

745058

系列之一

ORACLE 关系数据库管理系统
用 户 手 册
(5)

SQL * Report 用户指南

79

《计算机技术》编辑部

目 录

前言	(1)	将正文在列内对准.....	(12)
第一部分 SQL * Report 介绍		移向一新行.....	(12)
第 1 章 引言.....	(4)	列正文的自动对准.....	(12)
欢迎使用 SQL * Report	(4)	表的结束.....	(13)
概述	(4)	RPF 的更进一步的概念	(13)
第二部分 报告正文格式化 程序-RPF		列.....	(13)
第 2 章 RPF 概述	(6)	省缺表.....	(14)
引言	(6)	在省缺表内定义表.....	(14)
RPF 处理	(6)	表与列.....	(15)
语法	(6)	定义两列表.....	(15)
RPF 概念	(6)	命令的非正常结束.....	(16)
RPF 版本 5.1	(7)	\ 的使用.....	(16)
第 3 章 报告格式化程序的使用.....	(7)	强制或删除多余的控格.....	(16)
例 1: 一封简单信件	(7)	\ 的打印.....	(16)
使用 RPF 的四个基本步骤 ...	(7)	打印 RPF 命令	(16)
定义一个 RPF 表	(8)	RPF 命令一览表	(17)
启用一个 RPF 表	(9)	APN 交错页号	(18)
送入正文和其它格式化命令	(9)	B 空格	(18)
结束一个 RPF 表	(9)	CEN 居中	(18)
其它的格式化命令	(9)	CL 列直接量	(18)
省缺表	(9)	CS 列跳越	(19)
定义页: 确定上、下边界 ...	(9)	CUL 居中, 加底线	(19)
控制纵向空格	(9)	DT 定义表	(19)
控制行走	(9)	F 图	(20)
定义段.....	(10)	FR -对齐.....	(20)
运行 RPF	(10)	HS -水平间隔.....	(20)
例 2: 一份表格报告	(10)	I -回缩L-直接量	(20)
定义表 1: 确定总边界	(10)	N 新行 (下行)	(21)
定义列.....	(11)	N 新列 (下列)	(21)
居中和底线.....	(12)	NP 新页 (下页)	(21)
嵌套表.....	(12)	P 分页	(21)
向各列送入数据.....	(12)	PAGE 定页	(21)
		PAUSE 暂停	(22)
		R -右对齐.....	(22)
		RR -右不齐.....	(22)
		S 跳行	(22)
		SP 垂直间隔	(22)

SPN	开始页编号	(22)	报告的结构.....	(26)
T	启用表	(22)	RPT 版本 5.1	(27)
TE	表结束	(23)	新命令.....	(27)
TTL	定标题	(23)	标识符的兼容性处理.....	(27)
TTLU	标题加底线	(24)	出错信息处理.....	(28)
UL	底线	(24)	报告语法结构的预校验.....	(28)
VS	垂直间距	(24)		
执行 RPF	(24)	第 5 章 RPT 举例	(28)	
RPF 开关	(24)	例 1: 一封样本信件	(28)	
A-全黑体	(24)	例 2: 一个表格报告	(29)	
B-黑体	(24)	报告程序.....	(29)	
D: D-打印设备为 Diablo 型		执行报告程序.....	(39)	
打印机.....	(24)	例 3: 嵌套报告	(40)	
D: V-打印设备为 VT100 ...	(25)			
F- (FormFeed) 换页	(25)	第 6 章 SQL * Report		
I-预置空页	(25)	的语言语句.....	(51)	
P: N: M-页区间	(25)	语句的类型.....	(51)	
R-颠倒下划线的次序	(25)	语法结构.....	(51)	
S-假脱机至行式打印机	(25)	RPT 语句摘要	(51)	
U-大写	(25)	说明性语句.....	(51)	
W-暂停	(25)	宏定义语句.....	(51)	
		宏执行语句.....	(51)	
		程序控制语句.....	(52)	
		算术语句.....	(52)	
		其它各语句.....	(52)	
		附录.....	(68)	

第三部分报告生成程序-RPT

第 4 章 RPT 概述	(26)
引言.....	(26)
报告生成过程.....	(26)
执行 RPT	(26)

图例一览表

图 1. 报告生成过程	(5)
图 2. 使用 RPF 生成格式化文件	(7)
图 3. 一封简信-RPF 输入	(8)
图 4. 一封简信-RPF 输出	(10)
图 5. 表格报告-RPF 输入	(11)
图 6. 表格报告-RPF 输出	(13)
图 7. 例 1-省缺表	(14)
图 8. 例 2-省缺内的表	(15)
图 9. 例 3-当前列内的两列表	(15)

图 10.RPF 命令概述 (APN-F)	(17)
图 11.RPF 命令概述 (FR-PAUSE)	(17)
图 12.RPF 命令概述 (R-VS)	(18)
图 13.例 1-样本信件的输出文件	(31)
图 14.例 1-样本信件的程序表 (3 的第 1 部分)	(32)
图 15.例 1-样本信件的程序表 (3 的第 2 部分)	(33)
图 16.例 1-样本信件的程序表 (3 的第 3 部分)	(34)
图 17.例 1-样本信件的程序结构	(34)
图 18.例 2-表格报告的输出文件	(35)
图 19.例 2-表格报告的程序表 (3 的第 1 部分)	(36)
图 20.例 2-表格报告的程序表 (3 的第 2 部分)	(37)
图 21.例 2-表格报告的程序表 (3 的第 3 部分)	(38)
图 22.例 2-表格报告的部分临时文件 (2 的第 1 部分)	(38)
图 23.例 2-表格报告的部分临时文件 (2 的第 2 部分)	(39)
图 24.例 2-表格报告的程序结构	(39)
图 25.例 3-嵌套报告的程序表 (7 的第 1 部分)	(42)
图 26.例 3-嵌套报告的输出文件 (2 的第 2 部分)	(43)
图 27.嵌套报告的程序清单 (7 的第 1 部分)	(43)
图 28.嵌套报告的程序清单 (7 的第 2 部分)	(44)
图 29.嵌套报告的程序清单 (7 的第 3 部分)	(45)
图 30.嵌套报告的程序清单 (7 的第 4 部分)	(46)
图 31.嵌套报告的程序清单 (7 的第 5 部分)	(47)
图 32.嵌套报告的程序清单 (7 的第 6 部分)	(48)
图 33.嵌套报告的程序清单 (7 的第 7 部分)	(49)
图 34.例 3-嵌套报告的程序结构	(50)
图 35.报告语句结构	(58)
图 36.嵌套的报告结构	(59)
图 37.报告内多个报告	(60)
图 38.间断报告	(61)
图 39.样本 GOTO 程序	(63)
图 40.样本 IF 语句程序	(64)
图 41.例 1: 程序清单 (4 的第 1 部分).....	(69)
图 42.例 1: 程序清单 (4 的第 2 部分).....	(70)
图 43.例 1: 程序清单 (4 的第 3 部分).....	(71)
图 44.例 1: 程序清单 (4 的第 4 部分).....	(72)
图 45.例 1: 样本报告 (2 的第 1 部分).....	(72)
图 46.例 1: 样本报告 (2 的第 2 部分).....	(73)
图 47.例 2: 程序清单 (7 的第 1 部分).....	(74)
图 48.例 2: 程序清单 (7 的第 2 部分).....	(75)

图 49.例 2: 程序清单 (7 的第 3 部分).....	(76)
图 50.例 2: 程序清单 (7 的第 4 部分).....	(77)
图 51.例 2: 程序清单 (7 的第 5 部分).....	(78)
图 52.例 2: 列表 (7 的第 6 部分).....	(79)
图 53.例 2: 程序清单 (7 的第 7 部分).....	(80)
图 54.例 2: 样本报告	(81)
图 55.例 3: 程序清单 (4 的第 1 部分).....	(82)
图 56.例 3: 程序清单 (4 的第 2 部分).....	(83)
图 57.例 3: 程序清单 (4 的第 3 部分).....	(84)
图 58.例 3: 程序清单 (4 的第 4 部分).....	(84)
图 59.例 3: 样本报告 (2 的第 1 部分).....	(85)
图 60.例 3: 样本报告 (2 的第 2 部分).....	(85)
图 61.例 4: RPT 日期与 ORACLE 日期的关系 (2 的第 1 部分)	(86)
图 62.例 4: RPT 日期与 ORACLE 日期的关系 (2 的第 2 部分)	(87)
图 63.例 5: RPT 日期变量 (2 的第 1 部分)	(88)
图 64.例 5: RPT 日期变量 (2 的第 2 部分)	(88)

前 言

目的

本指南描述 SQL * Report 的使用方法。这是一套报告生成程序，基本上由两个 ORACLE 实用程序所概括，即

RPF ORACLE 的报告正文格式化程序，用于控制报告最终格式

RPT ORACLE 的报告生成程序，用于将存在 ORACLE 数据库中的信息纳入用户的信函和报告中。

本手册有两大目的：作为使用者的指南，它将说明 RPF 和 RPT 是如何工作的，并将演示它的使用过程；作为一份详细的参考手册，它将提供 RPF 和 RPT 的语法和命令的描述。本材料提供了所产生和所使用的文件的例子，并以图解方式展示其使用过程。

版本 5.1 的改变在“RPF 概述”和“RPT 概述”两章中有概要的说明。

读者

这个指南是为 ORACLE 关系数据库管理系统的这样一些使用者写的，他（她）们需要简单的正文格式化程序，并且将从 ORACLE 数据库中提取的数据纳入自己的文档之中。

使用 ORACLE 的正文格式化程序（RPF）并不需要什么准备知识。使用者只要会用自己的所使用的计算机系统上的编辑程序就够了。ORACLE 报告生成程序（RPT）的使用则要求熟悉 SQL 和 ORACLE RDBMS。尽管一般说来，对程序设计技巧不作要求，但如果想深入而完整地使用该软件产品，有一定的程序设计技巧将是很有用的。

本材料的组织方式

这个手册包括三大部分和一个附录，即有如下内容：

第一部分：SQL * REPORT 介绍

第 1 章是对 SQL * Report 的介绍，并且说明它的两大部分，即 RPF 和 RPT 的关系。

第二部分：报告正文格式化程序 RPF

第 2 章为 RPF 的概述。

第 3 章说明如何使用 RPF 去对简单信函和表格报告格式化。

第 4 章给出 RPF 语法和命令的完整描述

第三部分：报告生成程序 RPT

第 5 章为 RPT 的概述。

第 6 章说明如何使用 RPT 以嵌入在正文中的数据库信息来准备表格报告、嵌套报告及信函。

第 7 章给出了 RPT 语法及语句的完整描述。

附录

这个附录提供了使用 RPF 和 RPT 的五个实例。

如何使用本手册

如果使用者从前未用过 SQL * Report, 那么在使用 RPT 之前必须依次了解本手册的内容。如果是已熟知本手册的内容, 只是想深入了解某些语句或命令, 则可使用索引直接找到相应的章节。

版本 5.1 的改变在“RPF”及“RPT 概述”两章中作了简要说明。

本手册所使用的约定

本手册使用了以下约定:

文件名

文件名以大写字母出现, 如 INIT.ORA.

文件名中可能变化的部分以小写字母出现, 如 SGADEFx.ORA 保留字与关键字。

保留字与关键字均以大写字母出现在例子及正文中, 这表明它们应以大写方式键入, 而且它们在 ORACLE 中有具体规定的含义。

键名

键名以大写方式出现并括入方括号, 如 [RETURN]。

命令的语法结构

命令

命令的开始部分用来标识必须严格以原样键入的正文, 如: SELECT *
FORM

变量

变量以斜体出现。用户必须以适当的值置换变量。

arg1

变量也可以出现在尖括号内, 如 <arg1>。

多中择一项目

多中择一性选择永远以垂直线分离。如果从多个项目中选择一个, 则以花括号括出; 如果该项目可以不出现, 则以方括号括出 (见下面关于必须与可选项的表示约定。)

必须项目

必须项目以花括号括出。用户必须从多个项目中选定一个项目

.DEFINE{ <macro1>|<macro2> }

可选项目

可选项目以方括号括出之, 如

.FPRINT[spacing]<variable>

重复项目

以省略号表示在紧接它之前的结构可以重复出现任意多次。

.FPRINT[spacing]<variable> [[spacing][<variable>]]...

新内容或经修正的内容

新增加的、或经修正的材料以在边界处的竖线标示出 (详见本节的说明)。下述符号出现在命令格式中时, 均以右边那列的形式键入。

句号	.
逗号	,
连字符号	-
分号	;
冒号	:
等号	=
后斜线	\
单引号	'
双引号	"
井号	#
括号	()

与本材料有关的出版物

在使用这份材料的过程中，读者可能需要参阅由 Oracle 公司出版的下述材料。凡购买了软件产品的用户，都会得到有关的资料清单；当然不必要也不会需要全部资料。及时反映软件产品最新变化的更新注释随时都会出现。

有关 ORACLE RDBMS 的出版物计有

.ORACLE RDBMS 发行说明 ORACLE part No.3001

.对 SQL ORACLE part No.3801 的

ORACLE 概述与介绍

.ORACLE 数据库管理员指南

.ORACLE part No.3601

.ORACLE 实用程序用户指南

ORACLE part No.3605

有关 SQL * Report 的出版物有

.SQL * Report 发行说明

ORACLE part No.3012

对于支持 ORACLE 的每个操作系统，还应提供安装指南与用户指南，如：

.用于 DEC VAX / VMS 环境的 ORACLE 安装与用户指南 ORACLE part No.1001

.用于 IBM VM / SP 环境的 ORACLE 安装与用户指南 ORACLE part No.1003

第一部分 SQL * Report 介绍

第 1 章 引 言

欢迎使用 SQL * Report

SQL * Report 将正文格式化功能与 SQL 的询问功能结合了起来。这种强有力的能力使得用户能使用存在 ORACLE 数据库中的信息以及任何其它希望出现的正文来生成各种各样的文档报告。

SQL * Report 的使用潜力极为广泛。在最简单的情形下，它可以生成最普通的信件和表格文件报告。也可以产生一份报告，其数据来自一个 ORACLE 表，有一定的列标头形式，表中还可以有小计与合计数据。

SQL * Report 能生成的更为复杂的文档包括具有多级嵌套的报告、分离性报告、列中可有多处中断以及各种各样的小计与总计。还可以写出过程语言宏，能够强调指出 SQL * Report 正文处理能力的的应用包括有将从 ORACLE 数据库中导出的信息与姓名，地址等信息交叉而产生计算机生成的公函信件，或者周期性的预算和产品价格报告。SQL * Report 还可以用于填写预打印的格式。

概述

SQL * Report 由两个实用程序组成，这两个实用程序协同使用即可从数据库中导出信息并将这些信息以所希望的方式组成报告。报告生成程序 (RPT) 从数据库 (经由 SQL 语句) 取得数据；报告正文格式化程序 (RPF) 则根据正文中的命令将这些数据信息格式化。

通过准备一个称作报告控制文件的文件，就可以生成一份文档。这个报告控制文件中将包括为 RPT 所使用的 SQL * Report 语句、为 RPF 所使用的正文格式化命令以及其它将在最终报告中出现的正文材料。使用者可以经由任何标准的正文编辑程序来生成这个报告控制文件。

一旦完成了报告控制文件，就可以给出命令去处理该文件并生成所希望的文档报告。整个工作分两步完成：

- 首先执行 RPT，产生一份临时文件

- 然后，经由 RPF 处理该临时文件，以产生最终的文档报告。

RPT 读报告控制文件以便扫描 SQL * Report 的诸语句。这些语句将会让 RPT 打印指定的 ORACLE 数据，以便为要生成的报告准备信息，为导出所希望的数据，报告程序中包含了所需要的 SQL 询问，有种种语句可使 RPT 得到报告的头、底信息，使用特定的数据输出格式或者按一定条件转去执行其它的 SQL 语句或 RPT 语句。RPT 可以完成在数据库信息中夹入 RPT 命令和用户所提供的正文的功能。

RPT 处理完成后，临时文件就生成了。这个临时文件包含由用户所提供的正文从数

据库中所抽取的信息以及指出这些信息在最终的报告中如何格式化的 RPF 命令。这种临时文件不可单独使用，必须由报告格式化程序 RPF 来处理，才能生成所希望的报告。

本手册在后面将介绍利用 SQL * Report 产生文档报告的各种实例。下面的图 1 说明了 RPT 和 RPF 产生最终报告的过程。

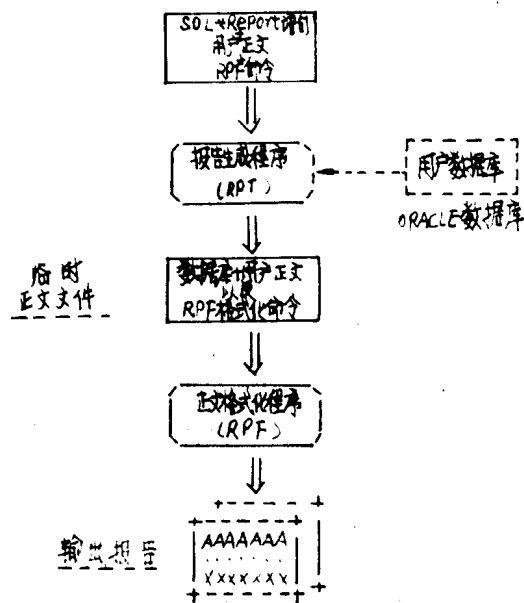


图 1 报告生成过程

第二部分 报告正文格式化程序—RPF

第2章 RPF 概述

引言

报告正文格式化程序 (RPF) 是一个通用程序, 可以对各种字处理应用, 包括新闻报告、备忘录和报告格式化。使用者可将 RPF 命令置入自己文档的正文中以便控制信息的格式。

RPF 命令可以指定与说明如下内容:

.水平和垂直两个方向的边界

.居中和底线

.制表

.页编号

.正文的间距与实际布局

.针对行式打印机、终端或 CRT 等输出设备而产生的输出。

用户在使用 RPT 程序设计语句之前必须先熟悉 RPF。请留意, 尽管 RPF 可单独使用, 但 RPT 则必须与 RPF 联用 (或与其它格式化语言之一联用)。

RPF 处理

RPF 在同一文件中既接受它所处理的输入, 又接受控制处理过程的命令。因此, 嵌置于要格式化的文件中的这些命令将用来控制格式化的操作。使用者通常借助于正文编辑程序以手工方式将这些命令插入到要被格式化的文件中, 因而 RPF 就可以使用这些命令了。如果通过某个程序来构造这个文件, RPF 命令也可以由该程序实施插入。这一过程说明于图 2。

语法

所有 RPF 命令或者以 . 或者以 # 而开始。可以交替使用这两个符号而不影响命令的含义。如果命令拼错了, 尽管是以 # 或 . 开始, 系统均视这些“命令”为正文 (而不看成命令!) 如果想将合法的命令作为正文打印出来, 则应以符号 \ 前置于该命令。例如, #B 是一个插入一空行的命令, 但 \#B 将作为正文对待之, 其输出是 #B。

命令可以大写字母写出或者以小写字母写出。命令与相联系参数可以一个或多个空格、制表符 (tab) 或换页符 (fromfeed) 相分离。具有可变多个参数的命令或对一组字实施操作的命令, 均以单个句号 (.) 或井号 (#) 结束。

RPF 概念

RPF 视正文为字的序列。每个“字”由一个或多个字符组成, 每个字或以至少由一个空格组成的符号或以一个行结束符结束。RPF 将每个字一个接一个地置入到输出行中。每键入一个回车, 就有一个行结束符号进入正文文件。不论使用者在输入的字与字之间送

入了多少个空格在输出的字与字间只用一个空格就可分离。空、制表符、回车、换页符和换行都是分隔符，这些符号对于字在 RPF 输出中的布局格式没有影响。

如果不作说明，则送入的正文是右对齐的。这就是说，当一行占满而下一个字不送入时，它所包含的字将在行中向右对齐。如果不希望右对齐，亦可断离右对齐。每个字只能在一行内，不能分开占据一行以上。RPF 继续处理，直到输入文件结束时为止。

RPF 允许使用者定义输出行的开始位置与结束位置。正文在行中所占的区域叫列(column)。如果不指出列的边界，RPF 将把正文置到省缺位置上，即起点位置为 1 而终点位置为 255。一个报告的最大行宽为 255 个字符。

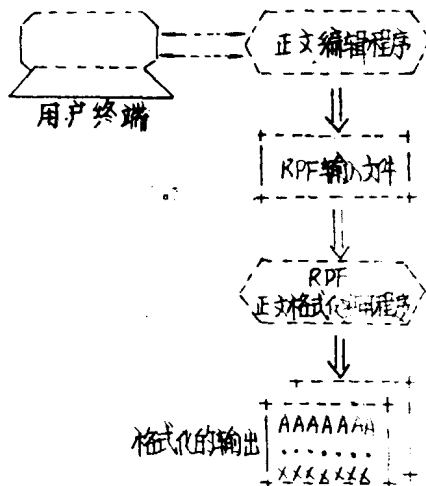


图 2 使用 RPF 生成格式化的文件

RPF 版本 5.1
本指南业已经过更新，以反映出对版本 5.1 对报告正文格式化程序所作的某些更动以及对本手册内容的若干修正。这些更动和修正包括：标题、对准、命令和非正常结束、W 开关以及 SP 命令。

第 3 章 报告正文格式化程序的使用

例 1：一封简信

让我们来说明一下，为产生一封简信而需要的输入文件究竟是什么样的一种文件。先得花点时间说明一下在图 3 和图 4 中反映的样本文件。图 4 是要生成的一封简信，图 3 是为产生这封信而需要的文件清单（输入文件）。

这个输入文件是使用标准正文编辑程序产生的。应该说明的是，这里的行号是为了在以后的讨论中便于说明而补加上去的。

注释：RPF 命令以句号(.)或井号(#)开头，用户的正文则不带有这两个开始符号。命令或用大写字母或用小写字母指明。为了区分 RPF 命令与 RPT 语句，本手册中所出现的所有 RPF 命令均以井号(#)开头。

使用 RPF 的四个基本步骤

为了使用 RPF 格式化程序，必须将正文以格式化窗口或以由一至多个列构成的“表格”形式送入。例如，图 4 中那封信在 RPF 中只是作为仅由一列构成的一个“表格”而定义的。这封信的全部正文就包含在这个单列中。

注意，RPF 表与 ORACLE 表不同，RPF 表只是一个具有特定长度与宽度的矩形格式的窗口。我们以后将会看到，这个表的宽度决定于命令 DT；这个表的长度决定于所选

择的数据库记录的个数以及由用户所直接送入的正文的总量。

对象这样一个表格式化, 需要经历四步:

1. 定义该表
2. 启用该表
3. 送入正文及各种格式化命令
4. 结束该表

现在我们以图 4 中所说明的那封信为例, 看看下述四步是如何运用的。图 3 说明用来生成该信的 RPF 文件表。

定义一个 PRF 表

图 3 中第一行的 DT 命令用于定义该表:

#DT 1 13 73 #

1:#DT 1 13 73 #

2:#page 6 58

3:#SP 1

4:#T 1

5:#S 3

6:January 10,1985

7:#S 3

8:Mr.Willian O.Smith#N

9:3752 Oakwood Drive #N Scaport,Mc. 96142

10:#S 1

11:Dear Bill:

12:#S 1 #P

13:I hope you have received the medical and dental insurance forms I sent.

14:#S 1

15:#P The form you submitted for your November 1,1984 visit to Dr.George

16:Smedley did not indicate the nature of the treatment. Please have the

17:doctor write me a note describing the reason for the visit and the

18:treatment which was prescribed.

19:#B #P

20:If you have any questions please give me a call at my office.

21:#S 1

22:Sincerely.

23:#S 3

24:Sharon Brown

25:#TE

图 3 一封简信-RPF 输入

第一个#号标志命令的开始, 在 DT 后面的数 1 指出表的名字 (表的“标识号”)。在一个 RPF 文件中最多可定义 30 个 RPF 表, 但必须对每个表指定一个标识号 (从 0 到

29)。

表标识号后面的一对数指定这个单列表的边界：13 为左边界，73 为右边界。每个 RPF 表最多可以包含 255 列。当列数超过 1 时，对每列都应指出确定其左右边界的一对数。

和很多 RPF 命令一样，DT 是一个“封闭”命令，必须以一个句号(.) 或井号(井) 结束。上述 DT 命令以井结束，它表明这个表只有一列，从而结束了整个表的定义。

启用一个 PRF 表

仅仅定义了表并不能自动地将正文置入于所指定的列边界内。在实施格式化前，必须先启用相应的表。第 4 行的 T 命令，井T1，就是启用表 1 的命令。注意，被启用的表的标识号紧接着命令 T。

送入正文和其它格式化命令

现在可以送入这个表应该包含的实际正文以及其它 RPF 格式化命令了。在图 3 中，正文和格式化命令位于 6-24 行中，其中的命令将在下面进一步说明。

结束一个 RPF 表

对一个 RPF 表格式化的最后一步是指出一个 TF 表结束命令（如图 3 中第 25 行）。这个命令告诉 RPF：表 1 已不再处于活化状态了。列边界又回退到省缺表（见下面的说明）。

其它的格式化命令

图 3 中包含了一些 RPF 格式化命令，这些命令将用于控制正文在已定义的表中的布局。下面就这些重要的命令加以说明。

省缺表

首先注意，RPF 具有某些省缺值，在用户指定新值前，这些省缺值一直起作用。例如，若用户的正文在第 4 行前已置入样本文件，则这些正文就按照省缺表的定义格式化。省缺表是一个单列表，起始位置为 1，终止位置为 255。

定义页：确定上、下边界

第 2 行的命令 PAGE 用于定义一页的上、下边界。RPF 在每页的第 6 行后放置正文，在放满 58 行后自动地跳到下一页。每页假定有 66 行。如果打印间距是 6 行/英寸，则 66 行将占据标准的一页（共 11 英寸）。

控制纵向空格

SP 命令用来控制行与行间的距离。由于省缺值是单间距，所以这个命令可以省略。第 5 行和第 7 行的命令 S3 将使系统跳过 3 行，当按规范格式打印一封信时，为了使得正文的第一行处于应有的位置，必须跳过若干行再打印。

空行命令 B 的作用同 S1，即插入一空行。

控制走行

看一下第 6 行，这里包含了用户正文的第 1 行。这行由三个字构成：“January”、“10”（后接一逗号，）和“1985。”请记住，输入文件中字与字间空格的个数对于输出的间距没有影响。这些字在其它行上可能已有输入；这时它们的形式不变。

现在看看第 8 行，它包含地址名，后接一走行命令 N。这个命令与打字机的返回键 (ReTurn) 同作用。第 9 行的 N 命令使得城市名与州名打印在街道地址的下一行上。

定义段

第 12 行上的分段命令 P 将使得它后面的正文在下一行上回缩五个空格后打印出来。信的主体在列边界之内打印。在这个例子中, 字“insurance”将不会与字“dental”位于同一打印行上。因此, “dental”应与右边界对齐, 余下的空间应均匀分布在同一行的诸字之间。

运行 RPF

为了运行 RPF, 需要键入下述格式的命令: RPF <文件名>[<输出文件名>][<设备名>][<开关>]

于是, 使用下述系统命令

```
RPF LETTER.ABC TT1:-I
```

将能在指定的打印设备上打印出同一封信, 其中 TT1: 表示所指定的打印设备的系统地址, 前面的 LETTER.ABC 是输入正文文件的名字。如果文件的限定符 ABC 省略, 就假定以 RPF 为限定符。

January 10, 1985

Mr. William O. Smith

3752 Oakwood Drive

Scaport, Mc.96142

Dear Bill:

I hope you have received the medical and dental insurance forms I sent.

The form you submitted for your November 1, 1984 visit to Dr. George Smc-
dldy did not indicate the nature of the treatment. Please have the doctor write me
a note describing the reason for the visit and the treatment which was prescribed.

If you have any questions please give me a call at my office.

Sincercrely,

sharon Brown

图 4 一封简信-RPF 输出

例 2: 一份表格报告

使用 RPF 的第二例子是在下面将要说明的表格报告。图 5 所展示的是产生图 6 所报告的输入正文文件。在深入阅读以后的讨论之前先得花点时间说明一下这些文件。同样, 这里的行号是为了讨论与说明的方便补加上。

定义表 1: 确定总边界

所定义的第一个表将设置页的总的左、右边界。在这个例中, 第一行的命令 DT 就是用于定义表 1 的总边界, 它表明这是一个单列表, 左、右边界分别为 13 和 73。这是本报告的水平边界。以后所定义的任何表都将“嵌套”在这个列边界之内。

```

01:##DT 1 13 73 #
02:##DT 2 15 9 28 32 39 42 49 52 0 #
03:##page 6 58
04:##T1
05:##S 4
06:##CUL " ALL SEASONS SPORTING GOODS" #
07:##S 2
08:##CUL MONTHLY SALES REPORT #
09:##S 4
10:##T 2
11:##ITEM #N NO.#NC
12:##CEN ITEM # #N #CEN DESCRIPTION # #NC
13:##R PREVIOUS #N MONTH #NC
14:##R CURRENT #N MONTH #NC
15:##R Y-T-D #NC
16:##S 2
17:2354 #NC NFL Footballs #NC $ 175.34 #NC $ 202.45 #NC $ 564.89
#NC
18:##S 1
19:6734 #NC Chicago Cubs Baseball Uniforms #NC $ 56.10 #NC $ 162.38
#NC $ 287.01 #NC
20:##S 1
21:8940 #NC Alpine Skis #NC $ 941.84 #NC $ 1005.93 #NC $ 3582.57
#NC
22:##S 1
23:##NC #NC-----#N $ 1173.28 #NC-----#N $ 1307.76
#NC-----#N
24:$ 4434.47
25:##TE ##S 3
26:##CUL END\OF\REPORT #
27:##TE

```

图 5 表格报告-RPF 输入

定义列

第 2 行定义表 2。这个表包含 5 列：第 1 列的边界从 1 到 5，第 2 个从 9 到 28，第 3 个从 32 到 39，如此等等。注意，第 5 列和最后一列是由最后一对数 (52, 0) 定义的。这里的 0 指出这一列的右边界在启用本表那一列内的右边界上。稍后，我们将会看到如何使用表 2 生成这个表格报告。

居中和底线

第 3 行确定该输出页的上、下边界而第 4 行启用表 1。由于表 1 只有一列，其后的正文将在该列的边界内被格式化。

第 6 行的 CUL 命令指出，处于该命令与命令结束符间的正文将在当前列内居中并定底线。由于当前列占据从 13 到 73 的位置，所以这一正文将居中于本页上。第 8 行上的命令 CUL 将产生居中和带底线的正文的第 2 行。

嵌套表

现在就可以生成表格报告了。该报告将包含跨越页的五个列。如果这个报告是用打字机打出的，用户大概对每列的第一个位置将设置一个停标志 (tab stop)。但使用 RPF 时，则必须启用一个新表以建立这些列边界。第 10 行上的命令启用已在第 2 行上作过定义的表 2。表 1 只有一列。表 2 则不同，表 2 提供五列输出正文。

当有一个表处于活化状态而启用另一个表时，这个新启用的表将细分“当前”列，使之变为新的当前列。新表的边界必须置于第一个表(“父表”)的当前列之内。如果第一个表在第二个表起用前结束，则对第二个表的边界没有这种限制。

这里，表 1 的当前列就是那唯一的列。因此，启用表 2 将会使从 13 到 73 的 61 个打印位置细分为五个独立的区域(或列)。

一个表的列边界是相对于当前列来解释的。例如，表 2 的第 1 列从相对位置 1 开始至相对位置 5 结束。由于表 1 的当前列从位置 13 开始，所以表 2 第 1 列的绝对边界是从 13 到 17。最后一列从相对位置 52 (绝对位置 64) 开始，以当前列的端点结束 (绝对位置 73)。

嵌套表必须完全置于所启用的列内。启用表 2 的列包含 61 个打印位置。如果表 2 的任何一列超出了相对位置 61，系统将指出一个错误。例如，最后一列曾定义作 52 65，因此在启用表 2 时 RPF 将返回一个出错信息：“Table width exceeds containing column width” (见第 10 行)。

向各列送入数据

在定义和启用表 3 之后，就可以将输出正文置入五个列的每个列中。第 11 行的正文将置入第 1 列。命令 N 将使得正文“NO”置于本列的下一行上。命令 NC 将停止对当前列布局正文。此后的正文将置于表 2 中下一列的第 1 行上。

将正文在列内对准

第 12 行上的 CEN 命令要求字“ITME”和字“DESCRIPTION”以第 2 列的前两行为中心。第 13、14 和 15 行上的命令 R (右对齐) 指出：对于 3、4、5 列的所有正文行均应在列内右对齐。注意，各列的列标头均向该列的右边界对齐。

移向一新行

第 16 行上的命令 S2 也将具有同样的效果并将跳过两行。第 17 行将数据的第 1 行送入该报告的每一列。注意，最后三列的美元值是右对齐的。第 19 行和第 21 行定义该报告的第 2 行和第 3 行的输出。

列正文的自动对准

正文“Chicago Cubs Baseball Uniforms”将不与“ITEM DESCRIPTION”列置于同一行上。正象前面已经说过的那样，RPF 将尽可能多的字置于某列的一行之内，当一行