

# HoloJava: Translating a Distributed Multiparadigm Language into Java

Jorge Luis Victória Barbosa<sup>1,3</sup>, André Du Bois<sup>1</sup>,  
Altino Pavan<sup>2</sup>, Cláudio Fernando Resin Geyer<sup>3</sup>

<sup>1</sup> Catholic University of Pelotas, Computer Science Department  
Pelotas, RS, Brazil  
{barbosa, dubois}@atlas.ucpel.tche.br

<sup>2</sup> Godigital Information Technology, Inc  
Porto Alegre, RS, Brazil  
altino@godigital.com.br

<sup>3</sup> Federal University of Rio Grande do Sul, Informatics Institute  
Porto Alegre, RS, Brazil  
{barbosa, geyer}@inf.ufrgs.br

## ABSTRACT

In this paper we describe the design and implementation of a system for translating Holo into Java. The system is called HoloJava. Holo is a new multiparadigm language for the development of distributed systems. This language is based on concepts of concurrency, blackboards and mobility. Besides that, Holo integrates aspects of logic, imperative and object oriented paradigms. Java is well known by being a portable and network-aware object oriented language. We propose the use of Java as an intermediate language in our development environment. There are Holo aspects that we do not translate directly to pure Java. We use special libraries to support these aspects. Specifically, logic programming characteristics are translated using JIProlog, blackboards are implemented using Jada tuple spaces and network mobility is supported using Voyager.

Keywords: Multiparadigm, Mobility, Programming Languages and Distributed Systems.

## 1 INTRODUCTION

In the last years the multiparadigm theme has been continually researched [2, 3, 4, 6, 18, 27, 28, 31]. Researchers have proposed models of software development through the integration of basic paradigms (mainly: imperative, logic, functional, and object oriented paradigms). Using this approach, they have been looking for two goals: (a) to overcome the specific limitations of each paradigm and (b) to take advantage of the most useful characteristics of each one through their combination.

Each paradigm has sources of implicit parallelism. For example, the exploitation of AND parallelism and OR parallelism in logic programming [5, 16, 36]. Another example is object-oriented paradigm that allows the exploitation of inter-object parallelism and intra-object parallelism [12, 29]. The multiparadigm approach integrates paradigms. Thus, it also integrates their parallelism sources. In this context, interest in automatic exploitation of parallelism in multiparadigm software has emerged. Enlargement of this approach guides the studies to distributed systems where mobility, heterogeneous hardware and the use of networks as parallel architectures are considered. The development of distributed software using multiparadigm models has received attention of the scientific community [6, 12, 20, 21, 29, 34].

In this context, we are creating the *Holoparadigm (Holo)*. Holo is a multiparadigm model oriented to automatic exploitation of parallelism and distribution. Holo has a coordination model based on levels of blackboards (called *histories*) encapsulated in new programming entities called *beings*. A new language (*Hololanguage* [6]) implements the concepts proposed by the Holoparadigm.