The Version Management Architecture of an Object-Oriented Distributed Systems Environment: OZ++

Michiharu Tsukamoto¹, Yoichi Hamazaki¹, Toshihiro Nishioka², and Hideyuki Otokawa³

¹ Electrotechnical Laboratory, 1-1-4 Umezono, Tsukuba, Ibaraki 305, Japan
² Mitsubishi Research Institute, Inc., 2-3-6 Otemachi, Chiyoda-ku, Tokyo 100, Japan
³ Sharp Corporation, 2613-1 Ichinomoto, Tenri, Nara 632, Japan

Abstract. Internet based information retrieval software attracts attention because of potentially great impact on our society brought about by its usefulness in information dispatch, sharing, and distribution. However, it is difficult to share, distribute, and reuse of software on wide-area network environment. OZ++ system is a software system to conquer this problem. Based on the concept of object-orientation, the system provides automatic distribution and upgrading function of software over networks. Using the system, software can be brought together from all over the network; furthermore, it has become possible to run such software immediately. This paper introduces the version management mechanism of the most attractive feature of OZ++. Its version management is based on the interface of classes.

1 Introduction

Internet based information retrieval software attracts attention because of Internet's potentially great impact on our society brought about by its usefulness in information dispatch, sharing, and distribution. However, many software developers remake and reintroduce similar softwares the world over, and this makes difficult to improve productivity and software reliability. It is considered that mutual sharing and reuse of software through Internet will improve the quality of shared and reused software, and enable us to concentrate our efforts on the new software development. That is, we expect not only data sharing, distributing and reusing on Internet, but also development of the software system, which will enable program sharing, distribution and reuse.

1.1 Object-Oriented Approach to the problems

The object-oriented approach is considered to be the most suitable approach for realizing a software system that enables sharing, distribution and reuse of programs because of its ability to devise parts and reutilize software resources. The development and execution methods based on this approach are as follows:

* Researcher, Tsukuba Laboratory, Open Fundamental Software Technology Project
- Development phase
  On the network environment, programming with a class sharing function, inter-utilizing and/or customizing the opened class library if necessary, will be enabled.
- Execution Phase
  When receiving a class object made like this, whenever the need arises, the customized class is obtained and executed by a function of class distribution.

For smooth implementation of this kind of development and execution, however, requires as follows:
- If various customizations exist, it does not matter if a class uses an old version.
- On customizing the class, only necessary parts are recompiled.
- When a program is executed, various coexisting class versions can be executed at the same time.

1.2 Contents of this Paper

OZ++ is a software system that meets these requirements. It operates in a distributed object-oriented environment that takes into consideration the wide-area network environment consisting of the OZ++ language system, an executing system, a management system and the development environment[1],[2],[3]. The characteristic functions are class sharing, class distribution, class version management and class certification function in the network environment. The prelemeny evaluation of OZ++ will be published in [4].

This paper discusses the concepts of class, version management systems and version propagating systems on the OZ++ system. Section 2 presents a discussion of related work in distributed systems, Section 3 discusses class sharing and upgrading, Section 4 introduces class to OZ++, Section 5 explains interface and version, Section 6 mentions the information managed by the object management system, Section 7 describes the basic version management system and Section 8 deals with the version distribution method.

2 Related Works

Some former studies of programming to utilize object-oriented languages include Eden[5], Argus[6], Emerald[7], Distributed Smalltalk[8] and OZ+[9]. They are however, basic studies because they applied object-based language on small-scale networks such as LAN, or aimed at design and execution systems for object-oriented languages.

OZ++ language is multiple-inheritance object-oriented languages which has week type system. OZ++ doesn’t use uniform objects model as in Emerald and Smalltalk but network-wide global objects and local objects model as in Argus and Eden. Because many of objects has locally of access and the performance of access to global objects and local objects differs greatly in large-scale network