

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. 19

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte MICHAEL J. BRANSON,
DAVID J. MISHESKI and
STEPHEN M. STUPCA

Appeal No. 1999-1858
Application 08/636,211

ON BRIEF

Before FLEMING, RUGGIERO and BARRY, **Administrative Patent Judges**.

FLEMING, **Administrative Patent Judge**.

DECISION ON APPEAL

This is a decision on appeal from the final rejection of claims 1, 20-23, 32-34, 58, 65-67, and 70-72¹. Claims 39-57

¹ Although Appellants' Brief states at pages 3 and 12 that claims 1-20 and 22-38 stand or fall together based on claim 1, and that claims 21, 58-61 and 65-69 stand or fall together based on claim 21, claims 2-19, 24-31, 35-38, 59-61, and 68-69 have not been rejected and are therefore are not before us on this appeal.

Appeal No. 1999-1858
Application No. 08/636,211

and 62-64 are allowed. Claims 2-19, 24-31, 35-38, 59-61 and 68-69 are objected to for depending upon rejected claims.

The invention relates to a computer system and method for processing a source code with an object oriented framework (specification, page 4, lines 1-2). The system uses a CPU (figure 8, item 810) and a main memory (figure 8, item 820). The main memory contains a source code processing framework (figure 8, item 870). The framework provides an infrastructure that embodies the steps necessary to process source code and a mechanism to extend the framework to define a particular source code processing environment (specification, page 4, lines 2-5). Certain core functions are provided by the framework that cannot be modified by the framework user, which interact with extensible functions provided by the framework user extending extensible classes in the framework (specification, page 4, lines 5-9; page 6, line 23 through page 7, line 3). The framework allows a developer to select the native source code processing tool of their choice, and provides an infrastructure for using that processing tool with other platforms (specification, page 4, lines 6-7).

Independent claim 1 is reproduced as follows:

Appeal No. 1999-1858
Application No. 08/636,211

32-34, 58, 65-67 and 70-72 under 35 U.S.C. § 103.

The Examiner has failed to set forth a *prima facie* case. It is the burden of the Examiner to establish why one having ordinary skill in the art would have been led to the claimed invention by the express teachings or suggestions found in the prior art, or by implications contained in such teachings or suggestions. *In re Sernaker*, 702 F.2d 989, 995, 217 USPQ 1, 6 (Fed. Cir. 1983). The Federal Circuit states that "[t]he mere fact that the prior art may be modified in the manner suggested by Examiner does not make the modification obvious unless the prior art suggested the desirability of the modification." *In re Fritch*, 972 F.2d 1260, 1266 n.14, 23 USPQ2d 1780, 1783-84 n. 14 (Fed. Cir. 1992), *citing In re Gordon*, 733 F.2d 900, 902, 221 USPQ 1125, 1127 (Fed. Cir. 1984). However, "[o]bviousness may not be established using hindsight or in view of the teachings or suggestions of the invention." *Para-Ordnance Mfg. v. SGS Importers Int'l*, 73 F.3d at 1087, 37 USPQ2d at 1239, *citing W. L. Gore & Assocs., Inc. v. Garlock, Inc.* 721 F.2d at 1551, 1553, 220 USPQ at 311, 312-13.

Appeal No. 1999-1858
Application No. 08/636,211

On pages 3-4 of the brief, Appellants agree with the Examiner's statement that Palevich discloses a framework in a computer system having a central processing unit and main memory, the framework inherently being stored in the main memory, that Palevich's framework includes at least one extensible class, and that Palevich does not explicitly disclose that the framework is for source code processing as claimed. Appellants disagree⁴ with

the Examiner's assertion that source code processing falls within the range of frameworks suggested by Palevich.

Appellants first assert⁵ that the range of frameworks relied on by the Examiner is a range of application frameworks and does not include frameworks for other types of software. Appellants point out⁶ that source code processing falls within the category of software development tools and not application

⁴ Brief, page 4.

⁵ Brief, page 4, final paragraph

⁶ Brief, page 5

software.

In addition, Appellants assert⁷ that the source code processing framework corresponding to the claimed invention can be used to process source code for software applications, but is not a software application itself, and is not an application framework as commonly understood in the art.

Appellants also note the five exemplary commercial application frameworks cited by Palevich⁸, and assert that they are used by programmers to write software applications, and as such are frameworks for developing software applications, and are not software development tools.

Appellants then posit that because a framework for source code processing cannot be properly characterized as an application framework, Appellants' framework for source code processing does not fall within the scope of application frameworks suggested in Palevich.

Similarly, Appellants argue⁹ that even if the range in

⁷ Brief, page 6

⁸ Column 9, lines 48-50

⁹ Brief, page 8.

Appeal No. 1999-1858
Application No. 08/636,211

Palevich is construed to include both application frameworks and system software frameworks, source code processing software is not understood in the art to be in the system software category.

Finally, Appellants assert¹⁰ that the Examiner's construction of Palevich is unduly broad absent some affirmative teaching or suggestion in the art regarding source code processing. Appellants contend that even assuming *arguendo* that the claimed invention is properly construed as an application framework or as a system software framework, the range of functions found in frameworks in the language of Palevich cannot properly be construed as an affirmative teaching or suggestion of every type of framework within that range, because the range of function of known application frameworks span a wide range of functionality does not automatically render obvious all the application frameworks in the entire range.

In the answer¹¹, the Examiner asserts that although Palevich does not explicitly disclose that the framework is

¹⁰ Brief, page 9.

¹¹ Pages 2 and 7.

Appeal No. 1999-1858
Application No. 08/636,211

for source code processing as claimed, Palevich discloses¹² that available frameworks range from high level application frameworks to lower level frameworks that provide basic system software services, thus covering source code processing functions which fall within the recited range. The Examiner then finds that it would have been obvious to one of ordinary skill in the art to implement a framework as disclosed by Palevich for source code processing functions to obtain the advantages of object oriented programming offered by the framework as suggested by Palevich¹³. The Examiner provides¹⁴ printing of the source code as an exemplary inclusion of source code processing in a framework, since Palevich discloses¹⁵ frameworks support printing.

In response to Appellants' argument that source code processing framework cannot be properly classified as an application framework, as the claimed framework is a software

¹² Column 9, lines 40-47.

¹³ Column 9, lines 31-38.

¹⁴ Answer, page 6.

¹⁵ Column 9, lines 45-47.

Appeal No. 1999-1858
Application No. 08/636,211

development tool framework and not software application or system software, the Examiner points¹⁶ to claim 1 reciting only "framework" for source code "processing" without specifying any properties thereof to indicate that the application is not an application program, except that it may be used for source code.

In response to Appellants' assertion that the source code processing development tool can be used to process source code to generate an executable application program, the Examiner notes that such limitation is not in the claims.

As regards Appellants' assertion that source code processing does not fall within the scope of application frameworks or system software, the Examiner replies that Palevich suggests that all computer functions may be implemented using frameworks, thereby including source code processing.

Initially, we note that all the rejected independent claims have limitations directed to a framework for processing source code.

Second, we note that Appellants agree with the Examiner's

¹⁶ Answer, pages 5-6.

Appeal No. 1999-1858
Application No. 08/636,211

statement that Palevich discloses a framework in a computer system having a central processing unit and main memory, the framework inherently being stored in the main memory, and that Palevich's framework includes at least one extensible class. The crux of the issues that Appellants and the Examiner disagree upon¹⁷ is the Examiner's assertion that source code processing falls within the range of frameworks suggested by Palevich.

As pointed out by our reviewing court, we must first determine the scope of the claim. "[T]he name of the game is the claim." *In re Hiniker Co.*, 150 F.3d 1362, 1369, 47 USPQ2d 1523, 1529 (Fed. Cir. 1998).

Appellants' specification states¹⁸ that "Source code in any compiled language needs to be processed". Source code is defined¹⁹ as "[t]he form in which a computer program is written by a programmer. Source code is written in some formal programming language which can be compiled automatically into

¹⁷ Brief, page 4.

¹⁸ Page 2, line 7.

¹⁹ The ©Free On-line Dictionary of Computing, 1993-2001, Denis Howe.

Appeal No. 1999-1858
Application No. 08/636,211

object code or machine code or executed by an interpreter."

Exemplary claim 1, the sole independent claims specifically addressed by the Examiner and Appellants, recites "the main memory containing a framework that provides an extensible source code processing system for processing at least one source code processing module within a source code program." The Examiner admits that Palevich does not explicitly disclose that the framework is for source code processing as claimed, but relies on the references statement²⁰ that "The types of frameworks range from high level application frameworks that assist in developing a user interface, to lower-level frameworks that provide basic system software services, such as communications, printing, file system support, graphics, etc."

We find that this framework range need not necessarily include source code processing. Other levels of code or languages are processed by the range of frameworks disclosed by Palevich. No mention is made of processing source code. When a reference is silent about the asserted inherent

²⁰ Column 9, lines 42-47.

Appeal No. 1999-1858
Application No. 08/636,211

characteristic, such gap in the reference may be filled with recourse to intrinsic or extrinsic evidence. Such evidence must make clear that the missing descriptive matter is necessarily present in the thing described in the reference, and that it would be so recognized by persons of ordinary skill. Inherency, however, may not be established by probabilities or possibilities. *In re Robertson,*

Slip Op 98-1270 (Fed. Cir. February 25, 1999) citing *Continental Can Co. v. Monsanto Co.*, 948 F.2d 1264, 1268, 20 USPQ2d 1746, 1749 (Fed. Cir. 1991). "Inherency, however, may not be established by probabilities or possibilities." *Id.* at 1269, 20 USPQ2d at 1749 (quoting *In re Oelrich*, 666 F.2d 578, 581, 212 USPQ 323, 326 (CCPA 1981)).

We are not inclined to dispense with proof by evidence when the proposition at issue is not supported by a teaching in a prior art reference or shown to be common knowledge of unquestionable demonstration. Our reviewing court requires this evidence in order to establish a *prima facie* case. *In re Piasecki*, 745 F.2d 1468, 1471-72, 223 USPQ 785, 787-88 (Fed. Cir. 1984); *In re Knapp-Monarch Co.*, 296 F.2d 230, 232, 132

Appeal No. 1999-1858
Application No. 08/636,211

USPQ 6, 8 (CCPA 1961); *In re Cofer*, 354 F.2d 664, 668, 148
USPQ 268, 271-72 (CCPA 1966). Furthermore, our reviewing
court states in *In re Piasecki*, 745 F.2d 1468, 223 USPQ 785,
788 (Fed. Cir. 1984) the following:

The Supreme Court in *Graham v. John Deere Co.*, 383
U.S. 1 (1966), focused on the procedural and
evidentiary processes in reaching a conclusion under
Section 103. As adapted to ex parte procedure,
Graham is interpreted as continuing to place the
"burden of proof on the Patent Office which requires
it to produce the factual basis for its rejection of
an application under section 102 and 103". *Citing*
In re Warner, 379 F.2d 1011, 1020, 154 USPQ 173, 177
(CCPA 1967).

Therefore, we will not sustain the rejections of claims
1, 20-23, 32-34, 58, 65-67, and 70-72 under 35 U.S.C. § 103 as
being unpatentable over Palevich.

Accordingly, the Examiner's decision is reversed.

REVERSED

MICHAEL R. FLEMING)
Administrative Patent Judge)
)
)
)
)
JOSEPH F. RUGGIERO) BOARD OF PATENT
APPEALS AND

Appeal No. 1999-1858
Application No. 08/636,211

Administrative Patent Judge)	INTERFERENCES
)	
)	
)	
LANCE LEONARD BARRY)	
Administrative Patent Judge)	

vsh

Appeal No. 1999-1858
Application No. 08/636,211

DEREK P. MARTIN
MARTIN & ASSOCIATES, L.L.C.
221 W. 4TH STREET, SUITE 2
CARTHAGE, MO 64836