Summary: Invocation servicing is an important aspect of many concurrent programming languages. Some invocation handling mechanisms allow for multiway servicing by multiple processes. This paper addresses fairness with respect to choosing which invocation to service and fairness with respect to choosing which process to perform the servicing. It examines how these fairness issues have been resolved in the SR concurrent programming language. This paper presents a new approach that eliminates several key restrictions. The new approach has been implemented in JR, an extended Java that includes SR-like synchronization mechanisms. This paper discusses design and implementation issues and tradeoffs.

MSC:
68N19 Other programming paradigms (object-oriented, sequential, concurrent, automatic, etc.)

Keywords:
Concurrent programming languages; Invocation servicing; Invocation handling; Fairness; Language design; Language implementation

Software:
Ada95; Linda

Full Text: DOI

References:
[18] Elrad T, Maymir-Ducharme F. Satisfying emergency communication requirements with dynamic preference control. In Pro-


This reference list is based on information provided by the publisher or from digital mathematics libraries. Its items are heuristically matched to zbMATH identifiers and may contain data conversion errors. It attempts to reflect the references listed in the original paper as accurately as possible without claiming the completeness or perfect precision of the matching.