

反编译技术研究现状及面临的问题

刘宗田 陈福安

TP311.56

(合肥工业大学微机所, 合肥230009)

摘 要

This paper analyses the current status of research on decompilation technology at home and abroad, discusses its applied prospects and relation with the Law of Software Protection, and put forward the ways to solve the technical problems caused by various types of languages, computers(CPUs), and compilation versions in the decompilation.

一、引 言

随着软件技术的不断发展,对现有软件的分析理解、改造和维护、以及利用和再开发的工作变得日益重要。虽然目前大部分软件是用高级语言编写的,但由于技术保密等原因,绝大部分软件是以机器码形式提供给用户,机器码可读性极差,因此人们需要有一种能将机器码程序翻译成易于阅读和理解的高级语言程序的软件工具,这就是反编译系统。反编译作为一种软件分析和理解的工具,是软件逆向工程的重要组成部分。可以说自编译技术产生以来,人们就开始了反编译技术的研究,但反编

译技术远不如编译技术发展迅速,原因在于:①反编译技术难度大,处理对象十分复杂,以至于很难研制出产品化的反编译系统;②虽然反编译系统可应用面广,但使用人员范围较小,主要是由高级的软件研制人员用于解剖分析先进的系统软件和应用软件。

近十年来,随着软件开发环境技术的发展,和软件逆向工程的研究,反编译系统作为软件开发、调试和解剖分析的辅助工具,渐渐引起人们的重视,国内外研究和应用反编译技术的报道增多。本文从回顾反编译研究的历史着手,分析了国内外反

Toolkits

- [10] P. P. Puncello, et al., ASPIS: A Knowledge-Based CASE Environment, IEEE Software, Vol.5, No.2, pp58-65
- [11] X. L. Qian, et al., Consistency Management in a Project Management Assistant, Software Engineering Notes, Vol. 15, No. 6 (ACM SIGSOFT'90), pp34-43
- [12] C. Rich and R. C. Waters (Ed.), Readings in Artificial Intelligence and Software Engineering, Morgan Kaufmann Publishers

- [13] Tom Strellich, The Software Life Cycle Support Environment, A Computer-Based Framework for Developing Software Systems, ACM SIGSOFT'88, pp35-44
- [14] A. J. Symonds, Creating a Software Engineering Knowledge Base, IEEE Software, Vol. 5, No. 2, pp50-57
- [15] Special issue on artificial intelligence and software engineering, IEEE Trans. on Software Engineering, Nov. 1985
- [16] 钱家骅, 软件过程·人工智能·软件环境, 国际学术动态, 1988年第1期

编译技术研究现状,讨论了反编译研究所面临的观念和技术方面的问题;反编译与软件保护法的关系,反编译技术的应用前景,如何解决反编译中多语言,多机种(CPU),多编译版本所引起的技术难题等。

二、国外近二十年反编译技术研究状况分析

根据我们对从1969~1991年反编译技术研究的国际联机检索资料的分析,国外对反编译技术的研究和应用可分为以下三类:

1. 用于程序分析的反编译器

文献[1]介绍了一个将机器代码程序转换为形式说明的程序分析工具CPU,它以程序和编程规划知识库为输入,在代码中识别过程规划和数据结构规划,然后以目标描述取代之,最后结果是输入程序的形式说明;文献[2]介绍了一个从中间代码一波兰码转换为BASIC程序的反编译算法。

2. 软件开发环境中的反编译器

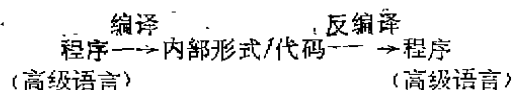
文献[3]介绍了在CONCERTO软件工程环境中,用于SDL说明语言的语法制导的图形编辑器,该编辑工具提供了SDL专用命令集和一个反编译器;文献[4~6]介绍了FORTH程序开发中的反编译技术的应用;文献[4]介绍了一个FORTH程序开发辅助工具——一个高级FORTH反编译器SEE,该工具显示带有变量存储器地址的源程序,使用它开发FORTH的词典结构和修改程序;文献[5]介绍了一个辅助初学者掌握FORTH语言特点,在FORTH中增加了语言反编译功能;文献[6]介绍了一个反编译器用于列出FORTH词典表中高层定义的所有词汇。

3. 特殊使用中的反编译器

文献[7]介绍了反编译技术在结构化文件/资料处理系统Mentor的语法制导编辑器中的应用,被编辑的对象文本内部组成抽象语法树,当用户需要文本时,反编译器的作用是从抽象语法中计算文本并在屏幕上显示;文献[8]介绍了神经网络模拟环境中反编译技术的应用,实际是一个图形编译转换为内部代码,需要显示/修改图形时,从内部代码反编译转换为图形的过程;文献[9]介绍了基于多处理器和多任务操作系统,具有面向对象特点的软件环境中所使用的通用的对象消息处理的编译/反编译器;文献[10]介绍了一个产品中表达式编译/反编译器的应用,以提高计算速度。

由上述列举的资料分析可看出,国外对反编译

技术的研究和应用并不以解剖分析程序为主要目的,主要是作为软件开发和调试的辅助工具,应用模式为:



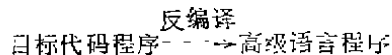
这里编译器和反编译器基本上是一家公司研制的,所以实现技术上难度小。即使用于程序分析目的的反编译器,还尚未达到从机器码程序转换为高级语言程序的技术程度。

三、国内反编译技术研究现状

由于需要解剖分析从国外引进的软件,以便于软件的汉化移植、改造和维护,以及从中学习和借鉴先进的程序设计技术,国内从八十年代初开始了以程序分析为目的、从机器码向高级语言程序转换的反编译技术研究。国内反编译对象主要是C语言程序,所用的机器均是国内主流机型:VAX的小型机系列和PC的微型机系列。例如,合肥工业大学微机所分别在Dual 68000上研制了能将符号信息的机器码程序翻译成C语言程序的68000 C反编译系统^[11],和在PC机上研制了能将无符号信息的机器码程序翻译成等价C语言程序的8086C反编译系统^[12],并将68000 C反编译系统改进移植到SUN-3/60工作站上,上海交通大学和武汉大学分别在VAX机上研制了C语言反编译系统^[13],另外还有北京信息工程学院、中科院计算所和辽宁大学、重庆大学等单位进行了反编译系统的研制^[14]。

对所实现的反编译系统中应用的各项反编译技术,合肥工业大学微机所做了连续报道,分别是符号执行技术^[15],程序变换技术^[16],C库函数模式识别技术^[12],数据类型恢复技术(录用待发表),中间语言设计^[17],语句翻译器设计^[18]等。

国内研究现状是:反编译技术研究是以程序解剖分析为目标,应用模式为:



针对单一机器上单一语言编译版本的反编译系统,在一定条件下接近实用程度。问题在于研制的系统专用性太强,受到不同语言、机型、和编译版本所造成的目标代码结构差别的影响。这种状况与我国目前软件开发水平落后于发达国家,国内市场需要

适合国情的软件,以及软件行业发展要走向国际市场的情况是分不开的,这里必须注意软件反编译中的版权问题。

四、反编译与软件保护法的关系

现在科学技术的任何进步都离不开对已有成果的借鉴,而软件产品的开发同样是需要借鉴前人的软件成果和技术,同时为了市场需要,软件的移植性开发和兼容性开发是必不可少,因此对于他人软件产品进行软件逆向工程(如反编译)以分析他人程序的功能、设计思想、算法、数据结构、和执行流程等设计要素,评估设计中的得失,作为自己开发新产品设计的参考,这在软件产业界是常见的行为。还必须指出将目标代码还原成高级语言源程序本身就需要很高的技术能力和创造性以及花费很大的代价。过去对他人软件进行反编译一般都认为是合法的,包括国际上在法律方面也未受到非议。但随着高技术产业中商业竞争的日趋激烈,国际计算机法律界正在热烈讨论软件逆向工程这个问题,尽管目前尚无定论,但我们必须要很好地研究反编译与软件保护法的关系。

1991年10月1日我国开始实施《计算机软件保护条例》,这是我国尊重知识产权、尊重国际惯例,防止软件非法传播的具体表现。我国实行的是计算机软件著作权保护,符合国际上用版权法保护软件的主导原则。研究软件反编译是否违反软件保护法,必须先分析我国著作权保护软件涉及软件反编译的条款:

① 计算机软件是指计算机程序(源程序和目标程序)及其有关文档;

② 对软件的保护不能扩大到开发软件所用的思想、概念、发现、原理、算法、处理过程和运行方法;

③ 侵权行为包括:未经软件著作权人或者合法受让者的同意修改、翻译、注释、复制和转让其软件作品;同时允许因科学研究等非商业性目的的少量的软件复制^[20]。

各国版权法所保护的是软件作品的外部表现形式,而不保护作品的创作思想,所以软件的合法持有者使用反编译工具解分析他人软件,探索程序的设计思想,算法,执行流程和数据结构,这显然是允许和合法的,但问题在于通过反编译还原软件源程序的使用目的。作为软件反编译主要目的是获取源程序后进行分析,以进行程序的修改和维护,以及开发新软件,必须指出进行反编译必然地要获得被

分析软件和使用它,并且获得的成果往往希望向他人提供,以取得经济效益,所以这就存在着软件使用中的版权问题^[20]和软件开发中的版权问题^[21]。

① 修改是软件著作权人的专有权利,但软件的合法用户可以为了自己使用的需要而对之进行修改,若无事先协议,修改后的软件版权仍属于著作权人,用户不得将修改的软件向他人提供。一般地说对他人作品的改造属于制作他人作品的演绎作品,如对国外软件产品进行汉化工作就是属于这种情况,所以对使用反编译工具辅助软件产品的修改和维护,如果是为自己使用这是合法的,而未经许可向他人传播则就不合法了。

② 在利用反编译分析他人软件的设计技术进行软件开发时,可分为两种情形:

(a) 纯粹为了开发原软件的附属和配套程序,以扩大原软件的功能,因为这一般只使用原软件的程序接口说明,也不会窃用原有程序,而程序接口说明是属于程序的思想概念,所以是合法的;

(b) 开发兼容软件:按照版权法原理,使用他人软件作品的构思独立地开发与其功能相似、接口兼容的程序是合法行为,问题在于开发兼容软件的手段上,因为这存在对原软件的使用和功能分析。美国各法院在判断兼容软件是否侵犯版权的准则是“实质相似加接触(Substantial Similarity and Access)”,因为证明程序复制行为相当困难。因而开发兼容软件特别注意在:1)程序的结构(指程序组成的语句,程序段、数据结构);2)程序执行流程;3)程序的组织(如程序嵌套、调用、串行/并行关系)方面不能存在相似性,另外所有实际的程序开发人员必须做到没有接触过原软件,所以出现了开发兼容性软件的“净室技术(Clean Room Technique)”。这里反编译工具用于将他人软件代码转换为高级语言,由分析人员从中提取不受版权保护的设计构思。

五、反编译技术的应用

前景和面临的问题

通过反编译使用合法性的分析,反编译技术可以应用以下三个方面:

① 软件的改造和维护

软件产品研制是产品开发过程中主要花费阶段,而软件产品的生产仅是简单的复制。因而改造已有软件以适应用户的需要,要比从头开发新软件经济得多。尽管现在绝大部分软件是用高级语言编写的,但厂家为了保护自身利益而不愿意向用户提

供源程序,源程序售价往往高出机器码程序的许多倍;同时软件的改造也不象其它工业产品那样可以通过直观检测和更换某些部件,软件产品是复杂的,未对整个程序有全面的理解,则连其中一个字节也难改动。机器码可读性极差,而使用反编译器将它翻译成高级语言程序,有助于用户阅读和理解程序,是进行软件改造的有效途径。

另外目前在软件生命周期中,程序维护是其最大的花费阶段。软件维护难度较大,再加上用户手中无源程序和文档资料,使得只有厂家才具备软件维护能力。而厂家的软件开发人员不可能将大量精力投入维护,已交付的软件产品一旦投入运行并超过规定维修期,开发部门就没有维护责任了,因而导致不少软件由于得不到维护而被闲置,造成严重的经济损失。使用反编译工具辅助用户维护,将降低软件维护费用和提高软件使用率。

③软件开发环境中的程序调试工具

反编译技术可以应用到软件开发环境中的高级语言动态调试工具,让用户直接观察到语句级执行,显示程序运行出错点的源代码,将给程序调试带来极大的方便。虽然目前一些DEBUG软件具有类似的功能,象Microsoft C中的Code View等,但这些软件的使用需要在编译中加选择项,以保存目标代码与源代码地址对应表和符号表。另外反编译技术的思想也可应用到某些复杂信息对象的压缩-还原过程,以节省存储资源。例如图形对象以内部编码存贮,当用户需要显示和修改时,再从内部编码反编译还原成图形对象显示。这些在开发国产化的语言编译系统中大有用场。

③学习先进软件的设计技术和开发兼容软件

使用反编译工具辅助用户解剖分析先进软件,从中了解软件的设计思想、使用的算法、定义的数据结构,以及执行流程,可以学习到他人的软件设计技术,提高自身的软件开发水平;同时这也是开发兼容软件的必要手段。

从前面反编译技术研究状况分析来看,目前针对特定机器(CPU)、特定语言和特定编译版本的反编译技术基本成熟,所研制的反编译器在一定条件下接近实用程度。但是,随着大规模集成电路和硬件设计技术的发展,以及软件工程的进步,国内外计算机

种类不断增加,同时所使用的程序设计语言种类也繁多,而且同一种语言在同一机器上又有多种编译版本,所以反编译技术研究主要面临以下三个方面的问题有待解决:

① 同一机器上,同一语言的不同编译版本存在目标代码结构上的差异,即多编译版本问题;

② 同一机器上,不同语言编译存在目标代码结构上的差异,即多语言问题;

③ 不同机器上,同一语言编译存在目标代码上的差异,即多机种(CPU)问题;

这些问题都影响到以程序分析为目的反编译系统的实用性和通用性。对这些问题,某些可能的解决途径是:

① 针对每一种情况,分别开发各自的反编译系统,这种穷举方法势必使反编译投资庞大,且研制的反编译器时效性差,显然不适合实际需要。

② 基于同一机型,设计良好的中间语言,开发标准的反编译系统,对各编译版本的影响或者不同语言的差别(目标代码结构上)采用预处理方法分别转换为标准的中间语言形式,这种方法是可行的(图1),从中间语言转换到高级语言的技术是成熟的,但要覆盖各种情况,因而开发量也很大。

③ 对于不同的机型和同一种语言,可以设计抽象的中间代码,开发虚拟的反编译系统,采用目标代码预处理方法分别转换到抽象的中间代码,这种方法效果类似②。

④ 运用人工智能技术,构造反编译系统的框架,将有关机器(CPU)、语言、编译版本方面的知识填入框架,则可形成某种应用对象的反编译系统,这是目前诱人的研究方向。

⑤ 构造反编译器的编译系统,用说明语言描述反编译器的输入和输出规范,自动生成相应的反

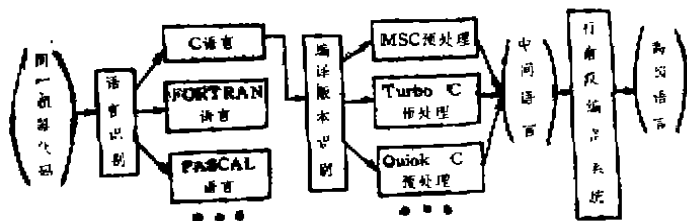


图1. 通用反编译过程示例

编译器,这是今后研究方向之一。

参考文献(略)