United States District Court,
N.D. California.

BUSINESS OBJECTS, S.A,
Plaintiff.
v.
MICROSTRATEGY, INC,
Defendant.

No. C 01-3908 CRB

May 1, 2003.

Daniel J. Furniss, Stephen Y. Pang, Joseph A. Greco, Townsend and Townsend and Crew LLP, Palo Alto, CA, Gregory S. Bishop, Goodwin Procter, LLP, Iris Sockel Mitrakos, Townsend and Townsend and Crew LLP, San Diego, CA, Mehrnaz Boroumand, Townsend and Townsend and Crew, LLP, San Francisco, CA, for Plaintiff.

James Charles Pistorino, James F. Valentine, Howrey LLP, East Palo Alto, CA, Joseph P. Lavelle, Howrey, Simon, Arnold & White, LLP, Matthew Scott Zises, Howrey, Simon, Arnold & White, Menlo Park, CA, Peter Edward Moll, Esq., Howrey, Simon, Arnold & White, LLP, Washington, DC, Thomas C. Mavrakakis, Palo Alto, CA, for Defendant.

CLAIM CONSTRUCTION ORDER

BREYER**, J.**

This suit involves the alleged infringement by defendant Microstrategy, Inc. ("Microstrategy") of Claims 1, 2, and 4 of U.S. Patent No. 5,555,403 ("the '403 patent") owned by plaintiff Business Objects, S.A. ("Business Objects"). Now before the Court is the task of claim construction.

The '403 patent relates to a software system for obtaining data from a relational database. Typically, data from a relational database is obtained by sending a query to the database specifying the information that is desired. The query usually takes the form of Structured Query Language ("SQL"), a query programming language. The complexity of SQL makes it difficult for a typical user to obtain information from a database. The '403 patent describes software that allows a user who is unfamiliar with SQL or the structure of the database to create queries by simply identifying familiar "business objects" (e.g., "Customers" or "Quantity Sold") and conditions (e.g., "Quantity Sold greater than 100") related to the desired information. The software takes the information provided by the user and generates SQL queries to send to the database using a "query engine." The software is designed to be semantically dynamic such that the meaning of a particular business object may change depending on the other business objects selected with it. For example, if a user selects the business objects "Customers" and "Sales Revenue," the information retrieved from the database will be a listing of sales revenue by customer. On the other hand, if a user selects the business objects "Items" and "Sales Revenue," the information retrieved from the database will be a listing of sales revenue by items sold. Thus, the meaning of the business object "Sales' Revenue" and the sales revenue data actually retrieved will be different depending on whether the user associates "Sales Revenue" with "Customers" or with "Items."

Before the software system can be used by end-users, certain information must be provided to the software by a software manager. This information includes a definition of the business objects as well as information pertaining to all the potential combinations of data that a user may be interested in (called "joins" in SQL). The business objects are defined by assigning them a familiar name that the user will recognize, elements of a "SELECT clause," and in some cases, elements of a "WHERE clause." "SELECT" and "WHERE" are commands used in SQL. The elements of SELECT and WHERE clauses provided in the definition of business objects contain information that the query engine uses to create the SQL query.

After the software manager has set up the system, a user can request information from the database by selecting one or more business objects and then optionally specifying conditions to restrict the query results. The user may also specify how the results should be sorted. The query engine follows an algorithm that uses the elements of SELECT and WHERE clauses in the definitions of the business objects that the user has selected, the conditions specified by the user, and other information provided by the manager to generate a SQL query that is sent to the database. The SQL query contains its own SELECT and WHERE clauses that, in general, are not the same as the elements of the SELECT and WHERE clauses contained in the business object definitions.

DISCUSSION

I. Legal Standards for Claim Construction

Patent infringement analysis involves two steps. The first step is to construe the asserted claims and the second step is to determine whether the accused method or product infringes any of the claims as properly construed. *See* Markman v. Westview Instruments, Inc. 52 F.3d 967, 976 (Fed.Cir.1995) (en banc), *aff'd,* 517 U.S. 370, 116 S.Ct. 1384, 134 L.Ed.2d 577 (1996). The first step, construction of the patent claims, is a matter of law and thus the responsibility of the court. *See* id. at 979.

"In interpreting an asserted claim, the court should look first to the intrinsic evidence of record, i.e., the patent itself, including the claims, the specification and, if in evidence, the prosecution history." Vitronics Corp. v. Conceptronic, Inc., 90 F.3d 1576, 1582 (Fed.Cir.1996). In examining the intrinsic evidence, the court should first look to the words of the claims themselves to define the scope of the patented invention. *See id*. Words in a claim "are" generally given their ordinary and customary meaning." *Id*.

Second, the court should review the patent specification "to determine whether the inventor has used any terms in a manner inconsistent with their ordinary meaning." *Id*. "The specification acts as a dictionary when it expressly defines terms used in the claims or when it defines terms by implication." *Id*. The Federal Circuit teaches that "the specification is always highly relevant to the claim construction analysis. Usually it is dispositive; it is the single best guide to the meaning of a disputed term." *Id*. Drawings included in the patent application have the same effect on claim language as other portions of the specifications. *See* Autogiro Co. of America v. United States, 181 Ct.Cl. 55, 384 F.2d 391, 398 (Ct.Cl.1967).

The third type of intrinsic evidence that the Court may consider is the prosecution history of the patent, if it is in evidence. *See* Vitronics, 90 F.3d at 1582. The prosecution history contains the entire record of the prosecution of the patent claim before the patent office, including any representations about the scope of the claim or the meaning of certain terms made by the applicant.

Ordinarily, the intrinsic evidence alone will resolve any ambiguity in a disputed term. By relying first on the claim language, the specification, and the prosecution history, a court can protect a patentee's rights while at the same time enabling the public to rely on the public record of the patentee's claim. "In other words, competitors are entitled to review the public record, apply the established rules of claim construction, ascertain the scope of the patentee's claimed invention and, thus, design around the claimed invention." Vitronics, 90 F.3d at 1583 (citing Markman, 52 F.3d at 978-79). For these reasons, "[o]nly if there [is] still

some genuine ambiguity in the claims, after consideration of all available intrinsic evidence, should the trial court [ ] resort[ ] to extrinsic evidence." *Id.* at 1584; *see* Key Pharmaceuticals v. Hercon Labs. Corp., 161 F.3d 709, 716 (Fed.Cir.1998) (noting that extrinsic evidence is appropriate if the intrinsic evidence "does not answer the question").

II. Prior Claim Construction

Business Objects has twice brought the '403 patent before this Court in infringement actions against other defendants. *See Business Objects, S.A. v. Brio Technology, Inc.,* No. C 97-4386 CRB (N.D.Cal.1997); *Business Objects v. Cognos, Inc.,* No. C 00-1816 CRB (N.D.Cal.2000). Claim construction orders were issued in both of these cases. Business Objects urges the Court to adopt its previous constructions of contested terms where available. Microstrategy presents arguments that would change or extend the constructions of some of the previously construed terms.

Since the Supreme Court issued its decision in *Markman,* a split has developed among trial courts regarding the preclusive effect of *Markman* orders. Some courts have held that *Markman* orders have collateral estoppel effect against plaintiffs in infringement actions even when the original action settles. *See, e .g.,* TM Patents v. IBM Corp., 72 F.Supp.2d 370 (S.D.N.Y.1999). Other courts have refused to recognize collateral estoppel in similar circumstances. *See, e.g.,* Kollmorgen Corp. v. Yaskawa Electric Corp., 147 F.Supp.2d 464 (W.D.Va.2001); *see also* Timothy Le Duc, *The* Application of Collateral Estoppel to Markman Rulings: The Search for Logical and Effective Preclusion of Patent Claim Constructions, 3 Minn. Intell. Prop. Rev. 297 (2002). The Federal Circuit has indicated a willingness to resolve this split, but it has not yet done so. *See* Kollmorgen Corp. v. Yaskawa Electric Corp., 2001 WL 1356727 (Fed.Cir.2001) (unpublished order granting permission to appeal interlocutory order certified by the district court), *dismissed by party agreement,* 2002 WL 554402 (Fed.Cir.2002) (unpublished order).

Whether or not a plaintiff is collaterally estopped from asserting constructions of claim terms different from those adopted in prior litigation, it is clear that a prior claim construction has no such estoppel effect vis-a-vis a defendant who was not a party to the earlier litigation. In its *Markman* decision, the Supreme Court noted that notwithstanding the goal of uniformity in claim construction, "issue preclusion could not be asserted against new and independent infringement defendants even within a given jurisdiction." 517 U.S. at 391. While a court may defer to a previous claim construction at its discretion, Texas Instruments, Inc. v. Linear Technologies Corp., 182 F.Supp.2d 580, 589 (E.D.Tex.2002); *see also* KX Indus., L.P. v. PUR Water Purification Prods., Inc., 108 F.Supp.2d 380, 387 (D.Del.2000) ("While the court's previous opinion does not have issue preclusive effect against [the defendant] in this case, to the extent parties do not raise new arguments, the court will defer to its previous construction of the claims."), "where [new] defendants have new arguments to bring to the attention of the court, defendants' rights to fully litigate their claims are particularly persuasive." Texas Instruments, 182 F.Supp.2d at 589.

III. Claim Construction

A. The "associating" step

Claim 1 calls for a method consisting of the following four steps:

associating a first familiar name with a first returned value, wherein the familiar name is also associated with the following: a SELECT clause describing the values returned using a combination of the functions and operators supported by the predefined query language; a WHERE clause describing a condition which can be used to restrict the scope of the returned value; and a plurality of tables containing the attributes on which the SELECT and WHERE clauses operate;

accepting signals from the user interface to specify a query, wherein the query includes the familiar name;

generating a query in the predefined query language, wherein the query includes the condition; and

using the query to access one or more attributes in the relational database.

In the first step, a "familiar name" must be associated with four things: 1) a returned value, 2) a SELECT clause, 3) a WHERE clause, and 4) a plurality of tables. In the *Brio* case, "associating" and "associated" were determined to be "clear and unambiguous" and it was stated that "no further definition is necessary or useful." *Brio* Claim Construction Order at 1 (Bishop Decl. Ex. B). However, the *Brio* construction only dealt with the narrow meaning of the word "associating" itself. Microstrategy now asks this Court to construe the "associating" step as a whole. Specifically, Microstrategy argues that the "associating" step in Claim 1 relates to the definition of a business object by a software manager, which, as described in the specification, occurs prior to the use of the invention by an end-user to generate a database query. Therefore, Microstrategy's position is that the "associating" step occurs prior to the "generating a query step. In contrast, Business Objects argues that the associations in the "associating" step cannot occur until a query is generated.

Business Objects' position is supported by the fact that in *Brio,* this Court construed the WHERE clause to be explicitly defined by the patentee in the specification at column 11, line 22 through column 11, line 43. This part of the specification describes the operation of the query engine. Thus, this Court determined that the WHERE clause in the "associating" step is created when the query is generated. Microstrategy argues that this Court erred in *Brio,* noting that the part of the specification that the Court cited describes how a WHERE clause is generated rather than defines it. Microstrategy further argues that the WHERE clause refers to the WHERE clause elements included in the business objects' definitions rather than the WHERE clause included in the generated query.

As Business Objects points out, the fact that the familiar name must be associated with a returned value implies that a query must be generated in order to accomplish the "associating" step. The language of the claims indicates that "returned values" are only obtained from queries. Claim 1 recites that a "relational database ... provides returned values responsive to queries specified in a predefined query language." Furthermore, Claim 2, which depends from Claim 1, calls for generating queries that "result[ ] in the retrieval of a first set of returned values" and "a second set of returned values."

Nonetheless, the bulk of the claim language and the prosecution history support the contrary conclusion that the "associating" step occurs prior to the "generating a query" step and is separate and distinct from it.

By laying out the "associating" step separately from the "generating a query" step, Claim 1 appears to envision that the four associations in the "associating" step are distinct from the acts contemplated by the "generating a query" step. That the two steps are distinct is highlighted by the fact that the "associating" step requires "a WHERE clause describing a condition which can be used to restrict the scope of the returned value." At the "generating a query" step, "the query includes the condition." Business Objects argues that the WHERE clause in the "associating" step *is* the WHERE clause contained in the generated query. However, the requirement that the query merely " *include* [ ] the condition" contained in the "associating" step's WHERE clause implies that the query's WHERE clause contains, but need not be limited to, the condition contained in the "associating" step's WHERE clause. If it were as Business Objects suggests, the language requiring inclusion of the condition in the "generating a query" step would be redundant.

In addition, dependent Claim 2 implies that at least part of the "associating" step occurs prior to the generation of a query. Claim 2 adds the following three steps to those in Claim 11:

providing a selection of familiar names, wherein each familiar name is associated with one or more of the attributes in the relational database as described in claim 1;

accepting signals from the user interface to select first and second combinations of familiar names from the selection of familiar names;

generating, by using the query engine means, first and second queries based on first and second combinations, wherein the first and second combinations both include a same first familiar name but different additional familiar names, wherein the first query based on the first combination results in the retrieval of a first set of returned values based on the first familiar name, wherein the second query based on the second combination results in the retrieval of a second set of returned values based on the same first familiar name from the same database.

The language of Claim 2 indicates that these three steps must be performed in order. The second step of "accepting signals ... to select first and second combinations of familiar names" requires that the first step of "providing a selection of familiar names" already have occurred. Likewise, the third step of "generating ... queries" requires that the second step already have occurred because the queries are "based on [the] first and second combinations' selected in step two. Step one indicates that the provided familiar names are "*associated* with one or more of the attributes in the relational database *as described in claim 1*." This statement must relate to the "associating" step of Claim 1 that provides for the association of a familiar name with "a plurality of tables containing the attributes on which the SELECT and WHERE clauses operate." Therefore, by the terms of Claim 2, this association occurs prior to generating a query. Furthermore, Claim 2 requires that all of the familiar names offered for selection in step one be associated with attributes; however, not all of the familiar names are necessarily selected by the user for use in generating a query. The association of familiar names with attributes must therefore be distinct from generating a query.

The prosecution history also provides support for interpreting the "associating" step as prior to and distinct from the "generating a query" step. During prosecution, the patentee stated that the "associating" step's "WHERE clause's 'condition' is *later* used in the 'generating' step." Hocker Decl. Ex. H at 7 (emphasis added). Thus, the patentee clearly contemplated that the "associating" step occurs prior to the "generating a query" step. Furthermore, in distinguishing Claim 1 from prior art, the patentee stated that, in the prior art, "any WHERE clauses are associated with the actual SQL query and not with familiar names, thereby implying that the WHERE clause recited in Claim 1 is not the WHERE clause contained in the query.

The specification supports the conclusion that a familiar name can be associated with a returned value prior to the "generating a query" step in Claim 1. As previously described, the specification indicates that a software manager must define business objects before an end-user can use the invention to generate a query. The specification explicitly defines a business object to include "a name, some elements of a Select clause, and some elements of a Where clause." '403 Patent, 15:55-57. Several examples are provided. Id . at 5:52-7:5. In the examples, the specification represents each business object in two ways. The first representation is by an illustrative list of data to which the business object "makes reference." The second representation is by a "Query Engine definition" consisting of a name, a SQL SELECT clause, and in some cases, a SQL WHERE clause. The specification makes clear that the business objects "do not have fixed data attached to them," Id. at 7:6-10; rather, the data associated with a business object depends on how the user uses the business object in generating a query. However, the specification further indicates that "[e]very Business Object has a general meaning and can be used 'as it is' to compose a query as far as it can be considered as a column header." Id. at 8:43-45. It appears that the illustrative lists of data provided in the examples of business objects refer to data returned when the business object is used by itself to generate a query. As such, the familiar name in a business object is associated by implication with the "returned value" resulting from a query generated using the business object by itself. *See* Bell Communications Research, Inc. v. Vitalink Communications Corp. 55 F.3d 615, 621-622 (Fed.Cir.1995) (construing an "associating step" to cover "both an implicit or an explicit approach"). In this sense, the association is made prior to the actual generation of a query.

Business Objects argues that the "associating" step cannot read on the definition of business objects as described in the specification because the specification indicates that in the definition, familiar names are associated with "some elements" of SELECT and WHERE clauses and not entire SELECT and WHERE clauses. '403 Patent, 15:55-57. However, Claim 1 does not require that the SELECT and WHERE clauses be complete. The fact that familiar names may be associated with only portions of SELECT and WHERE clauses merely indicates that those situations are included within the claim's scope.

For the reasons stated above, the Court finds that the "associating" step must occur prior to the "generating a query" step. In addition, the Court finds that the WHERE clause is not explicitly defined in the specification. FN1 Finally, the SELECT and WHERE clauses in the "associating" step need not be complete; it is sufficient if the clauses contain "some elements of SELECT and WHERE clauses. '403 Patent, 15:55-57.

B. "predefined query language"

Microstrategy argues that the phrase "predefined query language" should be construed to require that the query language be known prior to the associating step in Claim 1. By including the modifier "predefined," it is clear that Claim 1 requires that the query language be determined prior to some point. Claim 1 calls for "queries specified in a predefined query language" and "generating a query in the predefined query language." Thus, the query language must be determined at least before the "generating a query" step. The associating step of Claim 1 calls for "a SELECT clause ... using a combination of the functions and operators supported by the predefined query language." While this language uses the phrase "predefined query language," it does not necessarily require that the query language be known for use in the associating step's SELECT clause. Instead, the functions and operators used by the SELECT clause must be supported by the query language of the generated query. It is conceivable that various query languages may all support the same set of functions and operators. Thus, the "predefined query language" is query language that must be determined prior to the "generating a query" step, but not necessarily prior to the associating step. Furthermore, the "predefined query language" must support the functions and operators contained in the associating step's SELECT clause.

C. "computer system"

Claim 1 calls for "a relational database operat[ing] in a computer system" and "a user interface executing on a computer system." Furthermore, Claim 1 specifies that "the computer system executing the user interface includes a processor coupled to a memory." Claim 2, which depends from Claim 1, indicates that "the computer system includes a query engine means." Independent Claim 4 calls for "a computer system including a processor coupled to a memory." Microstrategy argues that a "computer system" must mean a single computer such that the relational database and user interface operate on the same computer, or that the relational database operates on a server computer and the user interface operates on a client computer. In essence, Microstrategy contends that "computer system" does not include a system wherein software is distributed across a network of computers so that different computers perform different functions. Business Objects contends that a "computer system" can mean one or more computers.

The specification indicates that "[t]he present invention can be utilized in a PC (personal computer) environment, such as is manufactured by International Business Machines Corporation, Apple Corporation, and the like." '403 Patent, 4:7-11. In addition, the specification illustrates the software operating on a client/server architecture in Figures 37 and 38. Figures 37 and 38 illustrate the database being interfaced to a server, which is then interfaced to several clients. The client computers include both user computers and a manager computer. It is clear in Figures 37 and 38 that the database is directly connected only to the server. However, the figures are ambiguous as to which computers contain which functionalities relating to the software described by the '403 patent. Therefore, it is possible that Figures 37 and 38 in fact illustrate the claimed software being distributed across more than one computer.

In any event, the claimed invention should not be limited to a preferred embodiment. *See* Comark Communications. Inc. v. Harris Corp., 156 F.3d 1182, 1187 (Fed.Cir.1998). Accordingly, the limitations proposed by Microstrategy will not be imported into the claim. The Court adopts Business Object's construction and construes this term to refer to a system containing one or more computers and associated software.

D. "query engine means"

Claim 2 calls for "generating, by using [a] query engine means, ... queries based on ... combinations" of familiar names selected by a user. Similarly, Claim 4 recites a "query engine means for generating queries in the predetermined query language based on a given combination of two or more selected and associated familiar names." Both parties agree that the interpretation of "query engine means" is governed by 35 U.S.C. s. 112, para. 6. As such, this phrase must "be construed to cover the corresponding structure, material, or acts described in the specification and equivalents thereof." 35 U.S.C. s. 112, para. 6. In *Brio,* the corresponding structure was determined to be the query engine algorithm described in column 4, lines 42-52 column 7, lines 48-54; column 8, lines 21-23; column 9, lines 14-40; and column 9, line 52 through column 13, line 2. Both parties agree that the structure identified in *Brio* is proper. However, Microstrategy asks the Court to additionally declare that an infringing product must have each of the eight steps and associated sub-steps recited at column 9, line 52 through column 13, line 2. Business Objects argues that no further interpretation by the Court is needed. Microstrategy's concern about whether all eight steps and associated sub-steps are required for infringement is premature. Such an issue is better addressed in the context of the second step of infringement analysis: comparing the accused product with the claims, either in the context of a summary judgment motion or jury instructions. *See* Vitronics, 90 F.3d at 1581-1582.

E. "selection means"

Claim 4 calls for a "selection means coupled to the user interface for allowing the human user to select familiar names and to associate two or more familiar names together." The parties agree that this phrase is governed by 35 U.S.C. s. 112, para. 6. As Microstrategy points out, the "selection means" must be capable of performing two functions: 1) allowing the human user to select familiar names and 2) allowing the human user to associate two or more familiar names. Business Objects argues that the corresponding structure for these two functions is the "suitable mouse type configuration" in column 13, lines 13-15. Microstrategy argues that the corresponding structure for the first function is the "List of Objects" window that displays a list of business objects and business object classes from which the user can select business objects. This window is described at column 13, line 52 through column 14, line 11. For the second function, Microstrategy points to the "Result Objects" window where the list of selected business objects is displayed. This structure is described at columns 13, line 35-45. Thus, the dispute centers on whether the "selection means" is hardware or software.

The "selection means" is "coupled to the user interface." '403 Patent, Claim 1. In *Cognos,* "user interface" was construed to mean "an interface that enables information to be passed between a human user and hardware or software components of a computer system." *Cognos* Markman Order at 10 (Bishop Decl. Ex. C). This definition encompasses all features of the interface including human input through hardware such as a keyboard and mouse, human perception of computer output through hardware such as a monitor or printer, and the specific output provided by the computer, such as through windows generated by the software. Because the "selection means" is "coupled to the user interface," and the "user interface" encompasses both hardware and software components, it follows that the "selection means" can also encompass both hardware and software components. Indeed, it is difficult to see how the claimed functions can be performed by either hardware or software in isolation. For example, a user could not use a mouse to "associate" something without the software providing a mechanism for the association. In the instant case, the mechanism is the "Results Object" window, which provides a location where "a user places Business Object[s] in [ ] empty cells, in order to construct a [ ] linear association of Business Objects." '403 Patent,

13:34-34. Likewise, software does not provide a means for allowing a human user to "select" or "associate"; hardware is required in order to receive signals from the user. Therefore, the structures proposed by both of the parties are required. The corresponding structure to "selection means" includes the mouse structure at column 13, lines 13-15; the "List of Objects" window described at column 13, line 52 through column 14, line 11 and Figure 4; the "Result Objects" window described at column 13, lines 35-45; and equivalents thereof.

IT IS SO ORDERED.

FN1. This finding is contrary to the Court's construction of "WHERE clause" in *Brio*.

N.D.Cal.,2003.
Business Objects, S.A. v. Microstrategy, Inc.

Produced by Sans Paper, LLC.