效率，因此数据仓库采用数据分区存储技术以改善数据仓库的可维护性，提升查询速度和加载性能，解决从数据仓库中删除旧数据时造成的数据修剪等问题，把数据划分成多个小的单元。

根据粒度的不同，可以把数据划分为早期细节级、当前细节级、轻度综合级和高度综合级等[6]。ETL 后的源数据首先进入当前细节级，并根据需要进一步进入轻度综合级乃至高度综合级。一旦数据过期，当前数据粒度的具体划分会直接影响到数据仓库中的数据量以及查询质量。图 3.6 显示了典型数据仓库的粒度结构。

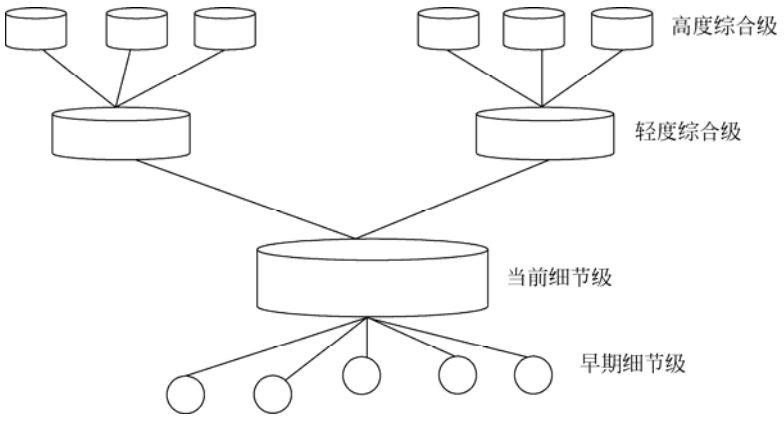

图 3.6　数据仓库粒度

数据仓库数据的多粒度化为用户使用数据提供了一定的灵活性，例如家用电器销售数据可以同时满足市场、财务和销售等部门的需要，财务部若要了解某地区的销售收入，只需改变相关数据的粒度即可。

3.5 ETL

数据仓库并不只是数据的简单累积，而是经过一系列的抽取、转换和装载的过程，简称 ETL。ETL 是构建数据仓库的重要环节，对数据仓库的后续环节影响比较大。目前市场上主流的 ETL 工具有 Informatica 公司的 Power Center、IBM 公司的 Data Stage、Oracle 公司的 Warehouse Builder 以及 Microsoft 公司的 SQL Server IS 等。下面简要介绍 ETL 的主要功能。

1．数据抽取

数据仓库是面向主题的，并非源数据库的所有数据都是有用的，所以在把源数据库中的相关数据导入数据仓库之前，需要先确定该数据库中哪些数据是与决策相关的，数据抽取的过程大致如下。

（1）确认数据源的数据及其含义。

（2）抽取。确定访问源数据库中的哪些文件或表，需要提取其中哪些字段。

（3）抽取频率。需要定期更新数据仓库的数据，因此对于不同的数据源，需要确定数据抽取的频率，例如每天、每星期、每月或每季度等。

（4）输出。数据输出的目的地和输出的格式。

（5）异常处理。当需要的数据无法抽取时如何处理。

2．数据转换

数据仓库的数据来自多种数据源。不同的数据源可能由不同的平台开发，使用不同的数据库管理系统，数据格式也可能不同。源数据在被装载到数据仓库之前，需要进行一定的数据转换。数据转换的主要任务是对数据粒度以及不一致的数据进行转换。

（1）不一致数据转换。数据不一致包括同一数据源内部的不一致和多个数据源之间的数据不一致等类别，例如在一个应用系统中，BJ 表示北京，SH 表示上海，GZ 表示广州。而在另一个应用系统中，对应的代码分别为 1、2 和 3。此外，不同业务系统的数量单位、编码或值域需要统一，例如某供应商在结算系统的编码是 990001，而在 CRM 中编码是 YY0001，这时就需要抽取后统一转换编码。

（2）数据粒度的转换。业务系统一般存储细粒度的事务型数据，而数据仓库中的数据是用于查询、分析的，因此需要多种不同粒度的数据。这些不同粒度的数据可以通过对细

粒度的事务型数据聚集（合）（aggregation）产生。

3．数据清洗

数据源中数据的质量是非常重要的，低劣的“脏”数据容易导致低质量的决策甚至是错误的决策。此外，这些“脏”数据或不可用数据也可能造成报表的不一致等问题。因此有必要全面校验数据源的数据质量，尽量减少差错，此过程是数据清洗（Data Cleaning），也叫数据的标准化。目前一些商务智能企业提供数据质量防火墙，例如Business Objects（SAP）的 Firstlogic，它能够解决数据的噪声。清洗后的数据经过业务主管确认并修正后再进行抽取。数据清洗能处理数据源中的各种噪音数据，主要的数据质量问题有以下几种。

（1）缺失（Missing）数据。即数据值的缺失，这在顾客相关的数据中经常出现，例如顾客输入个人信息时遗漏了所在区域。

（2）错误数据。常见的错误数据包括字段的虚假值、异常取值等。例如在教学选课系统中，选修某门课程的人数不能够超过该课程所在教室的座位数。这些错误数据产生的主要原因是由于业务系统在数据输入后不能进行正确性判断而被录入数据库。错误数据需要被及时找出并限期修正。

（3）数据重复。数据重复是反复录入同样的数据记录导致的，这类数据会增加数据分析的开销。

（4）数据冲突。源数据中一些相关字段的值必须是兼容的，数据冲突包括同一数据源内部的数据冲突和多个数据源之间的数据冲突。例如一个顾客记录中省份字段使用 SH（上海），而此顾客的邮政编码字段使用 100000（北京地区的邮政编码）。冲突的数据也需要及时修正。

4．数据装载

数据转换、清洗结束后需要把数据装载到数据仓库中，数据装载通常分为以下几种方式。

（1）初始装载。一次对整个数据仓库进行装载。

（2）增量装载。在数据仓库中，增量装载可以保证数据仓库与源数据变化的同期性。

（3）完全刷新。周期性地重写整个数据仓库，有时也可能只对一些特定的数据进行刷新。

在初始装载后，为维护和保持数据的有效性，可以采用更新和刷新的方式：更新是对数据源的变化进行记录，而刷新则是指在特定周期数据完全重新装载。

3.6 操作数据存储

数据仓库实现了操作型数据与分析型数据的分离，从而为企业建立了数据库—数据仓