Mapping of CGIF to Operational Interfaces

A. Puder

International Computer Science Institute
1947 Center St., Suite 600
Berkeley, CA 94704-1198
USA
puder@icsi.berkeley.edu

Abstract. The Conceptual Graph Interchange Format (CGIF) is a notation for conceptual graphs which is meant for communication between computers. CGIF is represented through a grammar that defines "on-the-wire-representations". In this paper we argue that for interacting applications in an open distributed environment this is too inefficient both in terms of the application creation process as well as runtime characteristics. We propose to employ the widespread middleware platform based on CORBA to allow the interoperability within a heterogeneous environment. The major result of this paper is a specification of an operational interface written in CORBA's Interface Definition Language (IDL) that is equivalent to CGIF, yet better suited for the efficient implementation of applications in distributed systems.

Keywords: CGIF, CORBA, IDL.

1 Introduction

Conceptual Graphs (CG) are abstract information structures that are independent of a notation (see [5]). Various notations have been developed for different purposes (see Figure 1). Among these are the display form (graphical notation) or the linear form (textual notation). These two notations are intended for human computer interaction. Another notation called Conceptual Graph Interchange Format (CGIF) is meant for communication between computers. CGIF is represented through a grammar that defines "on-the-wire-representations" (i.e. the format of the data transmitted over the network).

The reason for developing CGIF was to support the interoperability for CG-based applications that needed to communicate with other CG-based applications. We argue that for interacting applications in an open distributed environment this is too inefficient both in terms of the application creation process as well as runtime characteristics. Applications that need to interoperate are written by different teams of programmers, in different programming languages using different communication protocols.

A generalization of this problem is addressed by so-called middleware platforms. As the name suggests, these platforms reside between the operating system and the application. One prominent middleware platform is defined through the Common Object Request Broker Architecture (CORBA) which allows the interoperability within a heterogeneous environment (see [4]). In this paper we will show how to use CORBA for CG-based applications.
The outline of this paper is as follows: in Section 2 we give a short overview of CORBA. In Section 3 we discuss some drawbacks of using CGIF for distributed applications. In Section 4 we present our mapping of CGIF to CORBA IDL, which is further explained in Section 5 through an example. It should be noted that we describe work-in-progress. The following explanations emphasize the potential of using CORBA technology for CG-based applications. A complete mapping of CGIF to CORBA IDL is subject to further research.

2 Overview of CORBA

Modern programming languages employ the object paradigm to structure computation within a single operating system process. The next logical step is to distribute a computation over multiple processes on a single machine or even on different machines. Because object orientation has proven to be an adequate means for developing and maintaining large scale applications, it seems reasonable to apply the object paradigm to distributed computation as well: objects are distributed over the machines within a networked environment and communicate with each other.

As a fact of life, the computers within a networked environment differ in hardware architecture, operating system software, and the programming languages used to implement the objects. That is what we call a heterogeneous distributed environment. To allow communication between objects in such an environment, one needs a rather complex piece of software called a middleware platform.

The Common Object Request Broker Architecture (CORBA) is a specification of such a middleware platform. The CORBA standard is issued by the Object Management Group (OMG), an international organization with over 750 information software vendors, software developers, and users. The goal of the OMG is the establishment of industry guidelines and object management specifications to provide a common framework for application development. CORBA addresses the following issues: