ICAP PATENT BROKERAGE

Evaluation of Programming Language Statements

Xerox Corporation

Initial Bidding Guidance: Low 6 Figures

Within the field of programming, declarative and imperative techniques are both acceptable forms of directing software operations, each with costs and benefits. Desirable is a combination of both techniques, which allows concise and declarative notations while reserving fine-grain imperative specifications for difficult cases. This portfolio is directed to methods and systems for evaluating a programming language statement capable of dealing with arbitrary structural complexity while preserving relevant type control and keeping the amount of basic building blocks reasonably small. It is also specific enough, and possessive of defined syntax and semantics, such that it can be embedded into future languages.

The technique can serve to build transformation models which are less abstract and more general than rewrite systems, which perform implicit pattern matching and apply built-in strategies for rule application. The technique is abstract enough to simplify and extend general purpose programming languages, and is important in the design of new transformation techniques or in the extension of existing ones.

Four basic topics are addressed in a single approach: the definition of a data model, the definition of matching operations, the definition of a transformation model, and the integration of the transformation model in a programming language.

Forward Citing Companies: Bea Systems, IBM, Landmark Graphics, Microsoft, NEC Laboratories, Oracle, Siemens AG

Priority Date: 01-16-2002

Representative Claim: US 7,543,015 – Claim #1

A program stored on a computer-readable medium that causes a computer to execute steps to determine whether a computer-storable expression matches a filter, comprising: identifying and testing the structural form of the expression; choosing a transformation model for the filter that is compatible with the structural form of the expression; evaluating a first code structure representing the expression to determine a value of said expression prior to filtering; analyzing a second code structure representing the filter to determine the characteristics of the filter, wherein the second code structure comprises a plurality of filter characteristics corresponding to a plurality of structural forms of the first code structure, the plurality of structural forms comprising text, trees, and graphs, and wherein the second code structure applies a filter corresponding to the structural form of the first code structure; and filtering said evaluated value according to the filter characteristics, wherein said first code structure is constructed from a plurality of first programming language code structure elements and said second code structure is constructed from a plurality of second programming language code structure elements, each second structure element being symmetrically constructed to correspond to one of said first structure elements, and wherein evaluating, analyzing and filtering are performed upon explicit invocation of a matching operator, and filtering comprises returning a boolean evaluation result value.

Contact:
For more information on the assets available for sale in this portfolio, contact Paul Greco.
Paul Greco
Senior Vice President
Paul@icapip.com
(212) 815-6692

TECHNOLOGY

COMBINATORS FOR PROGRAMMING LANGUAGE BASED UPON STRUCTURAL PATTERN-MATCHING

NOVELTY

METHOD AND SYSTEM FOR EVALUATING A PROGRAMMING LANGUAGE STATEMENT CAPABLE OF DEALING WITH ARBITRARY STRUCTURAL COMPLEXITY, THUS ENABLING PATTERN-MATCHING OPERATIONS ON ARBITRARY COMPLEX DATA STRUCTURES WITH MINIMAL BASIC BUILDING BLOCKS. BY USING IMPERATIVE CONNECTORS, THE DESIGN OF NEW PROGRAMMING LANGUAGES OR THE EXTENSION OF EXISTING PROGRAMMING LANGUAGES ARE RENDERED POSSIBLE WITH CONSTRUCTIONS THAT PROVIDE INNOVATIVE EXPRESSIVENESS. RESULTING LANGUAGES ARE LOCATED AT AN INTERMEDIATE LEVEL OF ABSTRACTION, BETWEEN DECLARATIVE, FUNCTIONAL, AND IMPERATIVE LANGUAGES

IMPORTANCE

A VALUABLE PORTFOLIO FOR COMPANIES PRODUCING PROGRAMMING LANGUAGES AND PATTERN MATCHING SYSTEMS

NUMBER OF ASSETS

2

US PATENTS (2)

7,240,331
7,543,015