

# Generator and Search Objects in Java

Lloyd Allison

School of Computer Science  
Monash University  
Australia 3168  
<http://www.csse.monash.edu.au/>

*Generator classes are defined in the object oriented programming language Java by using continuation-style programming. Generator objects are used to write Prolog-like programs in Java to solve combinatorial constraint satisfaction problems. A collection of generators oriented to world wide web applications is demonstrated on searches for HTML pages.*

*Keywords: continuation, generator, constraint satisfaction problem, combinatorial search, world wide web search, Java*

## 1. INTRODUCTION

A continuation is a function (or procedure or method) used to represent a following computation. Continuations arose in denotational semantics as a way of defining the meaning of jumps and other sequencers in imperative programming languages. Here they are used in the object oriented programming language Java to implement combinatorial generators for constraint satisfaction problems and to write Prolog-like searches for world wide web pages.

The easiest languages in which to do continuation style programming are the lazy functional languages such as Haskell (Hudak *et al*, 1992). The applicative language Scheme (Hayes *et al*, 1986) even has continuations built into it as a predefined type. However, continuations can also be programmed in an imperative language such as C or Pascal if it is possible to pass a procedure as a parameter. The object oriented language Java does not allow procedures as parameters but references to *objects* can be passed as parameters and it is possible to define generator and continuation classes. These classes can be conveniently used in a Java application program or in an applet, if a task includes a search for the solution of a combinatorial problem, without having to switch to a non-deterministic language.

Continuations were first used to define jumps and other sequencers by Strachey and Wadsworth (1974) and Milne (1976). Strachey and Wadsworth attribute the origin of the idea to Mazurkiewicz's tail functions (Mazurkiewicz, 1971). Strachey, Wadsworth and Milne demonstrated that continuations could define arbitrary control mechanisms in programming languages. Their power has been used in semantics of Prolog (Nicholson and Foo, 1989; Finlay and Allison, 1993).

This paper demonstrates that continuations can be used in an object oriented programming language such as Java. It uses the technique to implement Java classes that can be used to write Prolog-like programs in Java to solve combinatorial constraint problems and, by also making use of

---

*Copyright© 2000, Australian Computer Society Inc. General permission to republish, but not for profit, all or part of this material is granted, provided that the JRPIT copyright notice is given and that reference is made to the publication, to its date of issue, and to the fact that reprinting privileges were granted by permission of the Australian Computer Society Inc.*

*Manuscript received: August 1999*  
Associate Editor: Janet Smith