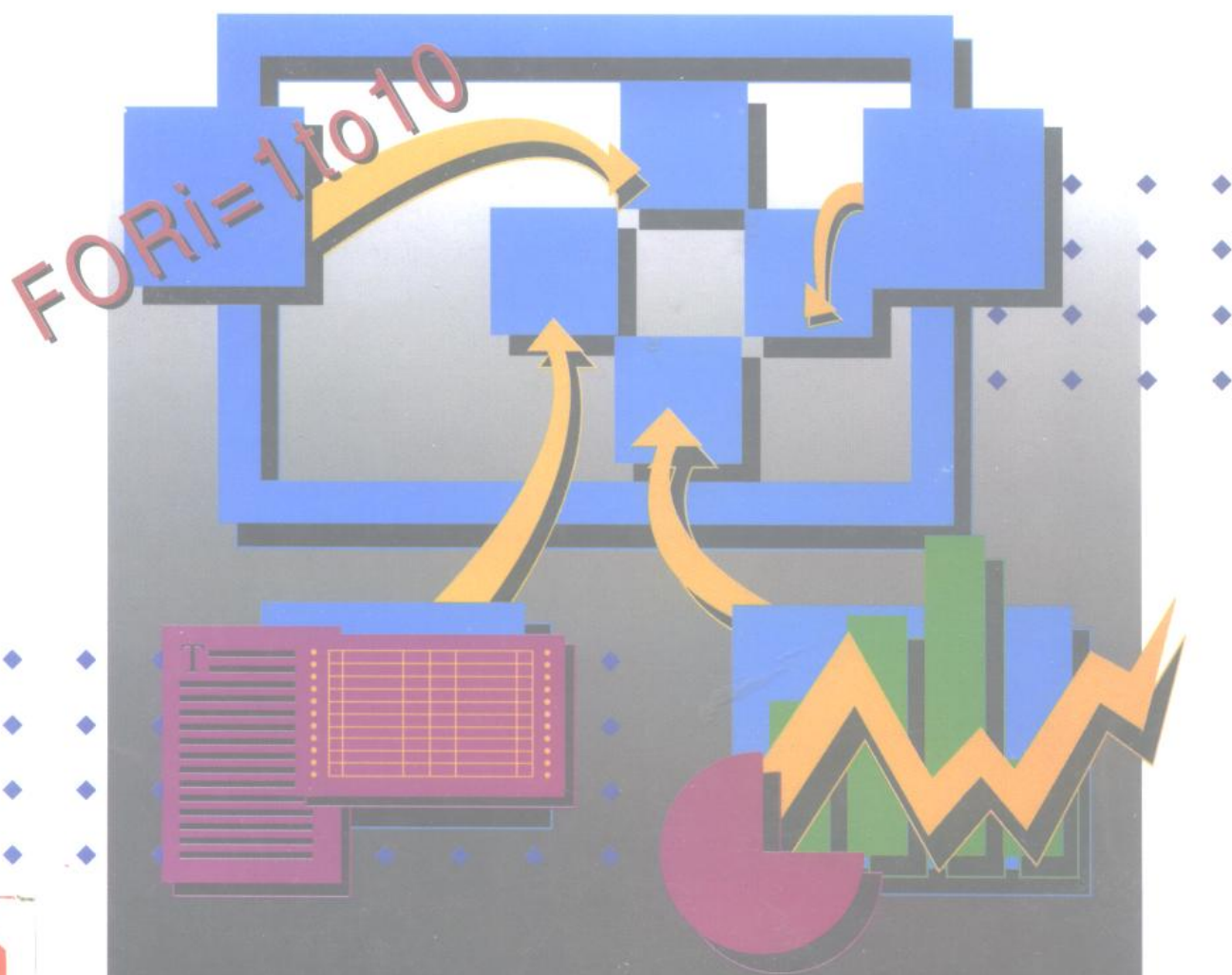


新一代 BASIC CA-REALIZER 使用及编程指南

潘正伯 车克健 主编



電子工業出版社

PUBLISHING HOUSE OF ELECTRONICS INDUSTRY

URL: <http://www.phei.co.cn>

新一代 BASIC CA-REALIZER 使用及编程指南

潘志雄 王光雄 王瑞 主编



新一代 BASIC—— CA-REALIZER 使用及编程指南

主编 潘正伯 车克健

编者 (以姓氏笔划为序)

陈宏志 陈传绂

耿国庆 鲁俊生

韩莹洁 瞿 燕



电子工业出版社

内 容 提 要

本书详细介绍了第三代 BASIC—CA-REALIZER 语言。该语言可同时用在 Windows 和 OS/2 两种环境下,仍采用我国计算机用户熟悉的过程模式,同时又吸收了事件驱动模式,并与 Quick BASIC 高度兼容。

全书共分为七章,分别介绍了使用 FormDer 作 GUI 开发、数据与运算、过程与函数、文件、CA-REALIZER 工具、串行通讯与错误捕获、CA-R 高级功能等。对掌握 BASIC 的用户迅速进入 Windows 和 OS/2 的开发领域及实践事件驱动模式会有极大帮助。

JS/194/10

书 名:新一代 BASIC——CA—REALIZER 使用及编程指南

著 者:潘正伯 车克健

责任编辑:刘文玲

特约编辑:吴 迪

排版制作:电子工业出版社排版室

印 刷 者:北京牛山世兴印刷厂

出版发行:电子工业出版社出版、发行 URL:<http://www.phei.co.cn>

北京市海淀区万寿路 173 信箱 邮编 100036 发行部电话 68214070

经 销:各地新华书店经销

开 本:787×1092 毫米 1/16 印张:20.75 字数:528 千字

版 次:97 年 3 月第 1 版 97 年 3 月第 1 次印刷

印 数:5000 册

书 号:ISBN 7-5053-3834-X
TP·1647

定 价:25.00 元

凡购买电子工业出版社的图书,如有缺页、倒页、脱页者,本社发行部负责调换
版权所有·翻印必究

目 录

引 言	(1)
1. 为什么选中了 CA-REALIZER	(1)
2. CA-REALIZER 的特色	(1)
3. 适用环境及安装	(4)
4. CA-REALIZER 开发环境	(4)
5. 程序结构及开发方法	(6)
6. 本书的安排	(8)
7. 本书使用的标记	(8)
第一章 使用 FormDev 作 GUI 开发	(11)
1.1 概 述	(11)
1.1.1 FormDev 简介	(11)
1.1.2 FormDev 环境	(12)
1.2 使用 FormDev 开发 GUI 程序	(20)
1.2.1 创建和编辑窗体	(21)
1.2.2 编辑对象并设计窗体	(23)
1.2.3 测试、运行窗体	(28)
1.2.4 窗体和项目的存储	(32)
1.3 菜 单	(39)
1.3.1 菜单编辑窗	(39)
1.3.2 添加菜单项	(40)
1.3.3 添加条目	(41)
1.3.4 编辑菜单执行代码	(43)
1.3.5 菜单的存储和装入	(44)
1.4 数据库窗体	(48)
1.4.1 创建数据库窗体	(48)
1.4.2 向窗体中添加字段	(48)
1.4.3 关闭数据库窗体	(49)
1.4.4 示例	(49)
1.5 FormDev 库与用户控制图	(51)
1.5.1 对象库	(52)
1.5.2 窗体库	(52)
1.5.3 菜单库	(52)
1.5.4 用户控制图	(52)
第二章 数据与运算	(54)
2.1 数据类型	(54)

2.1.1 简单数据类型	(54)
2.1.2 复合数据类型	(56)
2.2 变量和常量	(58)
2.2.1 变量	(58)
2.2.2 常量	(61)
2.3 类型说明	(62)
2.3.1 隐式说明与显式说明	(62)
2.3.2 用户自定义数据类型	(62)
2.4 数 组	(63)
2.4.1 数组的一般形式	(63)
2.4.2 数组的构成和连接	(64)
2.4.3 CA-R 独具的特殊数组处理功能	(64)
2.4.4 数组运算	(67)
2.4.5 数组的数组和子数组	(69)
2.4.6 数组边界的限制	(70)
2.4.7 打印数组	(71)
2.5 记 录	(71)
2.5.1 定义记录结构	(71)
2.5.2 说明记录变量	(72)
2.5.3 访问记录中的域	(72)
2.5.4 记录数组	(72)
2.5.5 复合数据结构嵌套中的记录	(73)
2.5.6 提取记录数组的某列	(73)
2.6 家 族	(73)
2.6.1 家族的建立	(74)
2.6.2 参考家族成员	(74)
2.6.3 操纵家族和成员	(74)
2.6.4 家族数组	(74)
2.6.5 复合数据结构嵌套中的家族	(75)
2.7 操作符和优先级	(75)
2.7.1 算术操作符	(75)
2.7.2 比较操作符	(76)
2.7.3 逻辑操作符	(77)
2.7.4 位操作符	(77)
2.7.5 运算的优先级	(78)
2.7.6 对数组进行操作	(78)
2.8 数值计算	(80)
2.8.1 三角函数和指数函数	(80)
2.8.2 整理函数	(81)
2.8.3 随机数函数	(81)

2.8.4 数组说明函数	(82)
2.8.5 使用日期-时间值	(90)
2.9 字符串处理	(95)
2.9.1 文本与 ASCII 码间的转换	(95)
2.9.2 字符串的比较	(96)
2.9.3 截取串的一部分	(96)
2.9.4 替换串中文本	(99)
2.9.5 改变大小写	(99)
2.9.6 插入和删除部分字符串	(99)
2.9.7 串的转换	(100)
2.9.8 格式化串	(102)
2.10 程序流的控制	(103)
2.10.1 语句的形式	(103)
2.10.2 控制结构	(104)
2.10.3 判别结构	(104)
2.10.4 循环结构	(106)
2.10.5 控制结构的嵌套	(108)
第三章 过程和函数	(110)
3.1 CA-Realizer 的程序组成	(110)
3.1.1 过程语句(PROC)	(111)
3.2 过程与函数的定义和说明	(112)
3.2.1 定义过程或函数	(112)
3.2.2 过程的说明	(112)
3.2.3 函数的说明	(113)
3.3 变量的生存域	(115)
3.3.1 全局量	(115)
3.3.2 局部量	(117)
3.4 参数传递	(117)
3.4.1 基本概念	(117)
3.4.2 形式参数与实际参数	(118)
3.4.3 传址与传值	(119)
3.5 过程和函数中的修饰词	(121)
3.6 可变数目参数	(121)
3.6.1 柔性参数表	(121)
3.6.2 获取未定量参数的数目	(122)
3.6.3 未定量参数和修饰词的引用	(123)
3.7 重定义、求值顺序与递归	(126)
3.7.1 过程与函数的重定义	(126)
3.7.2 函数与表达式的求值顺序	(126)
3.7.3 递归	(127)

3.8 模块的建立和运行	(128)
3.8.1 建库	(128)
3.8.2 把模块当过程用	(128)
3.8.3 设置漏沙光标	(128)
3.8.4 使用 EXECUTE 命令	(129)
3.9 综合示例	(131)
第四章 文件	(134)
4.1 标准输入/输出	(134)
4.1.1 标准对话框	(134)
4.1.2 输入框	(139)
4.1.3 输出	(141)
4.2 文件概述	(148)
4.2.1 文件类型	(148)
4.2.2 访问文件的命令和函数	(149)
4.2.3 基本文件概念	(149)
4.3 低级命令和函数	(153)
4.3.1 打开文件	(153)
4.3.2 写文件	(153)
4.3.3 关闭文件	(155)
4.3.4 读文件	(155)
4.3.5 文件指针	(157)
4.4 高级文件命令	(158)
4.4.1 数据的输入/输出	(158)
4.4.2 文件格式	(162)
4.4.3 读写 _Named 和 _Plain 数据	(163)
4.5 二进制文件	(165)
4.5.1 二进制 _Plain 格式	(165)
4.5.2 二进制 _Named 格式	(166)
4.5.3 读写 XBase 数据文件	(166)
4.6 一个例子	(167)
第五章 CA-Realizer 工具	(169)
5.1 概 述	(170)
5.1.1 创建工具	(170)
5.1.2 控制工具	(173)
5.1.3 定义工具过程	(177)
5.1.4 一个工具的实例	(180)
5.1.5 工具的字体和图形	(180)
5.2 窗 体	(182)
5.2.1 窗体对象	(183)
5.2.2 创建新窗体	(184)

5.2.3	控制窗体	(185)
5.2.4	获取有关窗体的信息	(185)
5.2.5	设置窗体中的颜色	(186)
5.2.6	添加窗体对象	(186)
5.2.7	处理和修改窗体对象	(192)
5.2.8	查询对象	(194)
5.2.9	查询对象的值	(194)
5.2.10	使用 Non_MDI 窗体建立工具条,状态条和对象拼板	(195)
5.2.11	窗体处理	(196)
5.2.12	模式窗体	(198)
5.2.13	非模式窗体	(201)
5.2.14	设置背景网格	(203)
5.2.15	设置快捷键	(204)
5.2.16	拖动和放置	(205)
5.3	电子表格	(206)
5.3.1	创建电子表格	(207)
5.3.2	控制电子表格	(209)
5.3.3	控制行、列表头	(210)
5.3.4	修改电子表格的内容	(211)
5.3.5	查询电子表格	(212)
5.3.6	设置电子表格过程	(213)
5.3.7	举例	(215)
5.4	图表	(218)
5.4.1	创建图表	(220)
5.4.2	控制图表的外观	(227)
5.4.3	高级图表功能	(233)
5.4.4	设置图表过程	(235)
5.4.5	打印图表	(236)
5.5	观察板	(236)
5.5.1	在观察板中显示数据	(237)
5.5.2	创建观察板	(237)
5.5.3	修改观察板的内容	(238)
5.5.4	设置观察板过程	(238)
5.6	文本簿	(239)
5.6.1	创建文本簿	(240)
5.6.2	在文本簿中显示信息	(240)
5.6.3	控制文本簿的外观	(240)
5.6.4	获取有关文本簿及其内容的信息	(241)
5.6.5	处理文本簿内的信息	(241)
5.6.6	设置文本簿过程	(242)

5.7 绘图板	(243)
5.7.1 创建图象	(243)
5.7.2 创建绘图板	(244)
5.7.3 选择绘图板	(244)
5.7.4 绘图板绘图	(244)
5.7.5 绘图板中鼠标点取的处理	(249)
5.7.6 缓存状态	(250)
5.7.7 控制重画过程	(250)
5.7.8 命令删除	(251)
5.7.9 综合示例	(252)
5.8 动画	(255)
5.8.1 动画的工作方法	(255)
5.8.2 创建动画	(256)
5.8.3 动画控制	(257)
5.8.4 设置特殊画面	(258)
5.8.5 鼠标点取消息的发送	(258)
5.8.6 如何在动画中使用图形	(259)
5.9 用户菜单和快捷键	(261)
5.9.1 菜单概述	(261)
5.9.2 创建菜单	(262)
5.9.3 选择菜单	(263)
5.9.4 菜单控制	(263)
5.9.5 菜单条目添加	(264)
5.9.6 修改菜单条目	(265)
5.9.7 设置菜单过程	(266)
5.9.8 在菜单中使用 CA-R 菜单命令	(267)
5.9.9 子菜单	(269)
5.9.10 快捷键	(269)
5.9.11 举例	(271)
5.10 定时器	(272)
5.10.1 定时器概述	(272)
5.10.2 引入定时器	(273)
5.10.3 查询定时器及其中的事件	(274)
5.10.4 举例	(275)
5.11 工具综合举例	(275)
5.11.1 CA-R 编程方法概述	(275)
5.11.2 举例	(275)
第六章 串行通讯与错误捕获	(285)
6.1 串行通讯	(285)
6.1.1 串行通讯概述	(285)

6.1.2 CA-R 的串行通讯	(285)
6.2 错误捕获	(288)
6.2.1 错误类型	(288)
6.2.2 错误捕获使用的命令和函数	(289)
6.2.3 错误捕获设计方法	(290)
6.2.4 CA-R 对算术错误的处理	(292)
第七章 CA-Realizer 高级功能	(293)
7.1 操作系统接口	(293)
7.1.1 其他程序的载入	(293)
7.1.2 运行操作系统命令	(293)
7.1.3 CA-R 与载入程序的同步 Windows	(294)
7.1.4 设置系统日期和时间	(295)
7.2 外部过程和函数	(295)
7.2.1 动态链接库 DLL	(295)
7.2.2 说明外部过程	(296)
7.2.3 说明外部函数	(299)
7.2.4 系统 DLL	(300)
7.3 用户控制图	(301)
7.3.1 何谓用户控制图	(301)
7.3.2 说明用户控制图	(301)
7.3.3 在窗体中添加用户控制图	(302)
7.3.4 向 FormDev 中添加用户控制图	(302)
7.3.5 创建用户控制图库	(302)
7.3.6 创建用户控制图 DLL	(305)
7.3.7 Windows 和 OS/2 消息	(305)
7.3.8 CA-R 通知消息	(306)
7.3.9 CA-R 查询消息	(306)
7.3.10 CA-R 数值消息	(307)
7.3.11 CA-R 字符串消息	(307)
7.3.12 用户消息	(309)
7.3.13 向 CA-R 发送消息	(309)
7.3.14 窗体颜色	(310)
7.4 动态数据交换	(310)
7.4.1 何谓 DDE	(310)
7.4.2 DDE 对话初始化与选择	(311)
7.4.3 DDE 数据格式	(312)
7.4.4 消息由客户向服务器的传送	(313)
7.4.5 处理服务器应答	(313)
7.4.6 等待 DDE 消息	(314)
7.4.7 结束一个 DDE 对话	(314)

7.4.8 CA-R 充当服务器	(314)
7.4.9 响应来自客户的信息	(316)
7.4.10 获取关于 DDE 对话的信息	(316)
7.4.11 两个例子	(316)
7.4.12 CA-RET 与 CA-R 的通信	(320)

引 言

1. 为什么选中了 CA-REALIZER

计算机应用的好坏,直接受制于计算机教育的普及程度和水平高低,这是任何人也不会怀疑的。

早在 11 年前,Microsoft 公司推出 Windows,取得了巨大的成功。国外软件界人士将它的出现形容为“在软件界爆炸了一颗原子弹”。它的竞争对象——IBM 开发的 OS/2 也已面市 9 年,这二者在国际上早已掀起巨波狂澜。由于 Windows 结构复杂、占用内存量大,整个编程环境发生了根本性的变化。这些变化不仅给计算机生产厂家、外设生产厂家和软件公司提出了一大批新课题,同时由于在 Windows 和 OS/2 下编程的高难度和繁复性,几乎使广大的业余程序员和中初级程序员遭受灭顶之灾。

我国教育界作为一个整体,对这场深刻而巨大的变化尚处于静观状态中,然而这种状态是不可能长期延续下去的,那么,下一步应迈向何方?

经过仔细地对比和认真地挑选,我们从众多的程序设计语言中选中了 CA-REALIZER。它是当今世界上第二大软件公司(Computer Associates International)的第三代 BASIC 语言。主要看中了它三点:

- 它可同时在 Windows 和 OS/2 两种环境下运行。
- 它仍采用我国计算机用户熟悉的过程模式,同时又吸收了事件驱动模式(用 FormDev 开发 GUI),它的最后成果有完整的程序清单,使日后的维护工作简化,而这正是既非软件生产厂家又非最终用户的一大批软件工作者的工作领域。
- 与 Quick BASIC 高度兼容,Quick BASIC 的 221 个保留字中有 104 个(占 47%)被 CA-REALIZER 完整地袭用;另一些保留字被加上分类字头(如处理文件的加上 File 字头:FileOpen,FileClose,FileRead,FileWrite 等;绘图语句加上 Chart 字头:ChartLine,ChartSelect,Chart Text 等)在 CA-REALIZER 中也得到广泛使用。这就使得从 Quick BASIC 过渡到 CA-REALIZER 变得非常容易,不要求用户花大量时间去学 C 语言或 C++。熟悉 Quick BASIC 的用户经过短期培训或自学,就能在 Windows 或 OS/2 下开发自己需用或市场需要的应用软件。

这里不妨将 DOS 比喻为一个筑于此岸的比较矮小、吃水线较浅因而只能停泊中小船只的码头,而 Windows 和 OS/2 则是筑于彼岸的比较高大大、吃水线深可以停泊巨轮的码头。去年我们发现了这样一种过渡方式:旅客从 DOS 码头登上 Quick BASIC 渡船,这渡船可以径直驶入 CA-REALIZER 巨轮的船舱中,CA-REALIZER 可以直航并停泊于 Windows 或 OS/2 码头,舱中的旅客就可进入 Windows 和 OS/2 的五光十色的新天地了。

2. CA-REALIZER 的特色

CA 公司在向公众介绍 CA-REALIZER 2.0 时讲它有以下 11 项特色:

- 能在 Windows 和 OS/2 下进行开发。

- 在 REALIZER 内建立预编码库和格式控制图库。
- 在 FormDev 内任意使用 Windows 与 OS/2 的用户控制图。
- 提供电子表格、图表、观察板、文本簿、动画、定时器、窗体、菜单、绘图板等 9 种可编程应用工具。
- 极易将 Quick BASIC 成果转移到 Windows 或 OS/2 下。
- 输入/输出 Lotus 1-2-3, Excel, SuperCalc, Compete! 及 XBase 文件。
- 交互式所见即所得应用设计。
- 生成用户可访问和可修改的 BASIC 代码。
- 全面支持 DDE(动态数据交换)和多种 DLL(动态链接库)。
- 从 DBF 文件实现即时数据库窗体。
- 可使用矩阵运算符和矩阵函数的动态可扩展多维全程数组。

从我们与 CA-REALIZER 的实际接触看,以上 11 点不但没有夸大其词,仅仅给人留下了轻描淡写的印象,恐怕初学者很难领会其中的奥妙。这里我们仅就语言机制本身补充几点,望读者明察:

(1) 大量吸收其他语言(特别是 C)的精华

①CA-REALIZER 的基本数据类型 11 种,实现了与 C 语言的全面沟通。

②抛弃了 DATA, READ 语句,采用花括号{...}直接向数组赋初值。

③由于 C 语言中有 7 个位操作符使它获得了“高级汇编”的美称;现在 REALIZER 有 9 个位操作函数,即除了等价于 C 语言的 7 个操作符之外,还多了一个位清除和一个位测试函数。

(2) 数组功能的巨变

①第二代 BASIC 的动态数组只能改变数组元素的多少,不能改变数组的维数。而且数组一旦被重定义,原先存放的数据全部丧失。REALIZER 的动态数组不但可以改变元素的多少,也可以改变维数的多少,变换以后的数组中原先存放的数据原封不动。

②原先,数组的赋值和运算都要在循环体内实现;但在 REALIZER 中可用一条简单的赋值语句极为快速地实现数组的赋值和运算。

③过去我们总是讲“数组是若干同一类型数据的集合”。CA-REALIZER 打破了这个框框,它提出了“子域”(Subrange),“子数组”(Subarray)和“数组的数组”(Array of array)等一系列崭新概念。结果使得不同类型的数据能够共处于一个数组的数组中。

(3) 全动态结构的“家族”(Family)变量

Quick BASIC 在创建文件前先要使用 TYPE 语句定义文件结构,C 语言则使用 struct 语句定义结构。它们定义的结构是静态的、不可改变的,这样就给日后文件的维护(例如要增加或减少字段)带来困难。CA-REALIZER 提出了一种新的复合数据类型——家族,家族的成员随时可增减,从而实现了全动态结构。不仅给应用软件的开发者,更重要的是给广大的从事软件维护工作的人带来了极大的方便。

(4) 函数功能的巨变

CA-REALIZER 比 Quick BASIC 增加了许多非常有用的函数(如统计函数、时间序列函数和矩阵运算函数等),然而这只是数量的增加,不足以引起我们的惊叹。我们注意到在函数运算的方式上有两点巨大的变化,给用户带来了极大的方便:

①任何函数如果将它的变元设置为数组,不管这数组有多庞大,只消通过一个简单的赋值号(=)就可在它的输出端得到返回的函数值的数组。

②REALIZER 在处理两个和两个以上数组时可以把它们作为相关数组考虑。例如 A,B 是一对相关数组,如果将数组 A 排序,可以依次输出数组 B 的对应元素。

(5) 柔性参数表的新概念

过程调用中的难点是参数传递,在多人共同开发项目中更容易纠缠不清。CA-REALIZER 提出了“柔性参数表”的新概念,不仅能妥善地处理已知参数和修饰词的传递,而且能处理未知参数和修饰词的传递,这为开发大型项目提供了极大方便。

(6) 全面的整理了关键字

CA-REALIZER 共有 417 个关键字,为了让用户好记好用,它作了一次彻底地整理。

①保留用户早已熟悉的 Quick BASIC 的关键字 104 个(占 Quick BASIC 关键字的 47%)。

②将 226 个关键字进行分类,加上分类字头(见下表)以便于记忆:

编号	字头	数量	含 义	编号	字头	数量	含 义
1	Add	3	加进	12	Menu	8	菜单
2	Animate	5	动画	13	MK	7	目录
3	Bit	9	位处理	14	Picture	7	图画
4	Board	7	板面	15	Q	9	查询
5	Chart	38	图表	16	Sched	6	定时器
6	Comm	5	通讯	17	Set	7	设置
7	DDE	7	动态数据交换	18	Sheet	11	电子表格
8	File	15	文件	19	Stat	12	统计
9	Font	5	字形	20	Std	5	标准输入/输出
10	Form	14	窗体	21	Str	5	字符处理
11	Log	12	簿	22	Tablet	25	绘图板

③使用 5 个有固定意义的字串,搭配出 50 个关键字。如下表:

Control	New	Q	QUnique	Select
BoardControl	BoardNew	BoardQ	BoardQUnique	BoardSelect
ChartControl	ChartNew	ChartQ	ChartQUnique	ChartSelect
DDEControl	DDENew	DDEQ	DDEQUnique	DDESelect
FileControl	FileNew	FileQ	FileQUnique	FileSelect
FontControl	FontNew	FontQ	FontQUnique	FontSelect
FormControl	FormNew	FormQ	FormQUnique	FormSelect
LogControl	LogNew	LogQ	LogQUnique	LogSelect
MenuControl	MenuNew	MenuQ	MenuQUnique	MenuSelect
PictureControl	PictureNew	PictureQ	PictureQUnique	PictureSelect
SheetControl	SheetNew	SheetQ	SheetQUnique	SheetSelect

这样就大大减轻了 CA-REALIZER 对用户记忆力的要求,多数用户在使用 REALIZER 三四个月之后,记住 90% 以上的关键字,是不会有太大困难的。

(7) 朝着“通用的宏语言”方向迈进

软件界的骄子、BASIC 的狂热爱好者比尔·盖茨在一篇纪念 BASIC 25 周岁生日的文章末尾最后一句话是这样写的:“我相信,就 BASIC 而论,这种语言正在欣欣向荣地成长,一旦 BASIC 变成一种通用的宏语言,依我看它正向此方向发展,那么它将比其他任何过程语言都长寿。实际上,BASIC 可能比我们这些人的寿命还长。”^[1]

看得出来,CA-REALIZER 的开发者们正在沿此方向前进。CA-REALIZER 共设置预定义常量 593 个,将不同的预定义常量(例如: _Read 和 _Write)放在同一个函数或命令里如:

FileOpen(文件号,“文件名”,_Read)

FileOpen(文件号,“文件名”,_Write)

构成两条功能不同的命令。593 个预定义常量与 417 个关键字相加是 1010 个不同的语法成分;二者相组合,能构成多么强大、多么复杂的语法功能是可想而知了。

3. 适用环境及安装

(1) 适用环境

CA-REALIZER 2.0 适用的硬件环境是 80286 以上及其兼容机,内存在 1MB 以上,若在 Windows 环境下运行 CA-REALIZER 的基本版本至少需要 2.8MB 的硬盘,完整的 CA-REALIZER 需要 7.6MB 硬盘;在 OS/2 环境下,则基本版本至少需要 3.5MB,完整版本需要 8.5MB。

CA-REALIZER 需要的软件环境是 Windows 3.0 及以上版本;OS/2 2.0 及以上版本。

(2) 安装

在 Windows 环境下的安装:将安装盘插入驱动器;从 Windows 的程序管理器的 File 菜单选择 Run 命令;而后键入 A:\install(或 B:\install),即可遵照屏幕提示进行安装。安装完毕后,CA-REALIZER 的图标都将出现在 Windows 的程序管理器当中。

在 OS/2 环境下的安装:将安装盘插入驱动器后,在 OS/2 的命令提示符下,键入 A:\install2(或 B:\install2),即可将 REALIZER 安装到指定目录下。

4. CA-REALIZER 开发环境

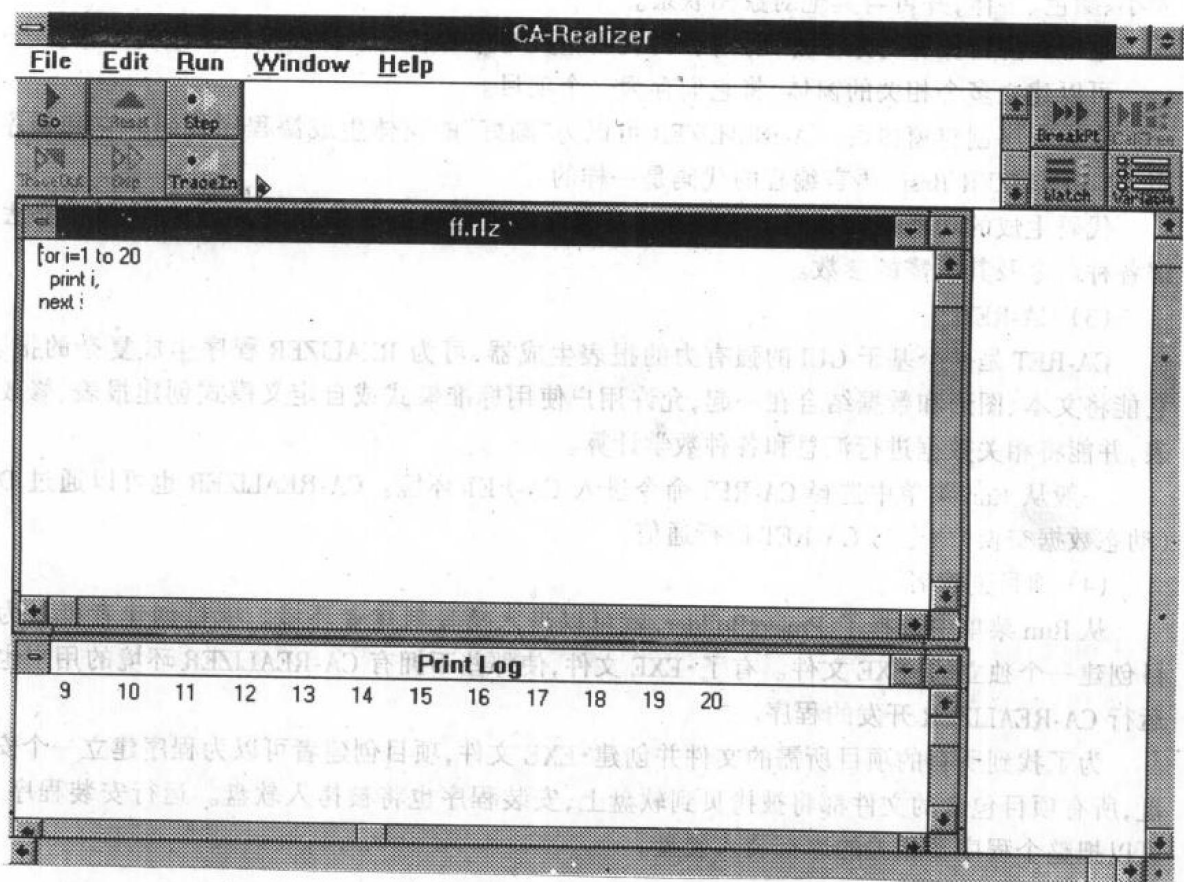
启动 CA-REALIZER 后,它会给出一个如图引-1 所示的菜单屏幕。通过选择执行菜单中的命令,可以打开诸如编程窗口,FormDev GUI 开发窗口,CA-RET 报表生成器以及项目创建者窗等各种不同的窗口。

(1) 编程窗口

从 File 菜单中选择 New 或 Open 命令就打开了编程窗口。程序员就在此窗口中敲入程序清单,进行编程和程序调试工作,这个环境中有几个辅助窗口:

- 输出窗:在 CA-Realizer 的 Window 菜单中选 Show Print Log 命令项,就打开了输出簿窗口。程序中 PRINT 和 ROWPRINT 语句输出的内容就显示在这个窗口上,使用 Ctrl-Shift-Del 组合键可以清出输出簿的内容。

[1] Bill Gates The 25th Birthday of BASIC 《BYTE》OCTOBER 1989 268~276P.



图引-1 CA-Realizer 环境

- **调试窗**:从 Window 菜单中选择 Show Debugger 命令项,就打开了调试窗。调试窗左上部设有 Go(运行),Reset(重新设置),Step(单步运行),TraceIn(跟踪),TraceOut(停止跟踪) 6 个按钮;右上部设有 BreakPt(中断点),Watch(观察点),Variable(观察变量)4 个按钮。在不同时刻有不同的活化按钮,点击活化按钮,调试器就会执行不同的功能(如运行或单步运行,跟踪或停止跟踪),或打开专用窗(如断点窗或观察变量窗等),为程序的调试提供了极大的方便。
- **错误簿**:当程序运行出错,它会自动出现在编程窗口上,说明出错的位置和出错的原因。用户看明白这些信息后按 OK 或敲回车键,错误簿消失;而错误簿中的信息以红颜色显示在调试窗的提示栏中,用户对程序进行修改后,先要用 Ctrl-Alt-F2 清除这些信息,使系统复位后才能重新运行程序。

(2) FormDev

从 Run 菜单中选择 FormDev 命令,就进入了 CA-REALIZER 提供的 GUI(图形用户介面)开发环境 FormDev。它是 CA-REALIZER 的图形化的、所见即所得的开发环境,为迅速开发出美观实用的 GUI 提供了方便。

窗体是 REALIZER 最基本的可编程工具,是一个可以包含任何对象的窗口,是向用户传送信息和从用户获取信息的媒介。

对象是窗体中的元素,如按钮、列表框、位图等,也可以是 CA-REALIZER 中的可编程工具,