

THIS OPINION WAS NOT WRITTEN FOR PUBLICATION

The opinion in support of the decision being entered today (1) was not written for publication in a law journal and (2) is not binding precedent of the Board.

Paper No. 16

UNITED STATES PATENT AND TRADEMARK OFFICE

---

BEFORE THE BOARD OF PATENT APPEALS  
AND INTERFERENCES

---

Ex parte NURCAN COSKUN  
and BRUCE A. TATE

---

Appeal No. 1997-4128  
Application 08/354,699<sup>1</sup>

---

ON BRIEF

---

Before URYNOWICZ, JERRY SMITH and FLEMING, Administrative Patent Judges.

URYNOWICZ, Administrative Patent Judge.

Decision on Appeal

This appeal is from the final rejection of claims 1-21, all the claims pending in the application.

The invention pertains to a method of debugging a program in a graphic user interface. Claim 1 is illustrative and reads as follows:

---

<sup>1</sup> Application for patent filed December 13, 1994.

Appeal No. 1997-4128  
Application 08/354,699

A method in a data processing system for debugging an object from a plurality of objects forming an application in an object oriented system utilizing a graphic user interface, wherein a number of the plurality of objects each includes at least one action slot, each action slot containing at least one action object, the method comprising the data processing system implemented steps of:

storing each action object within an action slot in an activation object in response to an event associated with the action slot generated by a user utilizing the graphic user interface; and

storing data sent to each action object in the activation object, the data being data required to recreate the event, wherein action objects responsive to the event may be debugged.

The references relied upon by the examiner as evidence of obviousness are:

Coplien et al. (Coplien)                      5,093,914                      Mar. 03,  
1992

Padawer et al. (Padawer)                      5,124,989                      Jun. 23,  
1992

The appealed claims stand rejected under 35 U.S.C. § 103 as being unpatentable over Coplien in view of Padawer.

The respective positions of the examiner and the appellants with regard to the propriety of these rejections are set forth in the final rejection (Paper No. 8) and the examiner's answer (Paper No. 14), and the appellants' brief (Paper No. 13).

Appellants' Invention

Appeal No. 1997-4128  
Application 08/354,699

The invention is adequately described at pages 2-5 of the brief.

The Rejection under 35 U.S.C. §103

At page 5 of the brief, appellants state that for purposes of this appeal, the claims will be grouped together.

With respect to Padawer, appellants argue that the reference is not to analogous art and may not be used to form the basis of the rejection. The claimed invention relates to debugging object-oriented programs using objects. The reference contemplates debugging programs using a command-line program. In contrast, appellants are concerned with object-oriented programs. Command-line programs are at opposite ends of the spectrum, employing widely disparate approaches and conceptual foundations, and operating differently. It is argued that command-line programming techniques are not reasonably pertinent to solving problems arising in object-oriented programming. Appellants contend that those skilled in object-oriented programming would not look to command-line programming techniques to solve problems encountered in object-oriented programs, particularly problems arising from program execution.

Appeal No. 1997-4128  
Application 08/354,699

The argument is made that nothing suggests that a history tape for debug commands as disclosed by Padawer may be combined with an object as disclosed in Coplien to achieve the functionality of the claimed invention which calls for saving the context of a user event.

Appellants urge that none of the references discloses or suggests employing action objects, action slots and activation objects to visually debug an object in a object-oriented program. Each of an action object, action slot and activation object requires a software object in an object oriented environment, not merely a function module or routine as disclosed in Coplien or a simple record as taught in Padawer. It is urged that an activation object stores the context of the user event, and that this feature is not taught or suggested by the references.

The examiner argues that Padawer is analogous prior art because it is in appellants' field of endeavor, which is debugging computer programs. The position is taken that command-line programming techniques, such as recording "records to permit each debug step to be precisely reproduced" (Padawer col. 5, lines 12-15), are demonstrated as object oriented programming techniques in

Appeal No. 1997-4128  
Application 08/354,699

appellants' claim limitations for storing "data required to recreate the event".

With respect to appellants' argument that there is no suggestion or motivation to combine the references, the examiner states "On the contrary, Padawer et al. suggests combining so that the user can, trace or debug, and verify the accuracy of a program's execution." The examiner contends that the combination would have a high expectation of success since the events recorded in Padawer are breakpoints which are described in both references associated with controlling a program.

The examiner contends that Coplien discloses action objects, such as the functions move, refresh or create in Figure 5, where the methods that correspond to these action objects are shown as "refresh(){...}" etc., or functions illustrated in Figure 4. It is urged that "Coplien et al. show an action slot in figure 5 for a class window which defines events, such as, move, refresh and create, to be performed by either of the XWindow or SunviewWindow". The examiner states that Padawer teaches activation objects. In support of this position, the examiner asserts that in column 9, lines 40-50, Padawer discloses that debug commands stored as a record in debug tape 302 can activate

Appeal No. 1997-4128  
Application 08/354,699

the execution events for debugging a program as a re-execute function, and that column 4, lines 12-17 and lines 27-32, shows the record structure.

After consideration of the positions and arguments presented by both the examiner and the appellants, we have concluded that the rejection should not be sustained.

The examiner's position that Padawer suggests combining the references "so that the user can, trace or debug, and verify the accuracy of a program's execution" is not persuasive. Coplien teaches debugging and verifying the accuracy of a program execution (column 1, lines 42-44). Thus, one of ordinary skill in the art would not have combined Padawer with Coplien so that the user of Coplien can "trace or debug, and verify the accuracy of a program's execution" because Coplien teaches debugging and verifying.

Furthermore, although we agree with the examiner that Coplien teaches action objects and action slots, we agree with appellants that Padawer does not disclose or suggest activation objects.

It is clear that Coplien is concerned with object-oriented programs (column 3, lines 3-10) and discusses objects (column 5, lines 61-64 and column 8, lines 57-68). In discussing related art

Appeal No. 1997-4128  
Application 08/354,699

at page 4, lines 6-11, of their specification, appellants acknowledge that an object in object-oriented programming refers to methods (such as resize), data (such as color) and events (such as button-was-pressed). With respect to appellants' Figure 3, resize is the "Event" in an action slot 302, color is the "obj" (data) in an action object 306a within the action slot 302, and button-was-pressed is the method "m1" in the action object 306a within the action slot 302. Because Coplien discusses objects and appellants acknowledge that an object in object-oriented programs of the prior art includes the above three elements (event, obj and method), it is clear that Coplien discloses action objects and action slots.

At page 17, lines 24-27, of their specification, appellants disclose that an activation object is an object that contains context required to reproduce the event triggering the action slot and an action slot. At page 16, line 30, of their specification, appellants further disclose that context is information required to recreate an event. Padawer discloses context in that the debug history tape contains data used to re-execute debug commands, that is, recreate an event. However, Padawer discloses no action slot and, contrary

Appeal No. 1997-4128  
Application 08/354,699

to the examiner's position, cannot disclose an activation object because an activation object requires context and an action slot according to appellants' definition of the term. Accordingly, even if there existed some motivation or suggestion to combine the teachings of Coplien and Padawer, the subject matter of the claims would not have been met by the combination.

Although we will not sustain the rejection of the claims, the examiner's position that Padawer is analogous prior art because both Coplien and Padawer relate to debugging computer programs is reasonable and persuasive. Appellants have submitted no evidence to rebut the examiner's rationale and in support of its position that command-line programming techniques are not reasonably pertinent to solving problems arising in object-oriented programming. Relevant prior art includes that reasonably pertinent to the particular

Appeal No. 1997-4128  
Application 08/354,699

program with which the inventor was involved. In re GPAC, Inc.,  
57 F.3d 1573, 1577, 35 USPQ2d 1116, 1120 (Fed. Cir. 1995).

REVERSED

STANLEY M. URYNOWICZ, JR.	)	
Administrative Patent Judge	)	
	)	
	)	
	)	
	)	
	)	BOARD OF PATENT
JERRY SMITH	)	
Administrative Patent Judge	)	APPEALS AND
	)	
	)	INTERFERENCES
	)	
	)	
	)	
	)	
MICHAEL R. FLEMING	)	
Administrative Patent Judge	)	

Appeal No. 1997-4128  
Application 08/354,699

SMU/kis

Andrew J. Dillon  
FELSMAN, BRADLEY, GUNTER  
& DILLON, LLP  
Suite 350, Lakewood on the Park  
7600B North Capital of Texas Highway  
Austin, TX 78731