



US006345387B1

(12) **United States Patent**  
**Morrison**

(10) **Patent No.:** **US 6,345,387 B1**  
(45) **Date of Patent:** **Feb. 5, 2002**

(54) **COHERENT OBJECT SYSTEM ARCHITECTURE**

(75) Inventor: **Gordon E. Morrison**, Westminster, CO (US)

(73) Assignee: **COSA Technologies, Inc.**, Denver, CO (US)

(\*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **09/302,940**

(22) Filed: **Apr. 30, 1999**

**Related U.S. Application Data**

(60) Provisional application No. 60/083,779, filed on Apr. 30, 1998.

(51) **Int. Cl.**<sup>7</sup> ..... **G06F 9/445**

(52) **U.S. Cl.** ..... **717/11; 717/1; 717/2; 717/5**

(58) **Field of Search** ..... **717/1, 11, 2, 5; 709/318; 714/38; 700/86**

(56) **References Cited**

**U.S. PATENT DOCUMENTS**

5,339,430	A	8/1994	Lundin et al.	709/332
5,590,271	A	12/1996	Klinker	395/763
5,598,564	A *	1/1997	Barker, III	717/5
5,692,122	A *	11/1997	Bird	714/38
5,983,016	A *	11/1999	Brodsky et al.	717/1
5,995,753	A *	11/1999	Walker	717/2
6,038,378	A *	3/2000	Kita et al.	395/183.14
6,138,171	A *	10/2000	Walker	709/318
6,167,319	A *	12/2000	Harris et al.	700/86
6,269,049	B1 *	7/2001	Weinfrutner	365/233

**FOREIGN PATENT DOCUMENTS**

EP 0 889 401 A2 7/1999

**OTHER PUBLICATIONS**

Title: Transforming control-flow intensive designs to facilitate power management, author: Lakshminarayana et al., ACM., 1998.\*

Title: Improving Data-Flow analysis with Path Profiles, author: Ammons et al, ACM, 1998.\*

Title: An overview of hierarchical control flow graph model, author: Fritz et al, ACM, 1995.\*

McConnell, Steven C., "Code Complete: A Practical Handbook of Software Construction" (Microsoft Press, 1993.)

\* cited by examiner

*Primary Examiner*—Mark R. Powell

*Assistant Examiner*—Chameli C. Das

(74) *Attorney, Agent, or Firm*—Blakely, Sokoloff, Taylor & Zafman LLP

(57) **ABSTRACT**

A method and apparatus are provided for controlling the execution of a software program. According to one embodiment, control flow information including multiple states associated with a software program is made accessible. Each of the states includes information indicative of desired control flow of the software program and information indicative of desired behavior of the software program. A determination is made if a current status of the software program has a predetermined relationship with an expected state of execution of the software program. Based upon the result of the determination and the information indicative of desired behavior, the software program is caused to perform an action and caused to transition from the current state to a next state. According to another embodiment, specification of a control flow architecture associated with a software program is simplified to the provision of certain control flow information that is accessible to a control flow engine. The control flow information need only include information indicative of desired control flow of the software program and information indicative of desired behavior of the software program.

**72 Claims, 15 Drawing Sheets**

