

第四代语言(4GLs)的十个问题

张荣光 (山西大学计算机科学系)

摘要: 4GLs是八十年代的新一代计算机语言,它的基本特征是非过程性并获得数据库功能。在编码生产上比COBOL高一个数量级,很适合于最终用户使用,但它要高消耗硬件资源,也不很适合于开发极大容量或很复杂的应用。集成的4GLs可分为面向最终用户的和面向DP人员的。它大有取代COBOL之势,但并存局面将是长期的。

1、4GLs为什么是八十年代的新一代计算机语言?

4GLs的前辈是机器语言、汇编语言和高级语言。重要软件技术的里程碑有:1969—1975的报表生成,1976—1984的数据库技术以及在1982年以后开始的第四代语言系统。

虽然在七十年代中期以后已陆续出现了一些第四代语言的产品,但是只是到了八十年代才出现用于开发应用的更有效的软件,以DBMS为中心设施的应用生成器得到了有效的使用,加之到了八十年代,硬件资源的廉价促进了4GLs的成长,在某种程度上是把高成本的程序员劳动转嫁于计算机软件所消耗的低廉的硬件资源。所以说,4GLs是八十年代的新一代计算机语言。大约在1982—1983,第四代语言开始进入数据处理的主流。

2、4GLs是什么样的语言?

还没有一个普遍接受的精确的定义。第四代语言这术语是由James Martin首先创立并积极推广的,他和Pieter Minno下的简单定义是:

a.以第三代语言如COBOL、Fortran所需的时间之十分之一,用户就能得出结果。b.这系统与用

户是友善的,有诱惑力的,并易于学习和记忆。c.它适合于最终用户与DP专业人员两者的使用。

但是很多人心目中的第四代语言是包括有一系列集成组件的应用开发系统,其中有一个DBMS或能访问一个或多个DBMS的能力,还应有数据词典、报表工具和查询语言等。

3、4GLs与COBOL等第三代语言的主要区别?

4GLs相互间在语义和语法上的不同远大于COBOL和Fortran之间的不同,特别是在应用类型和环境类型上,但也有若干共同点区别于第三代语言。

1)它基本上是面向结果而不是面向过程的。

2)它的编码行比第三代语言少得惊人。

3)它与用户的界面具有多样性与友善性,如自然英语、二维菜单、驱动窗口命令、图象符号、鼠标器等。

4)它改变了计算机工作模式,从编辑-编译-连接-运行的过程到一个交互的输入-运转(Enter-and-go)的方式。4GLs多数是高度交互地解释执行或结合有某些编译性特征,它不仅使用大量CPU,而且以不同模式去使用CPU。

五、结论

我们提出的新一代计算机模型HNM,是对归约型并行推理机进行改进后得到的,并行推理机是新一代计算机研究中的关键。模型HNM中的存贮器装置采用了Cache-主存-辅存三级存贮层次,目的是提高查找速度和扩大存贮容量;处理机装置采用了并行处理机形式,目的在于提高并行处理速度;

我们引入了自动机的概念,提出了资源也是知

识的思想。新的同步机构实际上是对知识的处理,由于把资源描述为自动机,对并发进程的同步控制较易实现。

新同步机构H I 和H II 类似于当代操作系统同步机构中的信号灯和管程。在H I 和H II 中,一进程对一资源进行操作期间,把资源的当前状态值置为-1,操作完后把该值置为资源当前状态值。对I型资源的实现可集中起来放在知识库中。

信号灯的值得由P、V操作改变,用信号灯实现同步的一般形式是, (下转24页)

5) 第四代数据的主要特征是每个域一个事实, 每个事实一个域。

4、4GLs是非过程性语言吗?

基本上是一个较复杂的4GL是由很多组件构成。如自然英语、二维菜单、命令式语言、查询语言、报表生成器、图象生成器、屏幕显示器/屏幕生成器、前端设施、易用编辑器等都具有非过程性功能。

非过程性的组件使用户相对容易地指定计算机去做其想做的事, 有经验的系统分析员可以用非过程性组件去建立应用系统的最初的原型系统。

然而, 一个4GL如果不具备过程性语言的功能, 其应用范围是有限的, 它不能执行多数应用所需的特定的逻辑。

某些4GLs, 如FOCUS、Nomad2、Natural Ideal和Application Factory等, 其过程性功能较强, 可代替COBOL语言而使用到多数商业应用上。而过程性功能较弱的4GLs, 能转到COBOL Fortran、PL/I、APL或汇编去, 以便执行特定的过程逻辑。

5、4GLs与DBMS有何关系?

多数4GLs都具有数据库的特性。有一类4GLs提供到一个或多个流行的数据库的界面并增加数据库功能, 使有一种容易方法来建立数据库文件和应用。有另一类4GLs是由数据库卖主提供的, 它和卖主的DBMS结合在一起。

考核一个产品是否是一个完全的DBMS, 有专家提出应具备下列九条: 必须有一个数据模型; 必须有能力去建立物理文件; 必须有一个数据描述语言(DDL); 必须有一个数据操作语言(DML); 必须有一个数据控制语言(DCL); 必须有系统安装能力; 必须有能力用宿主语言写程序; 必须能执行数据通信; 必须有一个数据字典或目录。

DBMS和4GLs结合的产品Focus、Nomad 2、Ramis I, 它们在数据字典或目录、系统安装方面有欠缺, 它们不是完全的DBMS。

6、4GLs有哪些优越性?

一是高生产。程序员以4GLs编程比COBOL或Fortran编同样功能的程序能节约至少十分之九的时间, 因而开发应用软件的周期大为缩短, 所以它对降低开发成本, 解决目前存在着的积压现象提供了好工具。

二是最终用户可积极参与应用开发。不需或所需甚少DP人员的帮助, 自己可开发不很复杂的应用。因为使用4GLs不需很多程序设计知识, 它易学易用。这对普及推广计算机应用是个好工具。

三是出错少, 查错修改方便。这不仅因为编码很少, 更重要的是由于在第三代语言的程序中, 80%多用在处理程序流的工作, 更可能包含着错误, 而这部分易错的工作现在交给4GL软件去做, 从而大大减少出错的机会。其次最终用户积极参与应用开发, 使应用系统更易于符合用户的需求, 这减少了修改的需要。

据此, 提高了应用的可靠性, 并降低维护成本。

四、特别地4GLs是信息中心的理想工具。

7、4GLs有哪些缺点?

首先, 缺少统一的行业标准, 各式各样的4GL产品花样繁多, 此外, 所连接的数据库管理软件往往也不同。因此, 使用这些4GLs来开发的应用不易移植。

第二, 4GLs的严重缺点是“很吃”计算机资源。它比第三代语言多消耗50%到150%的CPU和磁盘资源, 这在一定程度上抵消了使用4GLs带来的生产效益。

第三, 运行效率低, 响应时间长, 而且, 如不好控制应用的增长, 会危害公共数据库的完整性。

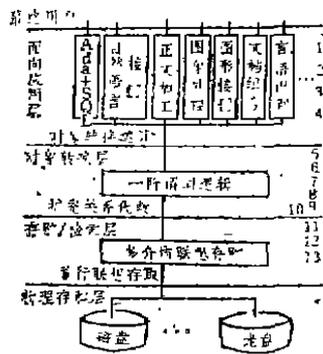
第四, 它不宜用于科学计算, 不宜于高速实时需求的应用系统, 也不宜于开发复杂的大容量的应用系统, 因为要求过多的过程逻辑, 必然减少4GL提供的总的生产效益。

8、4GLs如何分类? 举例之。

根据James Martin定义, 依应用生成的类型而区分为集成的个人计算机工具, 查询语言和报表生成器, 图象语言, 决策支持和财会模型系统, 面向最终用户的应用生成器, 面向DP专业人员的生成器, 可证明说明书的语言。

但很多人心目中的4GLs仅限于上述的应用生成器, 它是很多有用的功能集成一起的开发工具, 应能实现四个方面的应用编码: 屏幕设计, 报表打印, 数据库描述与操作, 过程编码。

认为4GLs是应用生成器时, 即是集成的系统而不是单功能的工具, 此时亦可区分为两类: 一类是为建造复杂的系统的专业程序员设计的, 如Natural, Mantis, ADS/O, Ideal, ADF, CSP, DMS, Info, MarKV Linc等。另一类是更多地为信息中



注：1——各种视图；2——各种表示；3——部分推理；4——转换演算；5——通用对象的组合变换；6——复杂请求分解为一系列扩充关系代数运算；7——部分推理(提高性能)；8——完整性检查；9——事务管理(广义事务)；10——授权检查；11——联想存贮；12——联想检索；13——并行存取。

种指导性的轮廓。很明显，很多问题尚未解决，还有很长的路要走。问题的清单可以提一大串，例如：

1. 知识表示的转换问题，如何从一种form转换为另一种form？
2. 自然语言理解问题。

3. 新的查询语言的定义。
4. 如何组合文档？如何存贮和检索它们？
5. 图象的表示和检索问题。
6. 规则库的组织管理问题。
7. 对象的表示和转换问题。
8. 如何高效地实行推理？
9. 广义事务的定义和管理。
10. 优化的准则和优化技术。
11. 事实，正文，图象，对象的存贮方法。
12. 并行存取问题。
13. 存取路径的建立和利用。

如上所述的通用的新一代DMS其体系结构过于复杂，也许可行的办法是设计和实现范围专门一些的系统。例如专用于CAD/CAM/CASE/CIM的系统，专用于ES的系统，专用于DSS，MIS的系统，专用于五代机的系统，专用于图象处理的系统，专用于正文处理的系统等等。将所有功能集成到一个通用的系统中，其可行性和性能还有许多值得研究的课题，期望不久将会有若干原型系统研制出来，它们将提供非常宝贵的经验和教训。

(上接封四)

[10] N. Adam, Rin Betting, On- 4GLs Or COBOL, CW July 9 1986

[11] Robert Morison, 4GLs vs COBOL, CW Aug.12 1985

[12] Richard Watts, Fourth-generation language how to choose the right one, data processing Vol.27 No.9 1985

[13] Al Hendrickson, Performance Benchmarks For 4GLs, CW July 9 1986

[14] John Desmond, Language barrier, The fourth generation at work, CW Nov. 11 1985

[15] Keith Nelson, Technical requirements of fourth-generation languages data Processing Nov. 1985

[16] 4GLs drawing bead on PCs, CW No.20 1986 P19

[17] Richard Cobb, Fourth-generation language, from backwater to mainstream, CW No.41 1985

[18] Dan TagKer, In Search of Fourth Generation Data, DATAMATION July1, 1987

[19] Rebecca Hurst, DBMS and 4GLs, Splitting the Difference, CW No.1A 1987

[20] E.F. Codd, Is your DBMS really relational?, CW No.41 1985