

TP31 **ECLIPSE** MV系列机资料

AOS / VS

BASIC 用户手册

苏州电子计算机厂情报室

ECLIPSE MV 系列机资料译丛

数据库管理系统 (DG/DBMS)

使 用 指 南

成都科技大学 汤德庆译 李宏信校

ECLIPSE MV 系列机资料译丛编辑部

此文译自 Data General 公司软件资料

A Guide to using the
Data General/Database Management
System
(DG/DBMS)

069—000025—00

出版说明

ECLIPSE MV 系列计算机是美国 DG 公司 1980 年 4 月宣布的 32 位超级小型机，时至今日先后完成生产和投放市场的有 ECLIPSE MV/4000, MV/6000, MV/8000 和 MV/10000 等 4 种机型。该系列无论是在软件和硬件方面都有其特点，并在继续向前发展。

为了满足 ECLIPSE MV 系列机用户的需要，我们组织力量对 MV 系列机的资料进行了翻译，并取名为《ECLIPSE MV 系列机资料译丛》，作为内部资料发行。

资料所涉及的范围十分广泛，说明清晰和具体。第一批资料共计 30 余种。本译丛并将追踪该系列的发展、新资料的发行，陆续翻译并发行。

第一批资料包括：

(1) MV 系列机的操作原理，并按机型有单行本的说明。

(2) 先进的带虚拟存贮的操作系统(AOS/VS)。它包括操作系统的生成、运行和程序人员手册等。

(3) 各种高级语言。包括各种常用的高级语言的新品种和新版本，如 FORTRAN77、COBOL、PASCAL 和 PL/1 以及其他一些常用高级语言。

(4) 系统检查、诊断和维护手册。包括现场维修手册、系统练习程序及可靠性检查有关资料。

(5) 其他。包括各种服务性程序及用户手册等。

《ECLIPSE MV 系列机译丛》不仅是 ECLIPSE MV 系列机的系统管理人员、系统操作人员、程序设计人员和用户所必须阅读的资料，而且对于从事计算机工作的科技人员和有关专业的大专院校师生都有很好的参考价值。

参加编译的单位有： 计算机管理总局 清华大学 西南师范学院 成都科技大学 山东海洋学院 华中师范学院 陕西师大 东北师大 北京第七建筑设计院 上海冶金所 苏州电子计算机厂等

编辑部成员：

总 编： 房家国(清华大学)

副 总 编： 李宏信(成都科技大学)

邱玉辉(西南师范学院)

刘连棣(天津计算机研究所)

编 委： 吴 奇(计算机管理局)

李邦几(华中师院)

曹豫莪(陕西师大)

刘长欢(东北师大)

徐 斯(山东海洋学院)

姚林声(上海冶金所)

责任编辑： 金传祚(苏州电子计算机厂)

限于译者水平，加之时间紧促，缺点错误在所难免，敬请读者及时提出批评指正。

ECLIPSE MV系列机资料译丛编辑部

目 录

前 言	(1)
第一章 DG/DBMS 简介	(3)
§ 1—1 DBMS 的一般概念	(3)
§ 1—1.1 关于CODASYL的传统	(4)
§ 1—1.2 数据库一瞥	(4)
§ 1—2 DBMS 的优点	(5)
§ 1—3 关于 DG/DBMS	(6)
§ 1—3.1 DG/DBMS 的操作环境	(7)
§ 1—3.2 DG/DBMS 的组成部分	(7)
§ 1—3.2.1 总控程序与控制程序	(8)
§ 1—3.2.2 数据定义程序	(8)
§ 1—3.2.3 COBOL 接口	(8)
§ 1—3.2.4 DG/DBMS IQ	(9)
§ 1—3.2.5 其它实用程序	(9)
§ 1—3.3 基本情况与轮廓外貌	(9)
§ 1—4 小 结	(9)
第二章 DG/DBMS 的数据描述能力和数据操纵能力	(10)
§ 2—1 数据描述语言	(10)
§ 2—2 模式各组成部分的描述	(10)
§ 2—2.1 记录类型与数据项	(10)
§ 2—2.2 系类型	(11)
§ 2—2.3 系统系	(12)
§ 2—2.4 系类型的描述	(12)
§ 2—2.5 属籍类别	(12)
§ 2—2.5.1 以插入方式分类	(12)
§ 2—2.5.2 以保持方式分类	(13)
§ 2—2.6 最大属记录数	(13)
§ 2—2.7 属记录排序方法	(14)
§ 2—3 关于子模式	(14)
§ 2—3.1 子模式的组成	(15)

§ 2—3.2 记录类型与系类型的选择·····	(15)
§ 2—3.3 数据的安全性·····	(15)
§ 2—3.4 规划完成之后·····	(16)
§ 2—4 数据库的操纵·····	(17)
§ 2—4.1 指 针·····	(17)
§ 2—4.2 处理块·····	(18)
§ 2—4.3 使用处理块·····	(18)
§ 2—4.4 后备及修复·····	(19)
§ 2—5 故障修复·····	(20)
§ 2—6 小 结·····	(20)

第三章 数据库设计注意事项和数据定义程序····· (21)

§ 3—1 一般性的设计注意事项·····	(21)
§ 3—2 哈特大学·····	(22)
§ 3—2.1 定义信息要求·····	(22)
§ 3—2.2 将实际情况模型化·····	(23)
§ 3—2.3 哈特大学数据库结构概要·····	(25)
§ 3—3 使用数据定义程序 (DDF) ·····	(27)
§ 3—3.1 功能键·····	(28)
§ 3—3.2 使用屏幕的一般知识·····	(28)
§ 3—3.3 启动 DDF ·····	(29)
§ 3—3.4 定义记录类型·····	(32)
§ 3—3.5 定义系类型·····	(33)
§ 3—3.6 输入子模式的定义·····	(35)
§ 3—3.7 DDF 的其余特点 ·····	(39)
§ 3—4 小 结·····	(40)

第四章 COBOL 接 口 ····· (41)

§ 4—1 COBOL DDL 语言 ·····	(41)
§ 4—1.1 描述子模式·····	(42)
§ 4—1.2 描述系·····	(42)
§ 4—1.3 描述记录和数据项·····	(42)
§ 4—2 DML 的命令 ·····	(43)
§ 4—2.1 打开子模式 (READY 命令)·····	(44)
§ 4—2.2 关闭子模式 (FINISH 命令) ·····	(44)
§ 4—2.3 开始一个处理块 (INITIATE 命令)·····	(44)
§ 4—2.4 结束一个处理块 (COMMIT 命令)·····	(44)
§ 4—2.5 退回一个处理块 (ROLLBACK 命令) ·····	(44)

§ 4—2.6 找出记录值 (FIND 命令)	(44)
§ 4—2.6.1 寻找指针 (FIND CURSOR)	(45)
§ 4—2.6.2 寻找主记录 (FIND OWNER)	(45)
§ 4—2.6.3 顺序寻找 (FIND Sequential)	(45)
§ 4—2.6.4 依次寻找 (FIND Ordinal)	(45)
§ 4—2.6.5 用数据项寻找 (FIND Using Data Item)	(45)
§ 4—2.6.6 用排序键寻找 (FIND Using Sort Key)	(45)
§ 4—2.6.7 用重复数据项和重复排序键寻找 (FIND Using Duplicate Data Items and Sort Keys)	(46)
§ 4—2.7 有关记录与系联接的操作	(46)
§ 4—2.7.1 向系中插入记录 (CONNECT 命令)	(46)
§ 4—2.7.2 从系中移走记录 (DISCONNECT 命令 和 RECONNECT 命令)	(46)
§ 4—2.8 有关记录值的操作	(46)
§ 4—2.8.1 输入数据 (STORE 命令)	(46)
§ 4—2.8.2 检索数据 (GET 命令)	(46)
§ 4—2.8.3 修改数据 (MODIFY 命令)	(47)
§ 4—2.8.4 从数据库中删去一个记录值 (ERASE 命令)	(47)
§ 4—2.9 关于选择性子句	(47)
§ 4—2.9.1 ASSIGN 子句	(47)
§ 4—2.9.2 AT ERROR 子句	(47)
§ 4—2.10 检查条件子句	(47)
第五章 使用 DG/DBMS IQ	(49)
§ 5—1 获取数据库中的信息	(49)
§ 5—1.1 自由检索命令	(50)
§ 5—1.2 其它有用的命令	(50)
§ 5—1.2.1 完成简单算术运算 (COMPUTE 命令)	(51)
§ 5—1.2.2 将数据域排队 (SORT 命令)	(52)
§ 5—1.3 宏查询命令	(53)
§ 5—1.4 对子模式的要求	(57)
§ 5—2 使用 DG/DBMS IQ 的准备工作	(57)
§ 5—2.1 实用程序 IQPATH	(57)
§ 5—2.2 实用程序 IQUER	(57)
§ 5—3 小 结	(59)
词汇表	(60)

前 言

- DG/DBMS 是什么?
- 数据库管理系统有哪些优点?
- DG 公司的数据库管理系统 (DG/DBMS) 能为你做哪些事?

本手册是为那些计划要用 DG/DBMS 来设计和实现一个应用子系统的人们提出、并回答上述问题的。我们要讨论数据库管理系统 (DBMS) 的优点, 还要讨论如何通过 DG/DBMS 的功能和特点, 来使这些优点得以实现。

本手册适于对下列各领域在技术上了解并且熟悉的人使用, 这些领域是: 软件开发的各个阶段, 计算机系统, 传统的文件系统, 以及普通的数据库知识。本手册一般性地介绍 DG/DBMS 的功能和特点, 并不想详细地叙述。对于这些特点以及有关的实用程序的更多的知识, 请参阅下列手册:

- DG/DBMS 参考手册 (093—000163)
- 交互式查询程序 (DG/DBMS IQ) 用户手册 (093—000168)

由于 DG/DBMS 是在先进操作系统 (AOS) 的管理下运行的, 而且, DG/DBMS 支持用 COBOL 语言写成的应用程序, 故, 可能还要参阅下列手册:

- 学习使用 AOS 系统 (069—000018)
- 先进操作系统 (AOS) 入门 (069—000016)
- AOS 命令行解释程序用户手册 (093—000123)
- AOS 程序员手册 (093—000120)
- AOS 系统管理员指南 (093—000193)
- AOS 操作员指南 (093—000194)
- COBOL 参考手册 (AOS) (093—000223)

DG 公司的其它有关手册的清单, 可在《AOS 软件资料指南》(093—000202) 中找到。本手册的内容组织如下:

- 第一章 从技术上介绍基本概念, 来回答这样一个问题: DBMS 是由哪些部分构成的? 本章还一般性地讨论 DG/DBMS 的组成、它的操作环境、有关的产品、以及它的能力。
- 第二章 叙述了 DG/DBMS 基本的数据描述功能和数据操作功能。
- 第三章 以一个假想的单位为例, 来讨论在设计数据库时所涉及到的某些基本概念。本章还展示了如何运用数据定义程序 (DDF) 来输入数据库的描述 (这是数据库设计的步骤之一)。
- 第四章 讨论了 COBOL 与 DG/DBMS 的接口。其中包括设计 COBOL 应用程序必须用到的数据描述语言 (DDL) 的句法和数据操纵语言 (DML) 的命令。
- 第五章 介绍了 DG/DBMS 的交互查询程序及其某些能力。

- 词汇表 包括了理解和使用 DG/DBMS 所需要的基本术语的简明定义。

请 读 者 注 意

本手册中，我们约定使用以下命令格式：

COMMAND required [optional] ...

其 中 意 义

COMMAND	你必须输入此命令(或它的可接受的缩写)。
required	你必须输入某个变元(如，文件名)。有时我们用以下形式
	$\begin{cases} \text{required 1} \\ \text{required 2} \end{cases}$
	来表示你必须输入其中一个变元。请勿把花括号也输进去，它仅表示其中为必选项。
[optional]	对是否输入此变元，你有选择自由。请勿输入方括号，方括号仅用来表示其中为可选项。
...	你可重复前已输入的一项或数项。它将准确地告诉你，可重复项是什么。

另外，我们以特别的方式使用某些符号：

符 号 意 义

↵ 按终端键盘上的 NEW LINE 键或 RETURN 键。

□ 一定要在此处放置空格。(我们只有在必须时才用这个符号；在一般情况下，你能看出何处应放空格。)

所有的数均是十进制的；指明者除外，例如 35₈。

最后，在例子里，用正体字符表示用户的输入信息，而用斜体字符表示系统的询问和响应。(译者注)

右括号) 是 AOS CLI 的提示符。

与 DG 公司的联系办法

- 若你对本册有建议——请使用附在索引之后的意见表。
- 若你需要另外的手册——请与当地的 DG 公司销售代表联系。
- 若你遇到软件实际使用中的问题——请通知当地的 DG 公司的系统工程师。

第一章 DG/DBMS 简介

Data General 公司的数据库管理系统 (DG/DBMS) 既支持在线应用方式, 也支持批处理应用方式。DG/DBMS 是在 ECLIPSE 计算机上的先进操作系统 (AOS) 的管理下运行的。DG/DBMS 使你能借助于数据库中记录间的相互关系来存贮、检索和更改记录。DG/DBMS 的接口既可以是 ANSI' 74 COBOL, 也可以是 DG/DBMS IQ, 后者是无以伦比的交互式整询工具。DG/DBMS IQ 使你能访问数据库, 且以易于阅读的报表形式显示和/或打印数据。DG/DBMS 的另一特点是具有强有力的后备和修复手段, 这样, 无论是软件或硬件发生故障时, 均能够保证数据库的一致性, 并使停用时间最少。

在这一章中, 我们首先讨论一般数据库管理的概念, 然后再讨论 DG/DBMS 的一般情况。

§ 1—1 DBMS 的一般概念

所有那些严重依赖于逐日变化的信息的单位都遇到一个相同的问题, 即: 我们怎样才能控制这些数据、而不是让数据来控制我们?

解决这个问题的一個办法就是借助于数据库管理系统。这样, 一个部门(或其一部份)就可把它的数据记录放到一个可共享的数据库中, 这就减少了数据共同部份的重复性, 从而使你对数据的访问和对数据的修改都更容易和更有效了。但是仅仅把数据汇集起来是不够的, 还必须有一个系统来管理数据库。这个系统在允许你开发信息库的同时, 应当提供某种安全措施和维护功能。换句话说, 数据库管理系统应当使多个应用程序(或者说多个用户)从数据维护工作和数据存贮结构的工作中解放出来, 而把精力集中到数据的操纵处理上。

为达到这些目标, 大多数数据库都由下列要素(以某种形式)组成:

- 文件管理系统。它是用来分配数据库的空间, 对物理存贮设备进行存取和完成输入输出操作的。这些功能的多数, 通常是由管理 DBMS 运行的操作系统来实现的。

- 数据操纵语言 (DML)。这个语言用来支持操纵数据库所必需的操作: 如存贮、修改、检索和恢复。

- 数据库控制/执行系统。它是数据库应用程序和操作系统(或文件管理系统)的中介面。它应能执行来自应用程序的 DML 命令, 同时要插入存取要求并保存修改信息以利于修复工作。

- 数据描述语言 (DDL)。这个语言是用来定义数据库各个不同的组成部份、它们的相互关系以及组成一体的途径。

- 数据定义程序。它能接受 DDL 的说明, 检查这些说明是否准确, 并产生出这些说明的源码形式以供应用程序存取时用

- 查询/报告处理工具。它使得用户借助于一个简单的命令集合, 就可以交互式地访问数

数据库。

- 保密控制措施。用以防止未经许可的对数据库的访问。
- 修复机构。在软件或硬件发生故障之后修复数据库。

另外,许多 DBMS 装置把设计和维护的责任交给专门一个人(或一组人)来承担,他通常被称为数据库管理员 (DBA—Database Administrator)。DBA 有责任控制数据和保持数据的安全。已获批准的任何用户均可使用数据,但是只能 DBA 才能改变整个的数据结构。

§ 1—1.1 关于 CODASYL 的传统

随着数据库管理系统的发展,对其标准化的需要也与日俱增。因规定 COBOL 语言 而闻各于世的数据系统语言会议 (CODASYL),从六十年代以来一直致力于研究有关数据库的详细说明。尽管 CODASYL 并未提出有关数据库工作的正式标准,然而它发表的一系列报告建立并澄清了有关数据库的概念。目前,已有许多已经实现的数据库管理系统都是建立在 CODASYL 的两个报告的基础之上的,这两个报告就是数据定义语言委员会 (DDLC) 1978 年研究报告和程序设计语言委员会 (PLC) 1978 年研究报告。

这些报告介绍并澄清了的某些有关数据库的基本概念如下:

- 数据库本身。它由所有不同的数据记录组成;每一个记录,依次又由许多称之为数据项 (有时又称为域)的基本信息单位(如姓名,住址,部门号)组成。数据库还包含有有关不同数据记录间相互关系的信息。

- 模式。它对欲存放到数据库中的所有不同类型的记录,以及在记录间想建立的关系进行详细的描述。这些记录间的关系叫做系; DBMS 把系作为路径来确定所需数据的位置。对模式的描述通常是独立于应用语言的。也就是说,这些描述是用独立于任何具体的程序设计语言的语法写成的。CODASYL 把它称为数据描述语言 (DDL)。

- 子模式。它描述了模式中记录类型和系类型的一个特定的子集,它是用来限制具体的用户对数据库的观察范围的。每个用户必须通过子模式才能操纵数据库。这也就是说, DBMS 应该只允许用户检查和/或修改那些包括在其子模式之中的数据项、记录类型和系类型。而且,子模式设计者应能将某些数据项排除于子模式之外,并能重新安排某些数据项,以使子模式对相关的应用程序尽可能有用。这样,子模式提供了一种限制用户对数据库的访问的手段,同时又给予每个用户全部所需要的数据库中的元素。

与模式不同,子模式是通过特定的应用程序设计语言来定义的。因此,每一个用户程序都要包含一个特定的子模式的副本才能访问数据库。

- 数据操纵语言和数据定义语言的语句。它们是针对标准的 ANSI COBOL 语言的扩充,与 COBOL 应用程序一起实现对数据库的操作。

§ 1—1.2 数据库一瞥

现在,让我们稍微深入一步看看数据库的多个部份。如上所述,数据项是最基本的信息单位。数据项的集合叫做记录类型,而记录类型是描述可存放于数据库中的一类数据值的模型。例如,叫做 STUDENT 的记录类型可包含的信息如下:

- 姓名
- 家庭地址
- 性别

- 出生日期
- 社会保险号
- 中学校名

由于数据库要保存的是特定的数据值、而非一般性的数据描述，故把数据类型与该数据类型的值 (Occurrence) 区别开来是非常重要的。记录值是由记录类型中各数据项的一般性描述所对应的具体数据所组成的。记录类型 STUDENT 的值的一例可以是：

Kevin H. Barrett
16 Wimpole Street
East Drummer, MA 01699
Male
8—29—50
021—36—9873
Hoover High School

类似地，系类型是描述数据库中两个记录类型之间关系的模型。例如，在某大学的数据库中，系类型 ON-CAMPUS-HOUSING 可定义记录类型 DORM 和 STUDENT 之间的关系。只要在数据库中存在记录类型 DORM 的一个值，也就存在这个系类型的一个值。因此，如图 1—1 所示，一个描述了特定宿舍 Towers 的记录类型 DORM 的值，在系类型 ON-CAMPUS-HOUSING 的一个值中，与名叫 Tracey Roach 的学生记录值相联系起来。

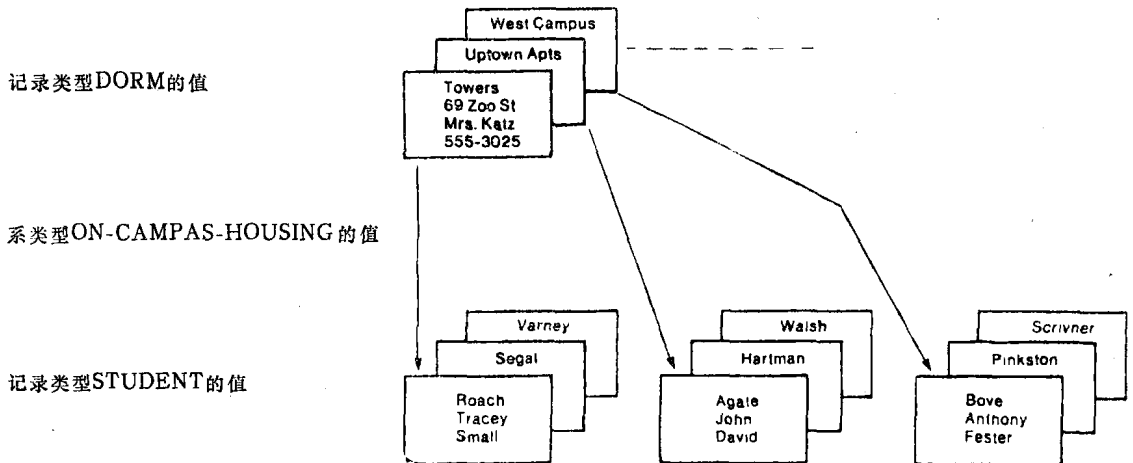


图 1—1 类型和类型的值

§ 1—2 DBMS的优点

DBMS应该把对数据的控制集中起来，对每一数据仅存贮一次而不管有多少应用程序要分享此数据；而且，数据库应该是很完善的，也就是说数据库应当能为多个用户和/或多种不同的应用目的保存数据。这就导致所谓“数据独立性”的概念。

我们所说的数据独立性是什么意思呢？简单地说来，就是数据的实际存贮形式与数据在

应用程序中的表现形式不必是一样的。应用程序使用的数据结构是逻辑结构，而实际存在于存贮介质上的数据结构则是物理结构。换句话说，程序设计师可用某种格式，即逻辑结构来想象数据，而数据的物理存在则可以有不同的方式。图 1—2 的例子表明了逻辑记录与物理记录的差别。

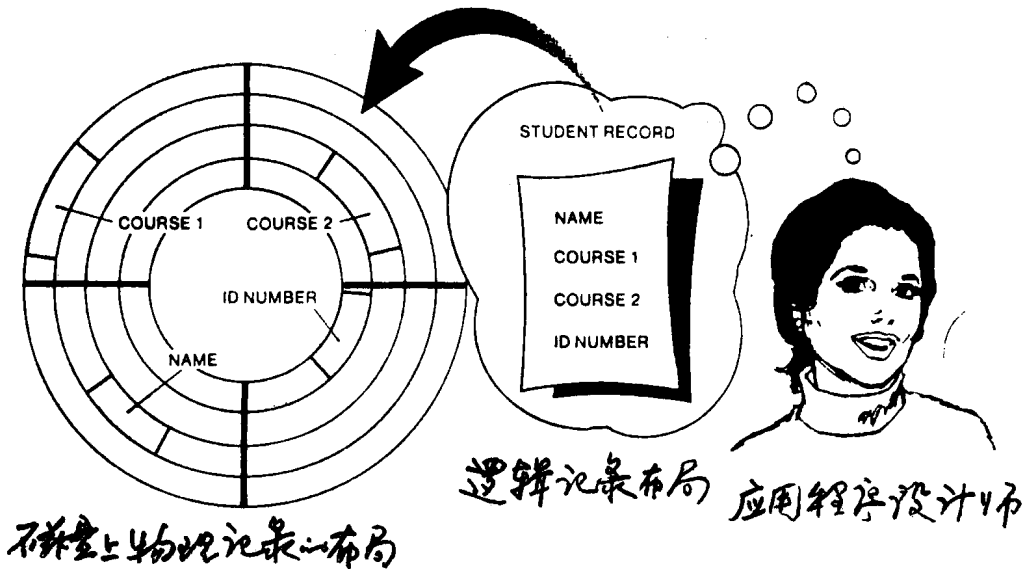


图 1—2 逻辑记录与物理记录之间的差别

物理数据独立性，即把数据的逻辑结构与数据的物理地点相分离，意味着使用数据的应用程序并不取决于任何特定的存贮结构与存取策略。数据的物理存贮结构可以修改以满足变化着的要求，然而并不影响现存的应用程序。

逻辑数据独立性指的是，把一种具体应用对数据的观察范围与所有其余的具体应用的观察范围分开，从而不影响别的应用。

DBMS 还必须保持数据库中的数据的安全性和保密性。数据安全性指的是，防止有意无意地把数据向未经许可的人泄露，或者是未经许可的修改或销毁数据。保密性指的是，个人或组织决定与其有关的信息何时、如何、到何种程度发送给他人的权利。换句话说，安全的数据能免遭偷窃与无意的破坏，保密的数据能不被未经许可的眼睛偷看。

由于各种各样的用户分享相同的数据，故无论安全性和保密性都是难题，因此 DBMS 应能允许你控制对数据的访问，标识并监视所有的数据库的用户。

因此，DBMS 应当是一个能使许多用户有效而方便地访问和操纵数据库的完善的软件包。下一节，我们讨论 DG/DBMS 是如何支持上述各个方面的。

§ 1—3 关于 DG/DBMS

在这之前，我们已一般性地讨论了数据库管理系统，现在让我们来看看 DG/DBMS 本身。Data General 公司遵循 CODASYL 的独创性建议并综合其许多可取之处研制了 DG/

DBMS。例如，DG/DBMS 支持 CODASYL 关于多键记录的概念和分级的网状数据库结构的概念。而且，我们还做了某些改进使我们的 DBMS 更加具有通用性和灵活性。

在 Data General 的实现方案中，DG/DBMS 是创建和操纵数据库的中心机构，而辅之以 COBOL 与 DG/DBMS IQ，它们是研制数据库应用程序的辅助工具。另外，由于 DG/DBMS 是在先进操作系统 (AOS) 的管理下运行的，所以应用程序可以使用 AOS 提供的全部硬件和软件资源。

为更好地理解 DG/DBMS，以下，我们简单地讨论 DG/DBMS 的操作环境，然后再谈谈 DG/DBMS 的主要组成部分。本手册以后各章会更详细地讨论这些部分的功能。

§ 1—3.1 DG/DBMS 的操作环境

图 1—3 表明了 DG/DBMS 是如何在 ECLIPSE 计算机的先进操作系统 (AOS) 的控制下运行的。

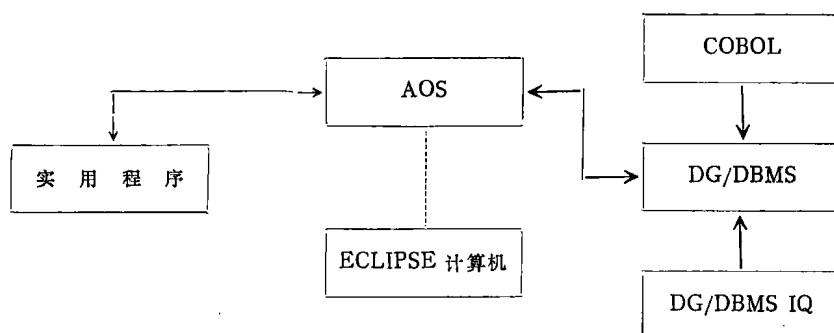


图1—3 DG/DBMS是AOS管理下的一个进程

AOS 本身的特性使得它非常适合于在联机多终端的环境中与大规模数据库有关的应用。AOS 是个灵巧的、多道程序设计的操作系统，担负着程序控制，输入输出以及文件存取功能，它同时对分时操作、实时操作和多个批作业流操作实行控制。

DG/DBMS 利用了 AOS 提供的一切优点，也就是说，你可获得如下功能：

- 设计一个能适于各种不同应用方面需要的数据库；
- 具有立即访问存放在数据库中的所有数据的能力；以及
- 使许多用户能同时(并行)访问数据。

如前所述，对 DBMS 说来，数据的完整性、遇到故障时的修复能力、以及研制应用程序的灵活性，都是至关重要的问题。AOS 是通过保密性的数据管理机构，成熟的多道程序设计技术，强有力的系统管理及操作员模块来处理上述问题的。前言中所列出的 AOS 手册中，包含有更多的、关于这个无与伦比的操作环境的知识。

§ 1—3.2 DG/DBMS 的组成部分

一旦运行起来，DG/DBMS 由下列各部分组成(如图 1—4 所示)；

- 整个系统有一个数据库总控 (Monitor) 进程；
- 每一个打开的数据库均有一个数据库控制 (Controller) 进程；
- 数据定义程序 (DDF)；
- 基于 CODASYL 建议的、与 Data General 的 ANSI' 74 COBOL 相联接的接口；

- DG/DBMS IQ 程序;
- 多种实用程序

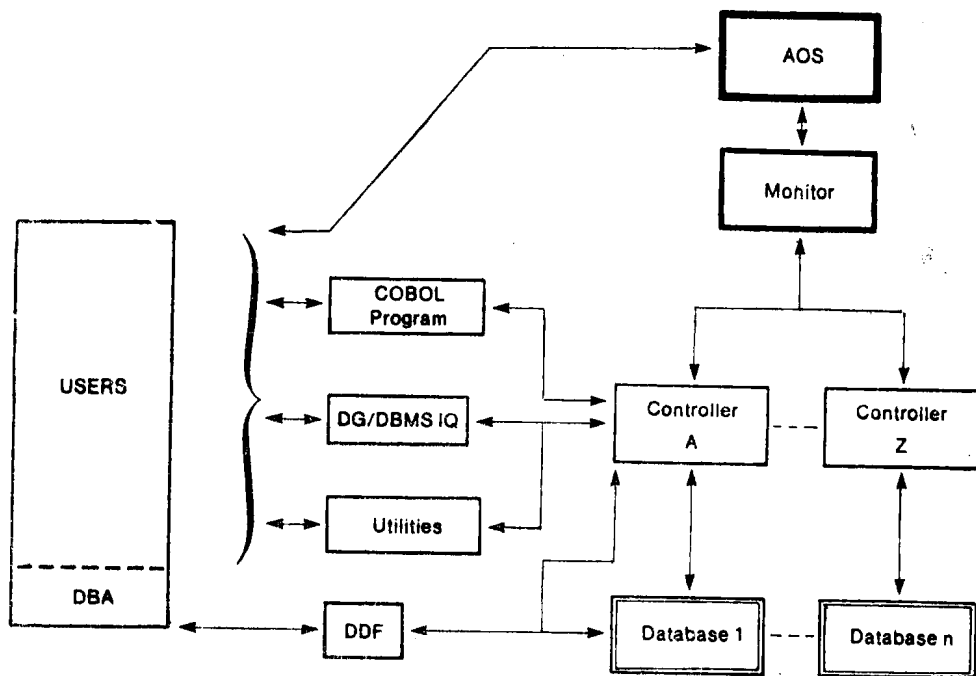


图1—4 DG/DBMS的组成部分

§ 1—3.2.1 总控程序 (Monitor) 与控制程序 (Controller)

基本说来, 总控程序通过启动每一个用户希望打开的数据库, 来初始化系统; 然后, 各控制程序分别处理给定数据库的用户与 DG/DBMS 之间的相互作用。

§ 1—3.2.2 数据定义程序 (DDF)

DDF 是用于描述数据库的数据和结构类型的交互式的实用程序。组成模式和子模式的数据项、记录类型以及系类型都具有一定的参数, 这些下一章要阐述。一旦你已定下这些参数, 你或数据管理员 (DBA) 就要使用 DDF 向系统输入这些数据描述。

DDF 将输入的信息转换为压缩中介数据 (PMD-Packed Metadata); 数据库控制程序就使用 PMD 中的数据描述来访问数据库。

DDF 的另一部份叫做分配程序, 它基于中介数据中的信息来分配足够的空间以存贮各种内部文件。

DDF 产生子模式的源码形式, 事先, 在应用程序中要包含有这种源码、然后再编译和执行。进一步了解 DDF 的知识, 请参阅第三章。

§ 1—3.2.3 COBOL 接口

在用 COBOL 程序访问数据库之前, 必须在程序编译之前就包括一个 DDF 产生出来的子模式, 这样就告诉总控程序, 你希望访问的数据库是哪一个; 还告诉控制程序, 哪些数据项、记录类型和系类型是本程序可以访问和处理的。另外, 使用数据库的每一个程序都具有一个工作空间 (缓冲区)、作为在程序和数据库之间接收和发送一切要传送的数据的区域; 这个

用户工作区 (UWA-User Work Area) 是程序存贮空间的一部份, 它为子模式中定义的每一类记录都单独保持一个位置。当程序访问数据时, 控制程序就把数据带到 UWA 以供处理。

再者, DG/DBMS 要求所有访问数据库的程序都要在处理块边界之内使用 DML 命令。这就保证了数据库不会含有未完成的修改, 这也成为后备和修复机构的基础。第二章更详细地讨论处理块和 DML; 有关 COBOL 接口的进一步知识请看第四章。

§ 1—3.2.4 DG/DBMS IQ

Data General 公司数据库管理系统交互式查询程序 (DG/DBMS IQ) 是与 DG/DBMS 的数据库文件联合一起使用的工具。这是个简单而有用的工具, 它使任何得到适当许可的用户得以访问存放于数据库中的数据。这意味着你可以不必写专门的程序就能够抽取数据库中的信息。本手册第五章稍详细地介绍 DG/DBMS IQ, 更进一步的阐述请看 DG/DBMS IQ 用户手册。

§ 1—3.2.5 其它实用程序

其它有助于简化 DBA 的工作的实用程序有:

- IDML (DML 的交互式的接口) 你可用它作为教学、检测或校核的辅助工具, 也可作为数据库的修复工具。

- Rebuild 在硬件发生故障之后, 你必须使用这个实用程序来使数据库恢复到故障发生之前时刻的状态。

- 操作员与总控程序的接口 它允许你设定或检查数据库的状态参数、终止数据库控制程序、打开与关闭数据库, 以及操纵实用程序 Rebuild (严重故障修复程序)。

DG/DBMS 参考手册详细介绍了使用这些手册的知识。

§ 1—3.3 基本情况与轮廓外貌

你将会发现, DG/DBMS 拥有的能力及限制使你在设计数据库结构时具有最大的灵活性。举例说来, 根据所在单位的规模和构成情况, 你可能希望生成不止一个数据库。DG/DBMS 就能够容纳任意数量的数据库, 而且可以在同一时间内使用多达 16 个不同的数据库; 在每一个数据库中 DG/DBMS 还允许存贮和使用多达 65 千兆 (Billion) 字节的数据。

DG/DBMS 允许每一数据库最多可有 255 个记录类型, 而每一记录类型的记录值数可为任意数; 每个记录类型最多可以有 512 个数据项, 任一数据库所具有的数据项的最大数目为 4096; 对每个数据库还可以定义最多 255 个系类型。

§ 1—4 小 结

DG/DBMS 支持 CODASYL 关于数据库设计的概念, 这意味着 DG/DBMS 用记录类型、系类型与数据项来构成模式和子模式。DG/DBMS 通过集中进行数据描述, 使你能把精力专注于数据处理而不是数据的存贮和访问的方案上。

DG/DBMS 还包含有专门的机构来处理 and 修复硬件和软件故障。以下各章, 我们详细说明本章提及的 DG/DBMS 的特点, 然后再说明如何将所有这些概念和要点都协调地联系起来。