



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
11/678,607	02/25/2007	Eric McCully	CA920085010US2	2186
63675	7590	03/08/2018	EXAMINER	
PATTERSON & SHERIDAN, LLP/IBM SVL 24 Greenway Plaza SUITE 1600 HOUSTON, TX 77046-2472			SYED, FARHAN M	
			ART UNIT	PAPER NUMBER
			2165	
			NOTIFICATION DATE	DELIVERY MODE
			03/08/2018	ELECTRONIC

**Please find below and/or attached an Office communication concerning this application or proceeding.**

The time period for reply, if any, is set in the attached communication.

Notice of the Office communication was sent electronically on above-indicated "Notification Date" to the following e-mail address(es):

PAIR\_eofficeaction@pattersonsheridan.com  
PSDocketing@pattersonsheridan.com

UNITED STATES PATENT AND TRADEMARK OFFICE

---

BEFORE THE PATENT TRIAL AND APPEAL BOARD

---

*Ex parte* ERIC MCCULLY

---

Appeal 2016-000110<sup>1</sup>  
Application 11/678,607  
Technology Center 2100

---

Before JEREMY J. CURCURI, BARBARA A. BENOIT, and  
JON M. JURGOVAN, *Administrative Patent Judges*.

CURCURI, *Administrative Patent Judge*.

DECISION ON APPEAL

Appellant appeals under 35 U.S.C. § 134(a) from the Examiner's rejection of claims 1, 3–11, and 13–21. Final Act. 1. We have jurisdiction under 35 U.S.C. § 6(b).

Claims 1, 3–11, and 13–21 are rejected under 35 U.S.C. § 103(a) as obvious over Anand (US 5,721,903; Feb. 24, 1998), and Kolton (US 5,778,357; July 7, 1998). Final Act. 5–19.

---

<sup>1</sup> Appellant identifies the real party in interest as International Business Machine Corporation. App. Br. 3.

We reverse.

### STATEMENT OF THE CASE

Appellant's invention relates to "customized retrieval and presentation of information from a relational database interfaced to a TCP /IP network."

Spec. ¶ 2. Claim 1 is illustrative and reproduced below, with the key disputed limitations emphasized:

1. An apparatus for generating a report, comprising:
  - a. one or more server systems, each having at least a processor and a memory storing one or more applications, executed on the processor to generate the report by:
    - i. providing a user interface configured to receive user input indicating a selection of a predefined report specification, the predefined report specification comprising code configured to retrieve information from a database and present a report of the retrieved information;
      - ii. prompting, via the user interface, for further user input, wherein the further user input specifies one or more modifications for the report generated by the predefined report specification;*
      - iii. generating a script corresponding to the further user input, wherein the script is configured to modify the code of the predefined report specification;*
    - b. a first database comprising predefined report specifications;
    - c. a second database comprising information for incorporation into reports;
    - d. wherein the one or more server systems further generate the report by:
      - i. finding, in the first database, the predefined report specification corresponding to the user input;
      - ii. loading the predefined report specification;
      - iii. applying the script to modify the predefined report specification to create a customized report specification by modifying the code of to the predefined report specification based on the further user input;
      - iv. running the customized report specification, including retrieving appropriate information from the second database to generate a customized report; and

v. outputting the report to the user interface.

## ANALYSIS

### *Contentions*

The Examiner finds Anand and Kolton teach all limitations of claim 1. Final Act. 5–10. In particular, the Examiner finds Kolton teaches the key disputed limitations of claim 1, and specifically finds Kolton’s user creating a query describes “generating a script” as recited in claim 1. *See* Final Act. 8–9 (citing Kolton Figures 3A–3B; col. 3, ll. 60–67; col. 4, ll. 21–23, 53–67; col. 5, ll. 1–16, 18–49).

The Examiner reasons

It would have been obvious to combine the teachings of Anand with the teachings of Kolton to include i. prompting, via the user interface for further user input, wherein the further user input specifies one or more modifications for the report generated by the first application and ii. generating a script corresponding to the further user input wherein the script is configured to modify code of the first application with the motivation for generating reports from a computer database which allows a user to retrieve and analyze data.

Final Act. 10 (citing Anand col. 1, ll. 54–55).

Regarding the key disputed limitations, Appellant presents the following principal arguments:

[A] series of overlaying windows to build a query does not disclose generating a script. And again the query built from the “overlapping windows” does not in any way “modify the predefined report specification to create a customized report specification by modifying the code of [] the predefined report specification.” Instead, the query is used to retrieve data from a database. Again, primarily financial market data retrieved using a database query.

That is, the “query” such [as] a database query is not a “script” that is “configured to modify code of the first application,” as claimed. The “series of overlaying window queries and prompts” are not used to generate a script. Instead, the “series of overlaying window queries and prompts,” are used to build a database query, again, one related to the pricing of a financial instrument. The resulting database query composed using the “series of overlaying window queries and prompts,” in no way “modifies code of the first application,” as claimed — or modif[ies] any code of any application.

App. Br. 15–16; *see also* App. Br. 17 (“First, again, there is no script being generated, a database query is being composed. Second, the query composed using this interface does not ‘modify code of the predefined report specification’ in the manner claimed (or otherwise). Instead, it provides a query used to retrieve and format financial data pulled from a database.”) and Reply Br. 4–5.

#### *Our Review*

We review the appealed rejections for error based upon the issues identified by Appellant, and in light of the arguments and evidence produced thereon. *Ex parte Frye*, 94 USPQ2d 1072, 1075 (BPAI 2010) (precedential).

Kolton (col. 3, ll. 60–67, emphasis omitted) discloses

The windows interface program 104 places in front of the user 105 templates which list the alternative query 10[7] formulations that the user can select using a mouse pointing device and a typewriter to prepare specific queries. The query windows, illustrated below, correspond to the language options that are set forth in a formal description of the query language that is also presented below.

Kolton (col. 4, ll. 21–23, emphasis omitted) discloses “When the user is finished defining the query and calls for the generation of reports based upon the query, the query representation 108 is compressed[.]”

Kolton (col. 4, ll. 53–67, emphasis omitted) discloses

The databases shown in FIG. 2 used in combination with the system 100 shown at FIG. 1 creates the potential for the user to generate reports that are of particular use to market traders and analysts in trying to examine past actions of the market at certain times, such as when dividends are paid or when special market events occur. In addition, the user may add the user's own market domain knowledge to the market domain knowledge 208 within the database 114 and thereby create a separate body of information for special reports. But to realize the potential of the database, a carefully human-engineered interface between the user 105 and the remaining parts of the system is provided in the form of the queries 107 which are implemented using overlapping windows and which are based upon an English-like search query language.

Kolton (col. 5, ll. 1–17, emphasis omitted) discloses

FIGS. 3A and 3B illustrate a way that user queries are processed in two different embodiments of the invention. In FIG. 3A, the user 105 interacts directly with the windows interface 104 to create and later to modify the query. During this interaction, information from the windows interface 104 enables an echo generator 302 to construct a natural language formulation of the query (or partial query) which is displayed on a screen 306 to the user but which is not directly editable by the user. When the query formulation process is completed, the windows data structure 304, created by the windows interface 104, is passed in a simplified form, stripped of redundancies, to an execution engine 110 which parses the data structure and then executes the database retrieval actions for and also causes the necessary reports to be generated, and formatted for printing and display. Note that report definition is inherent in the search query formulation process.

Kolton (col. 5, ll. 31–35) discloses “The user is thus given the choice of entering and revising the query either by writing and revising an ASCII command string or by working through the series of overlaying window queries and prompts.”

Thus, in at least some sense, Kolton describes claim 1's "prompting, via the user interface, for further user input, wherein the further user input specifies one or more modifications for the report" because Kolton's entering and then revising the query specifies and then modifies a report definition. *See* Kolton col. 5, ll. 16–17. However, even if we assume, but do not decide, that a query is a script, and that Kolton's query generation describes claim 1's "generating a script corresponding to the further user input," Kolton's query is not, as required by claim 1, "configured to modify the code of the predefined report specification."

First, Kolton's report is defined in the search query formulation process and is not a predefined report specification selected by the user that is modified by the query. *See* Kolton col. 5, ll. 16–17. And second, Kolton's query is executed to cause database retrieval actions and report generation; that is, Kolton's query itself is not *configured to modify code*, let alone to modify code of a (predefined) report specification. *See* Kolton col. 5, ll. 9–16. And, we do not readily see any articulation by the Examiner of how, or why, Kolton's query would or even could be configured to modify code of a (predefined) report specification.

The Examiner's further explanation in the Examiner's Answer does not change our result. The Examiner explains:

*The Examiner notes, see Appellant's disclosure, paragraph [0008] of the Background of the Invention, that it is well known in the art that in any managed reporting environment there is a report server, a computer running a program providing a user interface, receiving input from the user, requesting a specified predefined repor[t], ... generating a report, and sending the report to the user interface. Consistent with the Appellant's disclosure with respect to a computer running a program, Kolton, see column 5, lines 2-4, which discloses the user*

*interacts directly with the windows interface to create and later modify the query, thereby illustrating that the query script (e.g. program/code) may be modified by the user. See also Kolton, column 5, lines 31-35, discloses the user is given the choice of entering or revising the query by either writing and revising an ASCII command string or by working through a series of overlaying window queries and prompts. The Examiner interprets the series of overlaying window queries and prompts as a user generating a script and the choice of revising as a further user input, since the query and the reports were previously generated as described in column 5, lines 1-16.*

Ans. 6.

The Examiner emphasizes that Kolton's query (purported script) may be modified by the user. But as we explained above, the fact that the user in Kolton may modify the query is not sufficient to teach the key disputed claim limitations because Kolton's query itself is not *configured to modify code*, let alone to modify code of a (predefined) report specification. See Kolton col. 5, ll. 9–16.

We, therefore, do not sustain the Examiner's rejection of claim 1, or of claims 3–10, which depend from claim 1.

Independent claim 11 recites the same key disputed limitations. We, therefore, do not sustain the Examiner's rejection of independent claim 11, or of claims 13–21, which depend from claim 11.

#### ORDER

The Examiner's decision rejecting claims 1, 3–11, and 13–21 is reversed.

REVERSED