

The opinion in support of the decision being entered today was not written for publication and is not binding precedent of the Board.

Paper No. 13

UNITED STATES PATENT AND TRADEMARK OFFICE

---

BEFORE THE BOARD OF PATENT APPEALS  
AND INTERFERENCES

---

Ex parte JASON YAOHUA CAI, ANTHONY HASSITT, TSUNEO HORIGUCHI,  
KIMBERLY MARIE KETELL and MARILYN ZUPSICH SMITH

---

Appeal No. 2002-0695  
Application No. 09/132,731

---

ON BRIEF

---

Before KRASS, FLEMING and BLANKENSHIP, Administrative Patent Judges.

KRASS, Administrative Patent Judge.

DECISION ON APPEAL

This is a decision on appeal from the final rejection of claims 1-42, all of the pending claims.

The invention is directed to a method, apparatus and article of manufacture for supporting file input/output (I/O) operations to different file-systems through applications written in a high-level computer language. Source program programming

language statements are received and at least one of these received statements

includes a file-system identifier that permits selection of a file system for performing input and output operations in the source program.

Representative independent claim 1 is reproduced as follows:

1. A method of programming a computer, comprising the steps of:
  - (a) receiving a series of programming language statements comprising a source program into a memory of the computer, wherein at least one of the received statements includes a file-system identifier that permits selection of a file system for performing input and output operations in the source program and the file-system identifier is selected from a group comprising a literal, an environment variable, and a data name; and
  - (b) compiling the source program into an object program in the memory of the computer, wherein the object program includes instructions for performing the input and output operations according to the file-system identifier that permits the selection of the file system.

The examiner relies on the following references:

Archer et al. (Archer)	4,587,628	May 06, 1986
Coker	5,640,550	Jun. 17, 1997
Cai et al. (Cai)	5,819,276	Oct. 06, 1998
		(filed Oct. 06, 1995)

Claims 1-42 stand rejected under obviousness-type double patenting over claims 1-39 of Cai.

Claims 1-9, 15-24 and 30-37 stand rejected under 35 U.S.C. § 103 as unpatentable over Coker in view of Archer.

Reference is made to the briefs and answer for the respective positions of

appellants and the examiner.

### OPINION

At the outset, we note that appellants do not argue the rejection of claims 1-42 based on obviousness-type double patenting, stating only that this rejection is “not being appealed, and a Terminal Disclaimer will be submitted after a decision on appeal. Accordingly, we will summarily sustain the rejection of claims 1-42 under obviousness-type double patenting.

Turning now to the rejection of claims 1-9, 15-24 and 30-37 under 35 U.S.C. § 103, we will not sustain this rejection as, in our view, the examiner has not established a prima facie case of obviousness.

It is the examiner’s position that Coker discloses the claimed subject matter but for explicitly teaching how the file system identifier is selected, i.e., from a group comprising an assignment literal, environment variable, data name. The examiner turns to Archer for a teaching of “selective control of I/O responsive to the interface the routine selectively calls I/O routines provided by the caller” [answer-page 10] and concludes that it would have been obvious “to combine Coker and Archer because the extended parameter list of Archer would enable the passing of all control statements and enables the caller to specify the data definition name. Both systems increase the efficiency of execution by expanding COBOLS I/O statements/parameters” [answer-

page 11].

Despite our failure to understand, from the examiner's explanation, what would have led the artisan to combine the teachings of these references, the major flaw in the examiner's reasoning is the determination that, somehow, Coker teaches a "file system identifier" in source program statements. We fail to find any such "file system identifier" in a source program statement, as claimed, in Coker.

When it was pointed out by appellants, at page 6 of the principal brief, that Coker may teach the interchangeability of file systems for use in conjunction with COBOL programs but it does not teach the use of source language statements, acting as compiler directives, to identify the desired file system, the examiner's response was to change course and now hold, in the answer, that it is really appellants' admitted prior art [from page 2, line 11, through page 3, line 11, of the specification] that taught this feature. More particularly, the examiner points to the language, "As a general rule, a file name is specified by COBOL source language statements using the "SELECT" and "ASSIGN" clauses."

Notwithstanding the impropriety of the examiner changing the reasons for the rejection this late in the prosecution, we are unpersuaded by the examiner's "new" reasoning because we agree with appellants that while the "Background of the Invention" section of the instant specification may describe how a "file name" can be

specified using source language statements and it may assert the need for making the selection of a “file-system” as easy as possible [reply brief-page 4], it is appellants who solve this problem by allowing a “file system” to be specified by source language statements so that a resulting object program includes different instructions for performing the input and output operations depending on the “file-system” specified by the source language statements. The examiner has offered nothing as to why the claimed file-system identifier provided by the source program statements would be the same as merely specifying a “file name.”

The examiner has pointed to nothing within Coker, Archer, or the background section of the instant specification which convinces us that it was known or would have been obvious to provide for “wherein at least one of the received statements includes a file-system identifier that permits selection of a file system for performing input and output operations in the source program and the file-system identifier is selected from a group comprising a literal, an environment variable, and a data name; and...compiling the source program into an object program in the memory of the computer, wherein the object program includes instructions for performing the input and output operations according to the file-system identifier that permits the selection of the file system.”

## CONCLUSION

Appeal No. 2002-0695  
Application No. 09/132,731

We have sustained the rejection of claims 1-42 under obviousness-type double patenting but we have not sustained the rejection of claims 1-9, 15-24 and 30-37 under 35 U.S.C. § 103.

Accordingly, the examiner's decision is affirmed.

No time period for taking any subsequent action in connection with this appeal may be extended under 37 CFR § 1.136(a).

AFFIRMED

ERROL A. KRASS	)	
Administrative Patent Judge	)	
	)	
	)	
	)	
	)	BOARD OF PATENT
MICHAEL R. FLEMING	)	APPEALS
Administrative Patent Judge	)	AND
	)	INTERFERENCES
	)	
	)	
	)	
HOWARD B. BLANKENSHIP	)	
Administrative Patent Judge	)	

eak/vsh

Appeal No. 2002-0695  
Application No. 09/132,731

GATES & COOPER, LLP  
HOWARD HUGHES CENTER  
6701 CENTER DRIVE WEST  
SUITE 1050  
LOS ANGELES, CA 90045