

软件构件

软件复用

软件开发

青鸟工程

计算机科学 1999 Vol. 26 No. 5

## 支持构件复用的青鸟Ⅲ型系统概述

An Introduction to JB3 System Supporting Component Reuse

杨美清 梅宏 李克勤 袁望洪 吴穹

(北京大学计算机科学技术系 北京 100871)

90-55

TP311.52

**Abstract** Based on former researches concerning software production industrialization technology, the task of Jade Bird Project in the National 9<sup>th</sup> Five-Year Plan is to provide technological support for national software industry. The focus is to develop industrialized software-producing system—the Jade Bird Software Production Line System, which is the software development technology and system based on component-architecture reuse. Thus, the Jade Bird Project provides a holistic solution to software development, prompts industrialized software production, and facilitates the scaling of software industry.

The JB3 system described in this article provides all-around support to software production industrialization technology based on component-architecture reuse. It proposes JB Component Model and Component Description Language, supports specialized component production, i. e., intentional component development and component acquirement from legacy system based on reengineering technology, and supports specialized architecture development with domain engineering technology. At the same time, it provides powerful component and architecture library mechanism, supports design and control of software process, as well as system composition based on component-architecture reuse.

**Keywords** Software Engineering, Software Reuse, Software Component Technology, CASE

## 一、引言

## 1.1 软件构件

软件构件技术是支持软件复用的核心技术,是近几年来迅速发展并受到高度重视的一个学科分支。其主要研究内容包括:

1. 构件获取:有目的的构件生产和从已有系统中挖掘提取构件;
2. 构件模型:研究构件的本质特征及构件间的关系;
3. 构件描述语言:以构件模型为基础,解决构件的精确描述、理解及组装问题;
4. 构件分类与检索:研究构件分类策略、组织模式及检索策略,建立构件库系统,支持构件的有效管理;
5. 构件复合组装:在构件模型的基础上研究构件组装机制,包括源代码级的组装和基于构件对象

互操作性的运行级组装;

6. 标准化:构件模型的标准化和构件库系统的标准化。

## 1.2 青鸟工程

青鸟工程是国家支持的重点科技攻关课题,历经“六五”、“七五”和“八五”,已有十余年的发展,其目标是建立我国软件产业的基础,推行软件工程化、工业化生产技术和模式,为我国软件企业提供必要的工业化生产手段和装备,为形成软件产业的规模经济,培养人才,增加科技储备,加强产品开发的能力和实力。

在国家“九五”计划中,青鸟工程的任务是在前期成果的基础上,为形成我国软件产业规模提供技术支持。重点是研究软件的工业化生产技术,针对工业化生产的需求,完善和实现软件生产线的思想,制定软件工业化生产标准,强化采用面向对象技术,支持以软件复用为基线的,基于“构件—构架”模式的

软件工业化生产技术,开发基于异构平台、可访问多信息源的应用系统集成(组装)环境青鸟Ⅱ型(JB3)系统,为软件开发提供整体解决方案,推行软件工业化生产模式,促进软件产业规模的形成。

### 1.3 青鸟Ⅱ型系统

青鸟Ⅱ型系统的研制开发是青鸟工程在“九五”期间攻关工作的一个重要组成部分,它对基于构件—构架复用的软件工业化生产技术进行全面的支 持,其中包括:提出青鸟构件模型和构件描述语言;支持专业化的构件生产,即有目的的构件开发和采用再工程技术从已有系统中获取构件;采用领域工程技术,支持专业化的构架开发;强有力的构件库、构架库管理;支持软件过程设计和控制;支持基于构件、构架复用的应用系统集成(组装)。

本文概略介绍青鸟Ⅱ型系统对软件复用的支持,基本设计思想和主要构成成分。

## 二、青鸟Ⅱ型系统

### 2.1 青鸟软件生产线

青鸟工程提出的软件生产线的概念和思想<sup>[2]</sup>,将软件的生产过程划分为三类不同的生产车间,即应用构架生产车间、构件生产车间和基于构件、构架复用的应用集成(组装)车间,从而形成软件产业内

部的合理分工,实现软件的工业化生产,青鸟软件生产线如图1所示。

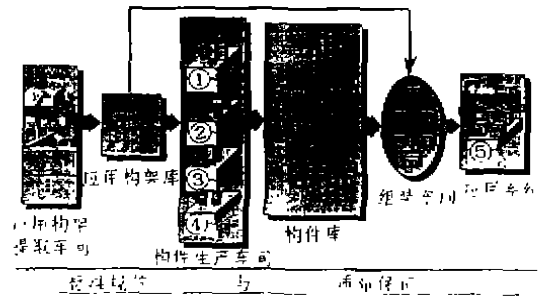


图1 青鸟软件生产线

在青鸟软件生产线中,软件开发人员被划分成三类:构件生产者、构件库管理者和构件复用者。这三种角色所需完成的任务是不同的,构件生产者负责构件的生产、描述;构件库管理者负责构件分类以及构件库的管理工作;而构件复用者负责进行基于构件的软件开发,包括构件查询、构件理解、适应性修改、构件组装以及系统演化。

### 2.2 青鸟Ⅱ型系统支持的软件开发过程

青鸟Ⅱ型系统支持的软件开发过程如图2所示,其中主要包括三个过程:①利用领域工程技术对

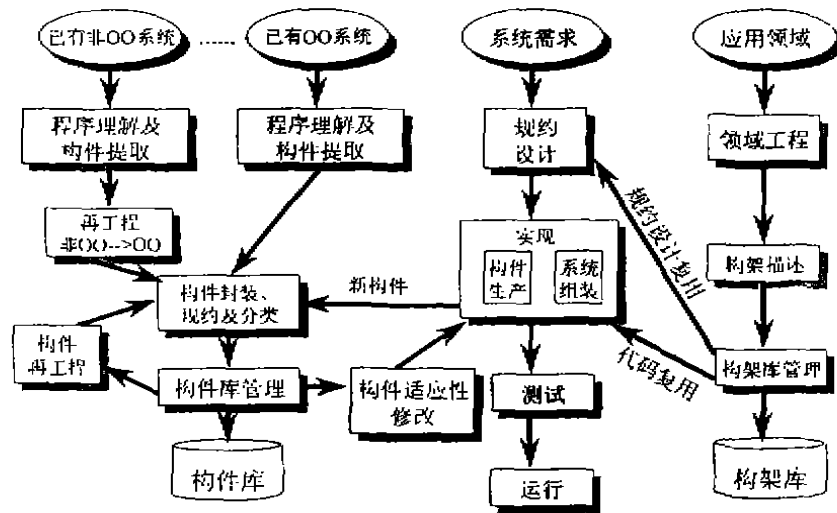


图2 青鸟软件开发过程

应用领域中的现有系统进行分析,提取特定领域的软件构架,对构架进行描述后加入构架库;②对现有系统进行程序理解和构件提取,对于非面向对象的系统还要进行向面向对象构件的再工程,对构件进行封装、规约及分类后加入构件库;③在进行应用系

统开发时,首先根据软件构架进行需求规约和设计,然后依据构架从构件库中选取构件,进行必要的适应性修改,或生产新的构件,组装成为新的系统,测试后投入运行,在这个过程中生产的新构件也可以经过封装、规约及分类后加入构件库。

### 2.3 主要组成成分

JB3 系统支持通过领域分析,获取领域构架;支持有目的的构件开发及从已有应用系统中提取可复

用构件;支持构件、构架的规约、分类组织、存储及检索;支持通过构件、构架的组装而实现应用系统的集成。

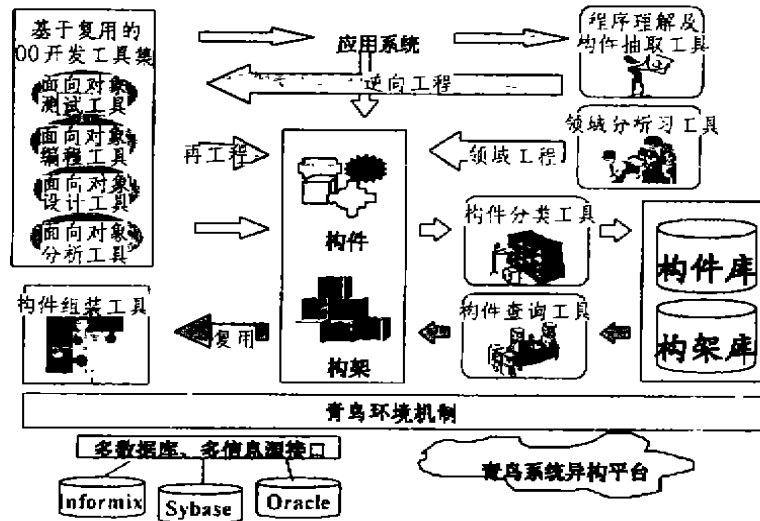


图3 青鸟III型系统的体系结构

在设计上,JB3 遵从 JB2 的环境框架<sup>[1]</sup>,继承了 JB2 环境的全部功能,其主要扩充体现为:

1. 以可复用构件、构架库为核心,支持基于复用的软件开发技术和过程。
2. 运行于异构平台之上,支持在异构网络环境下的群组式软件开发工作及不同平台上工具的互操作。
3. 具有多数据库接口,支持多信息源访问、面向多种数据库系统的应用系统集成及数据仓库管理。
4. 更灵活的环境剪裁、组装机制,使得环境可以容易地剪裁为专用应用开发平台。

JB3 系统以青鸟构件模型和青鸟构件描述语言为基础,其应用系统部分主要由以下几部分构成:青鸟异构平台;多数据库、多信息源接口;青鸟构件库系统—JBCL;青鸟程序理解及构件获取工具集;青鸟面向对象系列开发工具集;青鸟构件组装工具等。这些平台、系统和工具紧密结合,支持以复用为核心的软件开发。青鸟III型系统的体系结构如图3所示。

限于篇幅,本文将主要介绍 JB3 系统中与复用支持相关的部分。

### 三、青鸟构件模型和构件描述语言

青鸟构件模型是以对象计算模型为基础进行设

计的,它支持多种不同构件形态:类、抽象类、类簇、类树、框架(framework)和构架(architecture)。类构件、类簇构件和类树构件的引入为代码复用提供了三种不同粒度的支持。抽象类构件的引入为设计复用提供了一定的支持。框架构件的引入则对面向对象设计的复用提供了充分的支持。而构架作为一种系统级的框架,为系统级构件组装提供了支持。另外,青鸟构件模型具有自包含性,即要求构件组装而成的构件子系统仍是一个可以继续复合的构件。这使得原有的构件—构架两层结构扩展为构件—构件子系统—构架多层结构,从而拓展了该模型的表达能力<sup>[4]</sup>。图4为青鸟构件模型的示意图。

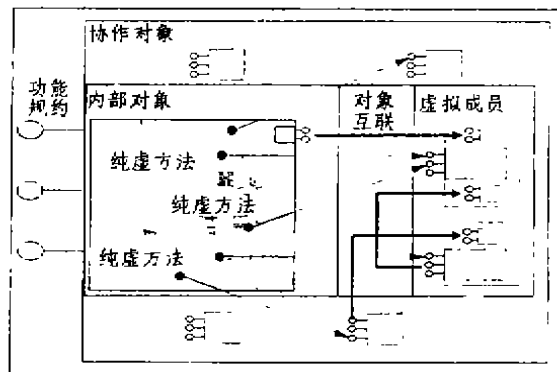


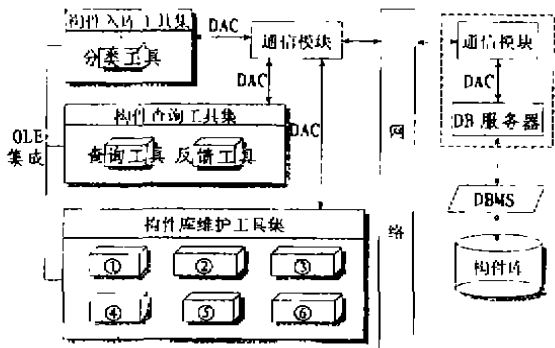
图4 青鸟构件模型

青鸟构件描述语言(JB-CDL)是在青鸟构件模型基础上设计的,其主要作用是描述构件的接口规约,以期在组装工具的支持下实现自动或半自动的构件组装。在青鸟构件描述语言中,一个构件的描述可能包括八个部分:①.规约声明部分;②.类声明部分;③.功能规约部分;④.协作对象部分;⑤.虚拟成员部分;⑥.内部对象部分;⑦.对象互联部分;⑧.纯虚方法部分,其中后六个部分与青鸟构件模型相对应。

#### 四、青鸟构件库系统—JBCL

JB3 作为一个支持复用的软件开发环境,构件的有效管理和查询是关键,其核心是一个构件库系统 JBCL。青鸟构件库系统用于对可复用构件进行描述、管理、存储和检索,以满足基于“构件—构架”复用的软件开发过程的需要。它以青鸟构件模型为基础,建立青鸟构件库数据模型,并与其它 CASE 工具相结合,支持构件的生产、描述(使用青鸟构件描述语言)、分类、存储、查询和复合<sup>[3]</sup>。

青鸟构件库提供一系列的支持工具来协助构件库的使用者和管理者共同用好构件库,发挥出构件库应用的价值。青鸟构件库系统的整体结构如图 5:



- ①用户数据库维护工具      ④反馈意见处理工具
- ②制作者数据库维护工具    ⑤术语空间维护工具
- ③管理员数据库维护工具    ⑥构件库信息统计工具

图 5 青鸟构件库系统的总体结构

JBCL 以构件库为核心,它用于储存构件及相关信息,数据库服务器直接对数据库进行操作。构件库的相关工具分为三个工具集,它们通过网络与数据库服务器通信,支持用户对构件库的使用。

1. 构件入库工具集:由构件库管理员使用,其中包括构件分类工具,主要功能是将新构件加入构件库及修改已有构件信息;
2. 构件查询工具集:由构件库用户使用,其中包括构件查询工具和构件使用意见反馈工具,协助

构件库用户浏览、查询、提取构件,反馈用户对构件的使用意见;

3. 构件库维护工具集:由构件库管理员使用,其中包括用户数据库维护工具、构件制作者数据库维护工具、构件库管理员数据库维护工具、用户反馈意见处理工具、刻面术语空间维护工具和构件库信息统计工具,对构件库的各部分内容进行维护。

青鸟构件库采用了以刻面分类策略为主,多种分类策略相结合的策略。在对构件进行分类时,除了为构件在各个刻面中关联一组术语以外,还要为构件填写属性,关联一组关键词,并在构件之间建立各种关系,相应地,也为青鸟构件库的用户提供多种查询手段,查询时除提供一组术语以外,还可以提供一些属性值,一组关键词进行查询,并可以依据构件之间的关系进行导航式查询。

#### 五、青鸟程序理解及构件获取工具集

青鸟程序理解系统 JBPA(S(Jade Bird Program Analysis System)是一个针对 C++ 语言的程序理解系统,包括一个 C++ 分析器前端和一组分析工具<sup>[3]</sup>,其体系结构如图 6 所示:

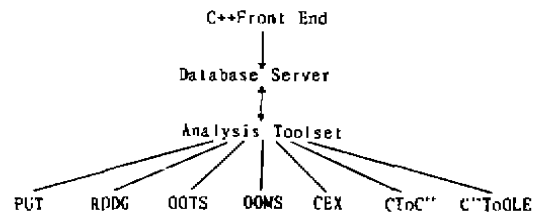


图 6 JBPA 体系结构图

分析器前端采用 EER(Enhanced Entity-Relationship)为 C++ 程序建立概念模型,该模型较为全面,以适合多种程序信息需求。通过增量分析技术静态分析程序源代码,按照概念模型抽取程序信息并将信息保存在增量数据库中,最后,启动增量库链接器,将各个增量数据库链接成程序信息库,分析器前端由信息抽取器和信息管理器组成,如图 7 所示:

分析工具包括:

1. 程序理解工具(Program Understanding Tool):支持用户在程序理解过程中采用的多种理解策略,用多层抽象视图表现程序,并方便程序理解过程中的用户交互。
2. 设计文档逆向生成工具(Reverse Design

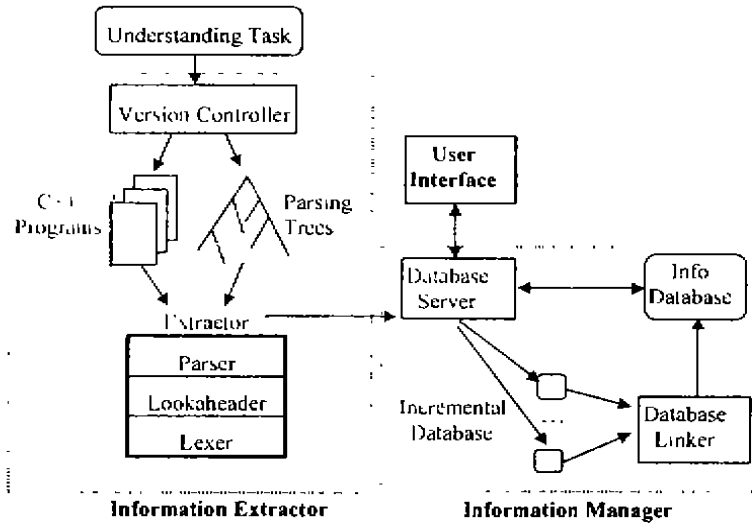


图7 JPBAS 分析器前端

Document Generator):利用程序信息库中的程序信息,自动生成程序的面向对象设计文档,以维护文档的一致性。

3. 面向对象测试支持工具(Object-Oriented Testing Supporter):利用插装技术跟踪程序的运行,以辅助测试用例的生成。

4. 面向对象度量工具(Object-Oriented Metrics Supporter):收集基于代码的度量数据并按特定的度量准则进行计算,以更好地反映程序的特性。

5. 构件提取工具(Component Extractor):基于类和类树的信息,对类和类树进行再工程以提取可复用的构件。

6. C到C++转换工具(C to C++ Translator):将C程序转换为功能等价的C++程序,以充分利用非OO程序。

7. C++到OLE转换工具(C++ to OLE Translator):辅助程序员将C++应用程序转换为遵从OLE接口标准的OLE构件。

## 六、青鸟面向对象开发工具集

JB3系统中的面向对象开发工具集(JB3/OO)以青鸟I型面向对象软件开发工具集JB2/OO为基础,在面向对象分析、设计和编程阶段中扩展了对主动对象的表示,支持各阶段产品的复用,具有多数据源接口。

软件复用的有效途径之一是采用由可复用构件组装软件的技术,即组装技术,在面向对象软件开发

各阶段产生的文档中,类是基本组成单元。类所具有的封装性和抽象性,使它作为构件具有良好的可复用性。

JB3/OO是青鸟软件开发环境中的工具集,支持对青鸟构件库JBCL的访问。可以从青鸟构件库里查找和提取合适的构件用于开发各阶段的建模。

JB3/OO还提供了类库管理器,用于建立和管理本地类库,支持本地平台的复用。本地类库具有以下特点:

1. 易于使用和管理。类库仅支持用于面向对象开发的可复用构件——类构件的管理,其用途较为单一,因此,概念简单,易于与面向对象工具集成。另一方面,本地类库完全在开发人员自己控制之下,类构件的入库和提取十分方便。

2. 支持构件的开发。软件构件与普通的软件一样需要经历版本更新的过程,不断地成熟、演化。可复用的构件尤其如此。当构件开发人员对应用领域了解不深时,以及当构件没有在开发实践中不断使用和演变时,是很难得到具有较高复用价值的构件的。类库的本地特性和易用性使得开发人员便于面向领域开发类构件,并且不断地改进、完善;当构件成熟后,即可作为标准构件提交给青鸟构件库。

## 七、青鸟构件组装工具

青鸟构件组装工具是一个可视化的工具,支持用户以图形方式描述构件及其连接关系,然后将图形表示转化为青鸟构件描述语言,实现构件组装,图

8 中给出了青鸟构件组装工具所对应的 Use Case 图。

青鸟构件组装工具的主要功能包括:

1 编辑构件的 CDL 描述。利用青鸟构件组装工具,用户可以以图形和文本两种方式编辑构件的 CDL 描述,并可以自由地在两种方式之间切换。在编辑完成之后,该工具将首先进行语法检查,然后根据一定的原则来进行语义完整性验证,并将该 CDL 描述以文本形式保存起来;

2 根据构件的 CDL 描述来生成 C++ 代码骨架。如果一个用户试图开发新的构件,那么在编辑构件的 CDL 描述之后,青鸟构件组装工具可以自动生成相应的代码骨架;

3 构件组装。依据青鸟构件模型,构件组装必须是在某个框架或构架的基础上完成的,实际上,构件组装可以看成是框架构件或构架的具体化过程,即将构件制作过程中抽象掉的信息重新由复用者按照他们自己的意愿添加到构件中;

4 利用分布式对象技术来实现构件在网络上的分布。利用青鸟构件组装工具,用户可以很简便地利用分布式对象技术(CORBA 或 DCOM)来实现构件在网络上的分布。

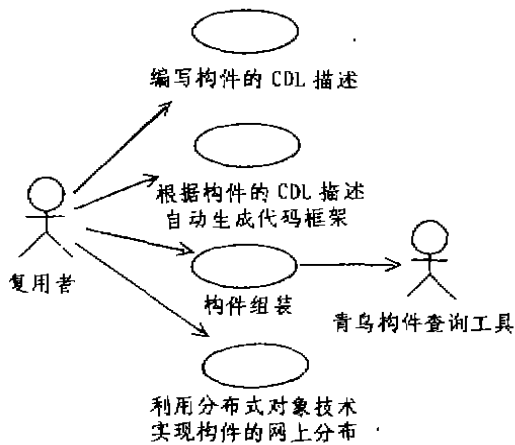


图 8 构件组装工具的 Use Case 图

结论:本文对支持基于“构件—构架”复用的软

件开发技术的 JB3 系统进行了概略的介绍,对 JB3 系统的几个关键部分如:青鸟构件模型、青鸟构件库管理系统、程序理解及构件获取工具、面向对象工具、构件组装工具进行了讨论。

JB3 系统是青鸟软件生产线思想的具体实现,体现了软件工业化生产的方法和技术,它的完成将会对我国软件开发方式的变革起到积极的推动作用,为我国软件企业提供良好的技术装备。

JB3 系统的设计和实现全面采用了面向对象技术,并使用 JB2 环境来支持其开发工作。在开发过程中,充分考虑了系统的构件化问题,为系统的进一步再工程和演化奠定了良好基础。

JB3 系统的研制工作仍在进行中。

### 参考文献

- 1 杨美清,邵维忠,梅宏. 面向对象 CASE 环境 JB I 型系统的设计和实现. 中国科学,1995(5)
- 2 杨美清. 青鸟过程现状与发展—兼论我国软件产业发展途径. 见:杨美清,何新贵编. 第六次全国软件工程学术会议论文集,软件工程进展—技术、方法和实践. 北京:清华大学出版社,1996
- 3 Mei Hong, et al. BDCOM-C++: AC++ Program Understanding System, Chinese Journal of Electronics, 1997, 6(2): 64~69
- 4 Wu Qiong, et al. JBCDL: An Object-Oriented Component Description Language. In: Jian Chen, et al. eds. Proc. of the twenty-fourth Intl. Conf. TOOLS ASIA, Beijing, 1997, 247~253
- 5 Li Kequn, et al. An Overview of JB(Jade Bird) Component Library System JBCL. In: Jian Chen, et al. eds. Proc. of the twenty-fourth International Conference TOOLS ASIA, Beijing, 1997, 261~267
- 6 Mih H, et al. Reusing Software: Issues and Research Directions, IEEE Transactions on Software Engineering, 1995, 21(6)